Optimizing the LSTM Model for Electricity Price Forecasting using the MSADBO Algorithm

Fatemeh Ahmadvand
Department of Control and Computer Engineering
Politecnico di Torino
Torino, Italy
fatemeh.ahmadvand@studenti.polito.it

Mahmood Hosseini Imani
Department of Energy (DENERG)
Politecnico di Torino
Torino, Italy
mahmood.hosseiniimani@polito.it

Hamid Taheri Andani
Department of Electronic Engineering
Isfahan University of Technology
Isfahan, Iran
hamid.taheriandani80@gmail.com

Mansour Taheri Andani
School of Manufacturing Systems and Networks
Arizona State University
Tempe, AZ
mtaheria@asu.edu

***Abstract*— Achieving accurate electricity price prediction is essential for market participants aiming to maximize their profits. In this respect, forecasting wholesale electricity market price plays a pivotal role. The advanced nonlinear modeling capabilities of deep learning networks have shown notable success in forecasting electricity prices. However, designing such networks, particularly selecting their optimal hyperparameters, still remains a big challenge. This paper presents a Modified Sine Algorithm Dung Beetle Optimization algorithm (MSADBO) as an improved Dung Beetle Optimization (DBO) algorithm to optimize the hyperparameters of the Long Short-Term Memory (LSTM) neural network to handle the electricity price forecasting in the Italian wholesale market. After optimizing the three most important hyperparameters of the LSTM using MSADBO, the efficiency of the model has shown a drastic improvement. Comparisons with conventional and machine learning-based approaches show that the LSTM-MSADBO model is highly precise in predictions with increased robustness and effectively captures volatility in the electricity market price.**

Keywords*— *Electricity Price Forecasting, Long Short-Term Memory (LSTM), Hyperparameter Optimization, MSADBO, Italian Electricity Market, PUN

I. Introduction

Electricity prices significantly impact on the dynamics of the electricity market, making accurate electricity price forecasting essential for stakeholders engaged in competitive market environments. Various forecasting methods for electricity prices include statistical models [1] and artificial intelligence (AI)-based approaches. Statistical techniques typically involve methods such as exponential smoothing and time series analysis, with notable examples being the AutoRegressive Integrated Moving Average (ARIMA) [2], Generalized AutoRegressive Conditional Heteroskedasticity (GARCH), and multivariate regression models [3].

Other AI-based approaches, such as Artificial Neural Networks (ANN) [4], ensemble models [5], Support Vector Machines (SVM) [6], and meta-heuristic optimization algorithms [7], show superior performance in capturing nonlinear patterns of data, thus yielding higher accuracy in the forecast of electricity prices. Deep learning has emerged as a prominent area of research, gaining attention for its exceptional performance across various applications, such as language modeling [8] and natural language inference [9].

Unlike traditional methods of machine learning, deep neural networks have the capability to capture high-order nonlinear complexities using hierarchical and distributed feature representation.

Recurrent Neural Network (RNN) is a powerful deep learning technique designed to handle sequential data, including applications like audio signals [10] and financial market data [11]. Long Short-Term Memory (LSTM) networks, a specialized type of RNN, was initially proposed by Hochreiter and Schmidhuber [12]. LSTM networks are designed to deal with vanishing and exploding gradients in the training process of RNNs. Thus, those are appropriate choices in several applications, such as predicting electricity prices and load consumption [13]–[16].

Different methods exist for selecting hyperparameters, trial and error, optimization, and grid searches. The first approach, trial and error, may not result in an optimal solution. Thus, optimizing the hyperparameters is a potential strategy to enhance the effectiveness of the LSTM models [17].

This study examines the impact of different optimization methods on LSTM performance and compares the performance of optimized models with other AI models.

The structure of the paper is outlined as follows: the forecasting methods and hyperparameter optimization algorithms are detailed in Section II. Section III describes the overall framework of the Italian day-ahead electricity market and outlines the input data used as a case study. A thorough analysis of the model outcomes and the study's conclusion are provided in Section IV and Section V, respectively.

II. Methodologies and Performance Evalouation Metrics

Different types of neural networks, including LSTM, have been widely applied in power system prediction, particularly electricity price prediction. Tunning their hyperparameters like the batch size, the number of LSTM units, and learning rate, may improve their performances. This study investigates and compares the impact of Grey Wolf Optimization (GWO), Dung Beetle Optimization (DBO), and MSADBO on the LSTM's performance.

A. Long Short-Term Memory (LSTM)

The LSTM network has been designed to handle the sequential data's long-term pattern and address the vanishing gradient issue associated with traditional RNNs. In this regard, LSTM has incorporated a memory cell alongside three gates: 1) forget gate, 2) input gate and 3) output gate. These gates control how information flows, allowing the network to determine what data should be retained, updated, or removed over time. The key mathematical operations within an LSTM cell are represented by the following equations [18]:

Forget gate (Eq. (1)): this gate decides how much of the prior cell state is necessary to retain at the current time step. h_{t-1} represents previous output, $x_t$is the current input, and σ denotes the sigmoid activation function.

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

Input Gate (Eqs. (2) and (3)): the input gate decides which values will update the cell state, while $\widetilde{C_t}$ is the candidate cell state calculated using a hyperbolic tangent (tanh) function.

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

$$\widetilde{C_t} = \tanh\,(W_C[h_{t-1}, x_t] + b_C) \tag{3}$$

Cell State Update (Eq.(4)): the cell state is updated based on the forget gate and the input gate, allowing the LSTM to maintain relevant information over long sequences. C_{t-1} represents the previous memory cell.

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \widetilde{C_t} \tag{4}$$

Output Gate (Eqs. (5) and (6)): the output gate regulates the hidden output state (h_t) that is passed to the next time step and the final output of the LSTM cell.

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

$$h_t = o_t \cdot \tanh\,(C_t) \tag{6}$$

In these equations, W and b denote weight matrices and bias vectors associated with each gate.

The selection of hyperparameters in an LSTM network is essential for attaining optimal outcomes. Optimization algorithms such as GWO, DBO, and their enhanced variants, like MSADBO, are increasingly utilized to systematically fine-tune these parameters—such as weight matrices, bias vectors, the number of hidden units, learning rate, batch size, and dropout rate—thereby improving the effectiveness of the LSTM model.

B. Grey Wolf Optimization (GWO)

The GWO is a metaheuristic optimization technique inspired by the cooperative hunting strategies of grey wolves and the hierarchical organization within their packs [19]. The leader, or best solution, is represented by the alpha wolf in this hierarchy, whilst the beta, delta, and omega wolves stand for inferior options. The fundamental mechanism of GWO is to update the wolves' positions relative to the prey's position, which is the optimal solution.

This process is mathematically modeled using a set of equations that simulate the hunting strategy of grey wolves [20]:

$$\left|\vec{C} \cdot \vec{X}_p(t) - \vec{X}(t)\right| \tag{7}$$

$$\vec{X}(t+1) = \vec{X}_p(t) + \overrightarrow{A} \cdot \vec{D} \tag{8}$$

where $\vec{D}$ is the distance between the prey's position $\vec{X}_p(t)$ and the wolf's current position $\vec{X}(t)$. $\vec{A}$ and $\vec{C}$ are coefficient vectors that control the movement of wolves, dynamically adjusted in each iteration. The coefficients $\vec{A}$ and $\vec{C}$ are defined as follows:

$$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a} \qquad \vec{C} = 2 \cdot \vec{r}_2 \tag{9}$$

where $\vec{a}$ is linearly reduced from 2 to 0 over iterations, $\vec{r}_1$ and $\vec{r}_2$ are random vectors in the range [0, 1].

Dynamic adjustment of the search space in the GWO algorithm makes it an appropriate algorithm to optimize complex tasks, e.g., hyperparameter tuning in neural networks.

C. Dung Beetle Optimization (DBO)

The DBO algorithm is designed based on dung beetles' behavior. The behavior involves rolling, dancing, foraging, breeding, and stealing [21]. This algorithm consists of a colony with three kinds of dung beetles: rolling Dung Beetles (DBs), breeding DBs, and stealing DBs. The updating process of the position of a rolling dung beetle can be described as [22]:

$$y_j(t+1) = y_j(t) + c \times \Delta y + \mu \times k \times y_j(t-1) \tag{10}$$

Here, $\Delta y = \left|y_j(t) - Y_{worst}\right|$, where t represents the number of iterations, and $y_j(t)$ is the position of the *j-th* rolling DB at iteration t. The parameter k denotes the deflection coefficient, and c is a constant between (0, 1). The variable μ is either -1 or 1, indicating the direction of the position update. Breeding DBs utilize a bounded region to spawn new positions. The boundaries for this process are:

$$H^+ = \min\,(Y^* \times (1-S), H) \tag{11}$$

$$L^- = \max\,(Y^* \times (1-S), L) \tag{12}$$

where, Y^* represents the current locally optimal position, while S is a factor adjusting the boundary. H^+ and L^- denote the upper and lower bounds, respectively.

The position of the *j-th* breeding DB at *t-th* iteration is given by:

$$C_j(t+1) = Y^* + c_1 \times \left(C_j(t) - L^-\right) + c_2 \times \left(C_j(t) - H^+\right) \tag{13}$$

In this equation, c_1 and c_2 are independent random vectors that aid in exploring the search space.

The boundaries of the feeding area for small DBs are defined as:

$$H^b = \min\,(Y^* \times (1-S), H) \tag{14}$$

$$L^b = \max\,(Y^* \times (1-S), L) \tag{15}$$

The location of the *j-th* small DB at the *t-th* iteration is calculated as:

$$y_j(t+1) = y_j(t) + D_1 \times \left(y_j(t) - L^b\right) + D_2 \times \left(y_j(t) - H^b\right) \tag{16}$$

where D_1 is a normally distributed random variable, and D_2 is a random vector.

The stealing behavior, involving a thieving DB, is described as:

$$y_j(t+1) = Y^b + S \times m \times (|y_j(t) - Y^*| + |y_j(t) - Y^b|) \quad (17)$$

In this case, m is a random vector following a normal distribution, and S is constant.

D. Modified Sine Algorithm Dung Beetle Optimization (MSADBO)

The MSADBO algorithm was introduced to address some limitations of the DBO algorithm, such as its tendency to get stuck in local optima. MSADBO integrates Bernoulli mapping, an enhanced Modified Sine Algorithm (MSA), and a Gaussian-Cauchy perturbation strategy [23].

In MSADBO, Bernoulli mapping is utilized for initializing the location of the DB individuals. The mapping formula is given by [22]:

$$x_{n+1} = \begin{cases} x_n/(1-\delta), & 0 \le x_n \le 1-\delta \\ x_n(1-\delta)/\delta, & 1-\delta \le x_n \le 1 \end{cases} \quad (18)$$

Here, δ is a parameter in the interval (0,1) that introduces chaotic behavior into the initialization process.

The MSADBO algorithm adopts the MSA to ensure robust global exploration. The iterative update equation for the MSA is given as:

$$y_j(t+1) = \phi_{t\gamma} y_j(t) + g_1 \times \sin(g_2) \times [g_3 q_j(t) - x_j(t)] \quad (19)$$

where g_1 is:

$$g_1 = \frac{\varphi_{max} - \varphi_{min}}{2} \cos\left(\frac{\pi t}{T_m}\right) + \frac{\varphi_{max} + \varphi_{min}}{2} \quad (20)$$

Where φ_{min} and φ_{max} represent the minimum and maximum values of φ_t, respectively, and φ_t, is defined as:

$$\varphi_t = \varphi_{max} - (\varphi_{max} - \varphi_{min}) \times \frac{t}{T_m} \quad (21)$$

The MSADBO algorithm incorporates the improved location update scheme as:

$$y_j(t+1) = \begin{cases} y_j(t) + c \times \Delta y + \mu \times k \times y_j(t-1), \\ \gamma < n \\ \phi_t y_j(t) + g_1 \times \sin(g_2) \times [g_3 q_j(t) - x_j(t)], \\ \gamma \ge n \end{cases} \quad (22)$$

where $\gamma = rand(1)$, and n is a threshold.

MSADBO utilizes an adaptive Gaussian-Cauchy perturbation approach to broaden the search space. The perturbation strategy is defined as:

$$U^b(t) = Y^b(t) + (1 + \tau_1 \times G(\sigma) + \tau_2 \times Ca(\sigma)) \quad (23)$$

Here, $G(\sigma)$ and $Ca(\sigma)$ denote Gaussian and Cauchy operators. The greedy rule for updating the population is:

$$Y^b = \begin{cases} U^b(t), & f(U^b(t)) < f(Y^b(t)) \\ Y^b(t), & otherwise \end{cases} \quad (24)$$

E. MSADBO-LSTM Model Process

The flowchart in Fig. 1 outlines the complete MSADBO-LSTM model process. Based on this flowchart, the MSADBO-LSTM model begins by defining the objective function and setting essential parameters. Next, it initializes the dung beetles' positions using a chaotic mapping technique based on Bernoulli distribution. The algorithm checks the convergence condition (δ) and, if converged, adjusts the rolling beetle's position based on the MSA equation. Otherwise, their positions are updated by using the DBO algorithm. In the next step, the positions of breeding beetles, small beetles, and stealing beetles are adjusted. The algorithm utilizes a Gaussian-Cauchy perturbation technique to adjust the current optimal solution and explore new potential candidates. The next step is identifying the optimal solutions by recomputing the fitness of each beetle. Then, the algorithm applies a Gaussian-Cauchy perturbation approach to perturb the current optimal solution and find new candidates. Next, a greedy algorithm is employed to decide whether the newly perturbed positions should replace the old ones. The final hyperparameters of the LSTM network are optimized and presented at the conclusion of the iterative process, which continues until reaching the predefined maximum number of iterations.

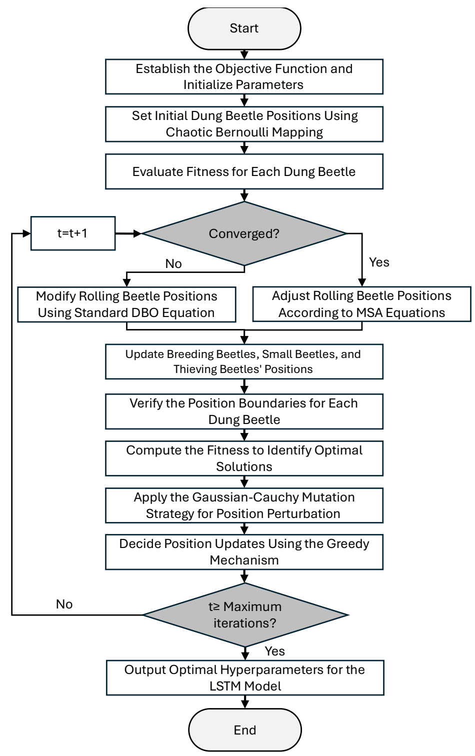

Fig. 1. Flowchart of the MSADBO-LSTM Model

F. Performance Evaluation Metrics

Numerous metrics have been introduced to evaluate the effectiveness of forecasting models. An analysis of key research highlights that the Mean Absolute Percentage Error (MAPE) and Mean Absolute Error (MAE) are among the most widely utilized indicators [24], [25]. Additionally, to compare our results with those in [26], we employed the Pearson correlation coefficient (R) as an evaluation metric. Equations (25), (26) and (27) define MAE, R, and MAPE, respectively [6].

$$MAE = \frac{1}{n} \sum_{j=1}^{n} |x_j - y_j| \quad (25)$$

$$R = \frac{\sum_{j=1}^{n}(x_j - \bar{x})(y_j - \bar{y})}{\sqrt{\sum_{j=1}^{n}(x_j - \bar{x})^2}\sqrt{\sum_{j=1}^{n}(y_j - \bar{y})^2}} \tag{26}$$

$$MAPE = \frac{1}{n}\sum_{j=1}^{n}\left|\frac{x_j - y_j}{x_j}\right| \tag{27}$$

where x_j indicates the actual value, y_j refers to the predicted value, $\bar{x}$ represents the average of the actual values, $\bar{y}$denotes the average of the predicted values, and n signifies the total number of records.

III. INPUT DATA AND SETTINGS

A short overview of the Italian day-ahead market is provided in this section. Comprehensive details about the dataset are available in reference [26]. To ensure consistency with prior research [26], this study uses the same dataset from the same period (2017–2020) for performance comparison.

The Italian day-ahead market (MGP) enables participants to submit bids and offers within a timeframe starting from the ninth day prior to the day of physical delivery until the day before delivery [27].

The National Single Price (PUN) in the MGP is determined as the weighted average of zonal prices, considering all transactions except those related to pumped-storage units and imports from neighboring countries' zones [27], [28]. Numerous factors affect the wholesale electricity price. For this analysis, relevant public data was selected based on a comprehensive literature review [29], [30]. The variables used for predicting PUN are outlined in TABLE 1. The dataset includes data on electricity consumption, electricity prices, and natural gas (NG) prices, sourced from *Gestore dei Mercati Energetici* (GME) for the years 2017, 2018, 2019, and 2020 [27] on an hourly basis, except NG prices, which are recorded daily. Except for 2020, which had an extra day as a leap year and totaled 8,784 hours, each year has 8,760 hours. In total, 35,064 hourly observations were recorded during the study period (2017–2020). Peak hours refer to 8 am to 8 pm on weekdays, while off-peak hours include weekends and weekday hours outside this range [27].

TABLE 1: FEATURES AND CORRESPONDING UNITS FOR PUN FORECASTING

Features	Units
Time-related features	
Hour of the Day	Values ranging from 1 to 24
Hour Category (Peak/Off-Peak)	0(Peak), 1(Off-Peak)
Day of the Week	1 (Sunday) to 7 (Saturday)
Day Category	0(weekday) 1(weekend)
Season Indicator	1 (Winter) to 4 (Fall)
Load Consumption	
Predicted Load Consumption	MW
Load at Same Hour on Prior Day	MW
Load at Same Hour One Week Prior	MW
Average Load of Previous 24 Hours	MW
Day-Ahead Electricity Price	
Price at Same Hour on Prior Day	€/MWh
Price at Same Hour One Week Prior	€/MWh
Average Price of Previous 24 Hours	€/MWh
Natural Gas (NG) Price	
NG Price on Previous Day	€/MWh
Average NG Price Previous Week	€/MWh

TABLE 2 provides an overview of the statistical summary for the annual load and price over the four-year period. The table reveals a downward trend in load after 2018, which corresponds to a declining trend in PUN.

The dataset was divided into training, validation, and testing subsets using the holdout strategy. Specifically, 75% of the data was allocated to training and validation, while the remaining 25% was set aside for testing (Fig. 2). The period from January 1, 2017, to December 31, 2019, was used for training and validation, and the entire year of 2020 was designated as the testing set [31].

TABLE 2: STATISTICAL SUMMARY FOR THE ANNUAL LOAD CONSUMPTION AND PRICE (2017-2020) [27], [32]

Year		2017	2018	2019	2020
Load (MW)	Max	50333	52412	53824	49964
	Min	17212	17610	17014	15334
	Mean	33271	33934	33579	31102
	Std	7568	7555	7731	7449
PUN (€/MWh)	Max	170	159.4	108.4	162.6
	Min	10	6.97	1	0
	Mean	53.95	61.31	52.32	38.91
	Std	16.46	14.84	12.68	14.65

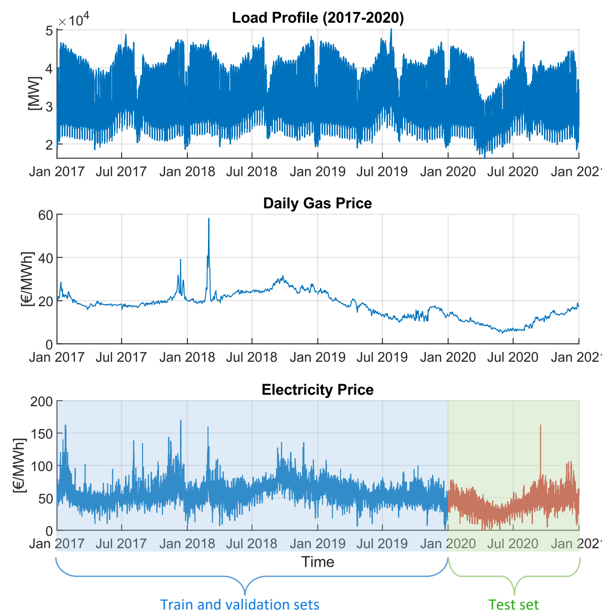

Fig. 2. Electricity consumption, PUN and Natural gas price, 2017-2020

IV. DISCUSSION ON THE RESULTS

This study investigates the effects of optimization algorithms, including GWO, DBO, and MSADBO, on the performance of the LSTM model in predicting the wholesale electricity market price in Italy. Furthermore, the prediction accuracy of the LSTM, as a deep learning approach, is compared with other machine learning models such as SVM-Linear, Multilayer Perceptron (MLP), Gaussian Process Regression (GPR), and Bagged Trees, as reported in [26].

TABLE 3 presents a comparative analysis of predictive models based on metrics such as MAE, R, and MAPE. The value of each index in this table represents the average of the calculated metrics in the test set period (8784 hours/year 2020).

As seen in this table, the LSTM model without optimizing the hyperparameters has an MAE of 4.085 and a MAPE of 16.095%, with a correlation coefficient (R) of 0.937. The performance of the LSTM model is significantly enhanced after optimizing the LSTM hyperparameter with DBO optimization, i.e., the MAE decreases to 3.863, and MAPE decreases to 14.467% while slightly increasing R to 0.939. The efficiency of the LSTM model is further improved by the GWO algorithm, which reduces the MAE to 3.731 and the MAPE to 14.835%.

The best results across different methods are highlighted in green, indicating that LSTM-MSADBO achieves the lowest MAE (3.699) and MAPE (14.074%), demonstrating superior accuracy in electricity price prediction. Additionally, LSTM-MSADBO and SVM-Linear attain the highest correlation coefficient (R = 0.941), signifying strong alignment between the forecasted and observed values.

TABLE 3: COMPARISON OF THE VARIOUS PREDICTIVE MODELS

	Current study			
	LSTM	LSTM DBO	LSTM GWO	LSTM MSADBO
MAE	4.085	3.863	3.731	**3.699**
R	0.937	0.939	0.939	**0.941**
MAPE (%)	16.095	14.467	14.835	**14.074**
	Reference [26]			
	SVM Linear	MLP {5 5}	GPR M.5/2	Bagged Tress 40
MAE	3.769	3.873	4.305	5.255
R	**0.941**	0.931	0.934	0.906
MAPE (%)	15.833	16.308	15.979	24.947

The LSTM-MSADBO model was retrained using the best set of hyperparameters, achieving a learning rate of approximately 0.0011, with 10 neurons in the hidden layer and an L2 regularization value of 1e-06 (TABLE 4).

TABLE 4: BEST SET OF HYPERPARAMETERS, LSTM-MSADBO

learning rate	Number of Neurons	L2 Regularization	Batch size
0.0011	10	1e-06	8

Fig. 3 compares electricity price predictions from various models against actual prices on December 20th, 2020 (as a sample day). The current study's methods are compared with the overall best prediction method from reference [26], SVM-Linear. Remarkably, the LSTM-MSADBO model, red curve, closely follows the electricity price trend and appropriately predicts the actual price in the peak hours and captures corresponding volatilities. In this figure, the period in which the LSTM-MSADBO model outperforms other models is magnified for better visualization.

Fig. 4 uses separated box plots to highlight the MAE index distribution across several forecasting models, namely SVM, Bagged Tree (B-Trees), GPR, MLP, LSTM, and LSTM optimized by DBO, GWO, and MSADBO. The MSADBO-LSTM model outperforms the other models, evidenced by its lowest average MAE value of 3.699. While traditional approaches such as SVM, B-Trees, and GPR exhibit higher error distributions, deep learning methods like LSTM with DBO and GWO optimizations show relatively lower MAE values. As seen in this figure, the performance of the LSTM model improves gradually by applying optimization algorithms and significantly reducing MAE.

Fig. 5 compares the monthly MAE between two forecasting models: SVM and LSTM-MSADBO. SVM model monthly performance shows noticeable variability in error rates across the year, with higher MAE values, particularly in September, October, and November. In contrast, the LSTM-MSADBO model's consistent and lower error distribution across all months indicates improved predictive stability.

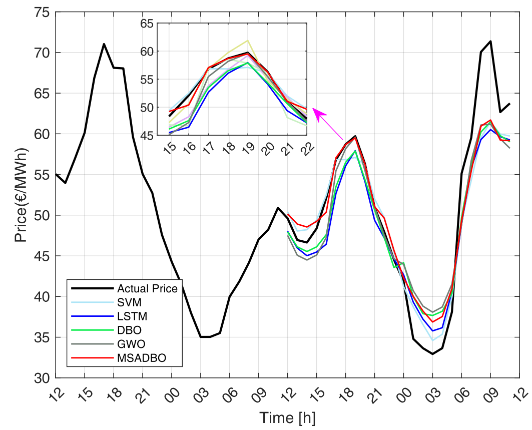

Fig. 3. Forecasts of different methods for 20-Dec-2020

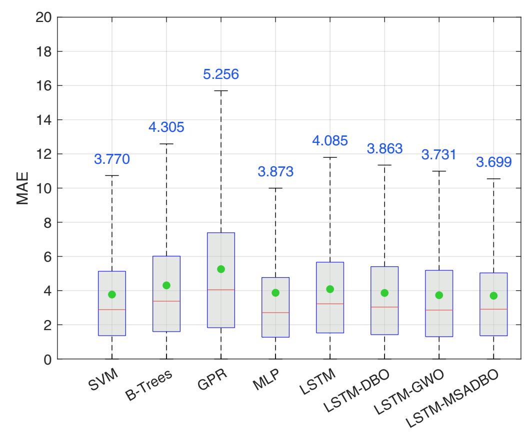

Fig. 4. Comparison of MAE distributions for different forecasting models

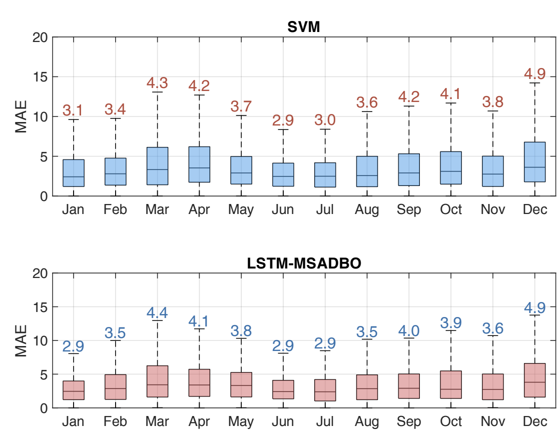

Fig. 5. Monthly MAE comparison between SVM and LSTM-MSADBO

V. CONCLUSION

The MSADBO optimization effectively enhances the forecasting accuracy of LSTM neural networks for wholesale electricity price predictions in the Italian market. The results demonstrate that advanced optimization algorithms significantly improve LSTM performance, with progressive gains achieved through DBO, GWO, and ultimately, MSADBO. Notably, the LSTM-MSADBO model consistently outperforms traditional and machine learning-based approaches, achieving the lowest MAE and MAPE, along with a high correlation coefficient (R).

Moreover, the model's ability to closely track actual price trends, capture market volatility, and maintain low error variability across different timeframes highlights its robustness and reliability. These findings emphasize the potential of the LSTM-MSADBO approach as a valuable tool for market participants, enabling them to make informed, profit-maximizing decisions based on accurate electricity price forecasts.

REFERENCES

[1] A. J. Conejo, M. A. Plazas, R. Espinola, and A. B. Molina, "Day-ahead electricity price forecasting using the wavelet transform and ARIMA models," *IEEE Trans. power Syst.*, vol. 20, no. 2, pp. 1035–1042, 2005.

[2] Z. Tan, J. Zhang, J. Wang, and J. Xu, "Day-ahead electricity price forecasting using wavelet transform combined with ARIMA and GARCH models," *Appl. Energy*, vol. 87, no. 11, pp. 3606–3610, 2010.

[3] M. Hosseini Imani, E. Bompard, P. Colella, and T. Huang, "Impact of Wind and Solar Generation on the Italian Zonal Electricity Price," *Energies* , vol. 14, no. 18. 2021, doi: 10.3390/en14185858.

[4] P.-H. Kuo and C.-J. Huang, "A high precision artificial neural networks model for short-term energy load forecasting," *Energies*, vol. 11, no. 1, p. 213, 2018.

[5] B. Neupane, W. L. Woon, and Z. Aung, "Ensemble prediction model with expert selection for electricity price forecasting," *Energies*, vol. 10, no. 1, p. 77, 2017.

[6] M. H. Imani, E. Bompard, P. Colella, and T. Huang, "Predictive methods of electricity price: an application to the Italian electricity market," in *2020 IEEE International Conference on Environment and Electrical Engineering and 2020 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)*, 2020, pp. 1–6.

[7] M. Rafiei, T. Niknam, and M. H. Khooban, "A novel intelligent strategy for probabilistic electricity price forecasting: Wavelet neural network based modified dolphin optimization algorithm," *J. Intell. Fuzzy Syst.*, vol. 31, no. 1, pp. 301–312, 2016.

[8] M. Sundermeyer, R. Schlüter, and H. Ney, "Lstm neural networks for language modeling.," in *Interspeech*, 2012, vol. 2012, pp. 194–197.

[9] Q. Chen, X. Zhu, Z. Ling, S. Wei, H. Jiang, and D. Inkpen, "Enhanced LSTM for natural language inference," *arXiv Prepr. arXiv1609.06038*, 2016.

[10] G. Parascandolo, H. Huttunen, and T. Virtanen, "Recurrent neural networks for polyphonic sound event detection in real life recordings," in *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 2016, pp. 6440–6444.

[11] K. Chen, Y. Zhou, and F. Dai, "A LSTM-based method for stock returns prediction: A case study of China stock market," in *2015 IEEE international conference on big data (big data)*, 2015, pp. 2823–2824.

[12] S. Hochreiter, "Long Short-term Memory," *Neural Comput. MIT-Press*, 1997.

[13] S. Anbazhagan and N. Kumarappan, "Day-ahead deregulated electricity market price forecasting using recurrent neural network," *IEEE Syst. J.*, vol. 7, no. 4, pp. 866–872, 2012.

[14] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 841–851, 2017.

[15] L. Han, Y. Peng, Y. Li, B. Yong, Q. Zhou, and L. Shu, "Enhanced deep networks for short-term and medium-term load forecasting," *Ieee Access*, vol. 7, pp. 4045–4055, 2018.

[16] U. Ugurlu, O. Tas, A. Kaya, and I. Oksuz, "The financial effect of the electricity price forecasts' inaccuracy on a hydro-based generation company," *Energies*, vol. 11, no. 8, p. 2093, 2018.

[17] S. Zhou, L. Zhou, M. Mao, H.-M. Tai, and Y. Wan, "An optimized heterogeneous structure LSTM network for electricity price forecasting," *IEEE Access*, vol. 7, pp. 108161–108173, 2019.

[18] Z. Chang, Y. Zhang, and W. Chen, "Electricity price prediction based on hybrid model of adam optimized LSTM neural network and wavelet transform," *Energy*, vol. 187, p. 115804, 2019.

[19] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014.

[20] Y. Xu, X. Huang, X. Zheng, Z. Zeng, and T. Jin, "1VMD-ATT-LSTM Electricity Price Prediction Based on Grey Wolf Optimization Algorithm in Electricity Markets Considering Renewable Energy," *Renew. Energy*, p. 121408, 2024.

[21] J. Xue and B. Shen, "Dung beetle optimizer: A new meta-heuristic algorithm for global optimization," *J. Supercomput.*, vol. 79, no. 7, pp. 7305–7336, 2023.

[22] J. Huo and J. Huo, "Hybrid Energy Storage Power Adaptive Optimization Strategy Based on Improved Model Predictive Control and Improved DBO-VMD," *Energies*, vol. 17, no. 13, p. 3312, 2024.

[23] J. Pan, S. Li, P. Zhou, G. Yang, and D. Lyu, "Dung Beetle Optimization Algorithm Guided by Improved Sine Algorithm," *Comput. Eng. Appl*, vol. 59, pp. 92–110, 2023.

[24] M. H. Imani, E. Bompard, P. Colella, and T. Huang, "Data analytics in the electricity market: a systematic literature review," *Energy Syst.*, pp. 1–35, 2023.

[25] L. Peng, S. Liu, R. Liu, and L. Wang, "Effective long short-term memory with differential evolution algorithm for electricity price prediction," *Energy*, vol. 162, pp. 1301–1314, 2018.

[26] M. H. Imani, E. Bompard, P. Colella, and T. Huang, "Forecasting electricity price in different time horizons: an application to the Italian electricity market," *IEEE Trans. Ind. Appl.*, 2021.

[27] GME-, "Gestore dei mercati energetici," 2024. https://www.mercatoelettrico.org/EN/ (accessed Jul. 20, 2024).

[28] M. Hosseini Imani, "Empirical Analysis of Inter-Zonal Congestion in the Italian Electricity Market Using Multinomial Logistic Regression," *Energies*, vol. 17, no. 23, p. 5901, 2024.

[29] D. Keles, J. Scelle, F. Paraschiv, and W. Fichtner, "Extended forecast methods for day-ahead electricity spot prices applying artificial neural networks," *Appl. Energy*, vol. 162, pp. 218–230, 2016.

[30] F. Keynia, "A new feature selection algorithm and composite neural network for electricity price forecasting," *Eng. Appl. Artif. Intell.*, vol. 25, no. 8, pp. 1687–1697, 2012.

[31] J. Stuart and P. Norvig, "Artificial Intelligence: a Modern Approach Prentice-Hall," 2010.

[32] "ENTSO-E." https://www.entsoe.eu/ (accessed Jul. 22, 2024).