https://doi.org/10.31449/inf.v50i11.9715 Informatica **50** (2026) 131–142 **131**

Optimizing LSTM Hyperparameters with Fish Swarm Optimization for Enhanced Power Load Prediction in Wireless Networks

Shuai Gao [1*], Hu Yue[2], Yujun Wang[2], Hao Hu[1], Meiying Zhu[1]
[1]State Grid Jibei Electric Power Co., Ltd. Measurement Center, BeiJing, 100045, China
[2]State Grid Jibei Electric Power Co., Ltd, BeiJing, 100054, China
E-mail: 3838146@163.com(corresponding author), yue.hu@jibei.sgcc.com.cn, 1658419050@163.com, hh6006@foxmail.com, 18332178933@163.com

Keywords: Power load prediction, wireless network, LSTM, hyper parameters, FSO

Received: June 13, 2025

Predicting power loads is an important part of managing the electricity system and provides a basic guarantee for the dependability and financial operation of state grid companies. In order to minimise energy production costs, it is advantageous to have an accurate prediction for efficient energy scheduling, which involves balancing power generation and demand. Many scholars have devoted their time and energy to creating trustworthy load forecasting models in the hopes of achieving the highest possible prediction accuracy. Utilising a variety of ML, DL, AI, and hybrid approaches allows for accurate power load prediction. An integrated model for power load prediction using the Fish Swarm Optimisation (FSO) algorithm and the Long Short-Term Memory (LSTM) neural network is presented in this research. By optimising the LSTM network's hyper parameters using FSO, this study speeds up model convergence and avoids becoming stuck in local optima, which reduces the effect of humans picking LSTM hyper parameters at random on prediction results. The proposed approach is evaluated using mean absolute error (MAE), mean absolute percentage error (MAPE), root mean square error (RMSE) and R2 metrics and compared with LSTM, CNN, Attention-LSTM, AC-BiLSTM and LSTM-PSO approaches. The proposed LSTM-FSO significantly reduces MAE between 3.75% - 49.59%, MAPE between 9.98% - 39.65%, RMSE between 11.17% - 56.10% and increased R2 between 0.1% - 8.07% compared to other methods. The results of the experiments show that the suggested system has great potential for use and significantly beats competing solutions. Power load forecasting is shown to be more stable and reliable using the FSO algorithm with LSTM, according to the simulation findings.

Povzetek: Raziskava predstavlja izboljšan model za napoved porabe električne energije, ki z uporabo kombinacije LSTM nevronske mreže in algoritma Fish Swarm Optimization dosega bolj natančne, stabilne in zanesljive rezultate kot obstoječe metode.

1 Introduction

Power load denotes the aggregate electrical energy consumed by a user's electrical apparatus from the power supply at a given instant. Electric load can be categorized as industrial, agricultural, transportation, and residential use based on various entities. The industrial power load consumption primarily occurs in light industry, energy-intensive industries, advanced manufacturing, and the mining sector. Currently, China's industrial power consumption constitutes 69% of the overall electric demand, which continues to rise due to ongoing output expansion. Furthermore, the production and consumption of industrial power must occur concurrently due to the impracticality of storing substantial quantities of generated power [1].

Power load forecasting is predicting future load data utilizing past data as the primary element [2]. The efficacy of power load prediction serves as a significant indicator of the modernization trajectory of electric power enterprise management. Precise power load forecasting is crucial for the modernization and scientific management of the power system [3]. The accurate forecasting of demand factors, including metrics such as hourly load, peak load, and total energy consumption, is essential for the effective management and planning of power systems. The classification of power load prediction into three separate categories, as illustrated below, addresses various application needs [4]:

- Long-term prediction (LTP): Spanning 1 to 20 years, LTP is essential for integrating new-generation units into the system and developing transmission infrastructure.
- Medium-term prediction (MTP): Ranging from 1 week to 12 months, MTP is crucial for establishing tariffs, coordinating system

maintenance, managing finances, and aligning fuel supply.

- Short-term prediction (STP): Covering 1 hour to 1 week, STP is vital for scheduling the operation of generation units, preparing spinning reserves, analyzing transmission system constraints, and assessing power system security.

The power load forecast is intricately linked to the dispatch and routine operations of national electricity consumption, influencing both the livelihoods of inhabitants and the overall functioning of the country [5]. The primary research endeavor centers on short-term power load forecasting, which are critical areas enabling universities and energy firms to dynamically modify their power generation and trading strategies within the market context [6]. This article mostly examines short-term power load forecasts. Numerous studies have been conducted to identify a suitable and effective model for load forecasting. The conventional linear statistical methods (autoregressive integrated moving average) [7] were examined for time series forecasting because of their assumptions of data being stationary, linear, and adhering to specific statistical distributions.

Artificial Intelligence (AI) is a fundamental component in numerous domains, with key subdivisions including machine learning and swarm intelligence. Machine learning (ML) methodologies are prompting the investigation of enhancements to forecasting systems, aiming to incorporate the most efficient and robust techniques to reduce errors [8]. Swarm intelligence was conceived by emulating the swarming behavior of many natural systems, and it is employed to address numerous optimization challenges [9]. Dai et al. [10] propose the usage of SVM for power load prediction and employ particle swarm optimization to optimize the parameters of the SVM. Guo et al. [11] suggests different machine learning models for predicting the power load. These algorithms enhance accuracy; however they depend significantly on the quality of the training data [12]. Currently, it is widely recognized that deep neural networks (DNN) have excelled in load prediction in recent years. Diverse deep learning models, such as RNN, LSTM, and GRU, are employed to forecast electricity load [13].

Individual techniques for load forecasting frequently exhibit various shortcomings, including inefficiency in calculation, complexity in computation, and elevated mistake rates. Researchers have been developing hybrid load forecasting methods and models throughout the years to achieve enhanced accuracy with minimal error rates [14]. Different hybrid models such as, MLR-LSTM [15], prophet-LSTM [16] and PSO LSTM-AE [17] are suggested for power load prediction. All models have produced optimal results; nonetheless, the error rate remains excessively high for effective forecasting.

This research presents a power load forecasting model that combines the Fish Swarm Optimization (FSO) algorithm with the Long Short-Term Memory (LSTM) neural network. This study utilizes FSO to optimize the hyper parameters of the LSTM network, markedly enhancing model convergence and averting entrapment in local optima, thus alleviating the influence of arbitrary human selection of LSTM hyper parameters on predictive results. This study demonstrates the clear benefits and enhanced predictive capabilities of the proposed FSO-LSTM methodology, which could transform energy load forecasting and improve taking decisions in power systems and market management.

The hypotheses of this research work are:

- An LSTM with hyperparameters optimized by FSO will get a reduced RMSE in power-load forecasting compared to the identical LSTM utilizing default or randomly chosen hyperparameters.
- LSTM+FSO models exhibit greater robustness compared to baseline LSTM models.
- FSO will determine hyperparameters that optimize the equilibrium between model complexity and error.
- FSO will generate hyperparameters that achieve statistically comparable or superior test RMSE compared to those derived from alternative metaheuristic methods.

The subsequent sections of this work are structured as follows: Section 2 examines the pertinent literature, Section3 provides the research background includes LSTM and FSO algorithm. Section 4 delineates the suggested power load prediction, Section 5 outlines the experimental methodology and result analysis, and Section 6 provides the conclusion.

2 Related works

Researchers have examined many models to develop an accurate power load forecasting model. The load characteristics and capacity of micro-grids determine the precision of the forecasting models. Load forecasts are classified into three temporal categories: short-term [18], encompassing predictions from hours to weeks ahead; medium-term, involving forecasts spanning months; and long-term, which pertains to predictions extending over a year [19]. This section explains the different power load forecasting.

Dong et al. [20] introduced a short-term power load forecasting approach that integrates K-means clustering with Support Vector Machines (SVM). This method encompasses data preparation, selection of analogous days, training of the SVM prediction model, and parameter optimization. This approach improves 39% accuracy compared to conventional methods.

Li et al. [21] developed a CEEMDAN-SE-LSTM model to predict ultra-short-term electricity demand in Changsha, China, incorporating meteorological and holiday variables. The long short-term memory neural network model was employed to forecast and overlay the rebuilt component series to get the final prediction outcomes. It significantly enhances the precision of ultra-short-term power load forecasting, facilitates ultra-short-term power dispatching in Changsha, and serves as a reference for other cities in developing short-term and ultra-short-term power load forecasting models.

Veeramsetty et al. [22] create an effective machine learning model employing gated recurrent units (GRU) and random forest (RF) to predict electric power load. GRU has been utilized to forecast electric power demand, while RF has been applied to diminish the model's input dimensions.

Wan et al. [23] offer a novel methodology for short-term power load forecasting that integrates convolutional neural networks (CNN), long short-term memory, and attention mechanisms. It improves the precision of short-term power load forecasting. A one-dimensional CNN layer is employed to extract high-dimensional features from the input data, succeeded by an LSTM layer that captures temporal correlations within historical episodes. An attention technique is implemented to refine the weight of the LSTM output, augment the significance of critical information, and improve the overall predictive model.

Veeramsetty et al. [24] create a machine learning model utilizing long short-term memory and factor analysis to forecast the load at a particular time on an electrical power substation. A novel long short-term memory architecture incorporating factor analysis is being designed based on the methodology employed for simulating substation load predictions in Microsoft Azure Notebooks.

Xiang et al. [25] propose a medium- and long-term power load forecasting approach utilizing a two-layer categorical boosting technique that incorporates multi-dimensional features. Additionally, a randomized search cross-validation regression model is employed for the optimization of model parameters. The training and test sets are derived from real data in a province in northeast China. This method demonstrates significant potential for medium- and long-term power load forecasting applications.

Ullah et al. [26] propose an intelligent deep learning-based method for power load prediction. Initially, data collected from household meters undergoes a pre-assessment step. Subsequently, the refined data sequence is input into a modified convolutional long short-term memory network, which gathers spatiotemporal correlations and generates feature maps. These feature maps are then forwarded to a deep gated recurrent unit (GRU) network for learning, ultimately yielding the final prediction.

Xu et al. [27] propose the development of short-term and mid-term power system load forecasting models utilizing hybrid deep learning techniques. In the data preprocessing phase, the exponential weighted moving average method addresses missing values. The method employed for outlier detection is the Generalized ESD Test ADSD. This study examines the past load data of a regional power grid and four industries, introducing a short-term power system load forecasting model grounded in Bi-directional Long Short-Term Memory (BiLSTM). For mid-term load forecasting, feature selection is initially conducted using random forest and the Pearson correlation coefficient. Subsequently, a hybrid deep learning model is formulated, integrating BiLSTM and random forest.

Dai et al. [28] propose an innovative short-term power load forecasting method that enhances the bidirectional long short-term memory model through the integration of Extreme Gradient Boosting (XGBoost) and an Attention mechanism. The weighted grey relational projection algorithm is employed to differentiate between holidays and non-holidays during data preprocessing. The Attention mechanism is incorporated into the Bi-LSTM model to augment the validity and accuracy of predictions. XGBoost, a recently developed and high-performing predictive model, is utilized alongside the Attention mechanism to optimize the Bi-LSTM model.

Liu et al. [29] suggest a new ultra-short-term power load forecasting model that integrates Convolutional Neural Networks, Bidirectional Long Short-Term Memory networks, and an Attention mechanism. This novel method leverages CNN and BiLSTM to obtain spatio-temporal features from load information, while the Attention system assigns optimal weights to the hidden states of the BiLSTM model, thereby enhancing critical historical load sequence data and reducing information loss. The model's final output is established via a fully connected layer.

3 Research background

This section provides a succinct yet thorough explanation of the theoretical principles underpinning LSTM and FSO to augment the reader's comprehension.

a. LSTM

The LSTMs were introduced by Hochreiter and Schmidhuber [30], evolving from RNNs by incorporating new modules to address the challenges associated with long-range dependencies and the retention of information over prolonged durations. The LSTM approach features a chain structure with a repeating module with an alternative configuration. In contrast to conventional RNNs, LSTMs are specifically designed to address the issue of long-term dependencies, which is an inherent aspect of their operation. LSTMs are composed of a series of recurrent modules, a characteristic they share with all RNNs. However, it is the arrangement of these recurring modules that distinguishes LSTMs. Unlike a single layer, LSTMs comprise four interrelated layers. The essential difference in LSTMs is the integration of a cell state—a horizontal pathway that facilitates uninterrupted information flow between the modules. The data transmission within the cell state is regulated by gates, which consist of a neural network layer utilizing the sigmoid function linked with a pointwise multiplication operation. The sigmoid layer produces values between 0 and 1, determining the degree of information transmission. Figure 1 clearly illustrates the essential architecture of the LSTM model.

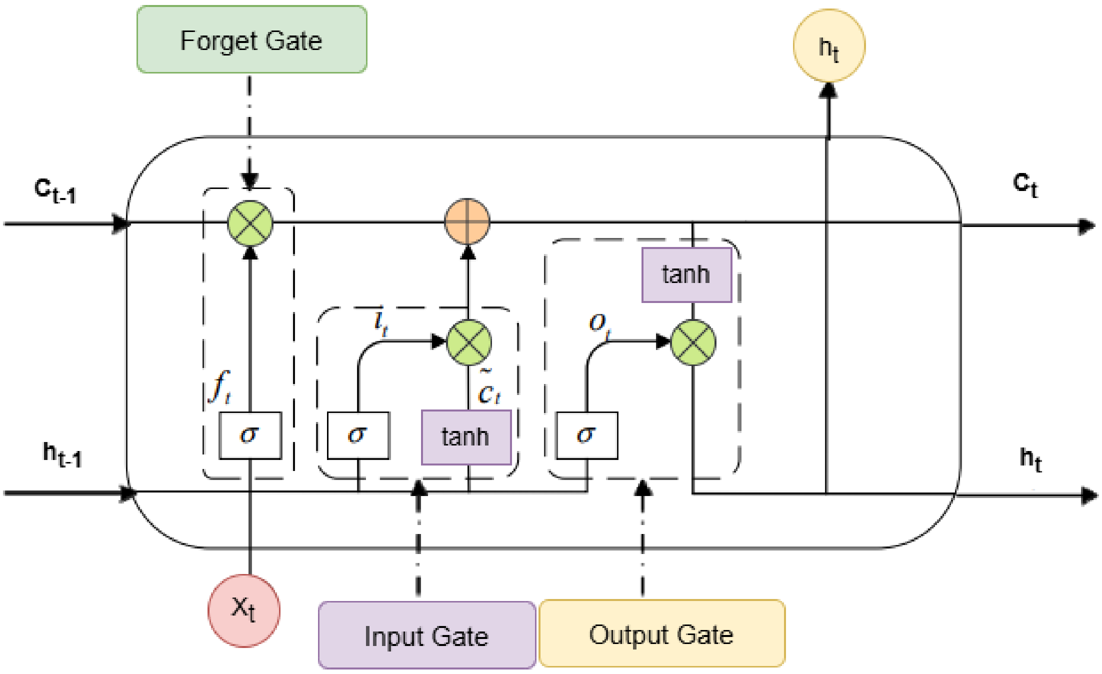

Figure 1: LSTM structure

LSTM networks utilize three specific gates to manage cell state: the forget gate, the input gate, and the output gate. The forget gate utilizes a sigmoid layer to ascertain which information elements should be discarded from the current cell state. The input gate consists of two essential components: a sigmoid layer that governs the updates to be implemented, and a tangent hyperbolic (tanh) layer that produces new potential values. The newly acquired information is integrated with the current cell state to provide an updated state. The output gate utilizes a sigmoid layer to identify the essential portions of the cell state that contribute to the final output. The processed cell state is then subjected to a tanh activation function and multiplied by the output derived from the sigmoid gate. This integrated procedure ultimately yields the final output [4].

$$f_t = \delta\left(\omega_f[h_{t-1}, x_t] + b_f\right) \quad (1)$$

$$i_t = \delta(\omega_i[h_{t-1}, x_t] + b_i) \quad (2)$$

$$o_t = \delta(\omega_o[h_{t-1}, x_t] + b_o) \quad (3)$$

$$\tilde{c}_t = tanh(\omega_c[h_{t-1}, x_t] + b_c) \quad (4)$$

$$c_t = f_t \times c_{t-1} + i_t \times \tilde{c}_t \quad (5)$$

$$h_t = o_t \times \tanh(c_t) \quad (6)$$

f_t, i_t, and o_t represent the forget, input, and output gate, respectively; ω represents the weight, the notation $[h_{t-1}, x_t]$ signifies the concatenation of the input measure and the hidden layer dimension from the preceding layer, and b indicates the bias term; δ is the nonlinear activation function sigmoid, while $\omega_f, \omega_i, \omega_o$ b_f, b_i, b_o, and b_c are the parameters that the model must learn.

b. FSO

Fish Swarm Optimization (FSO) is an innovative bionic algorithm that emulates the social behavior of fish in their natural environment, initially introduced by Qian et al. [31]. It is a metaheuristic algorithm designed for addressing optimization problems. The program employs the behaviors of fish swarms, encompassing predation, aggregation, and pursuit. Figure 2 illustrates the conceptual vision of artificial fish [32].

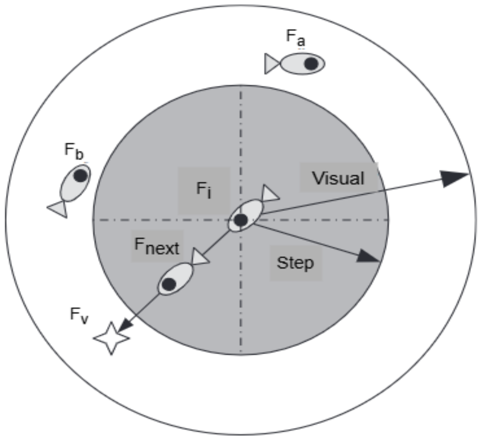

Figure 2: Conceptual vision of artificial fish

Let F_i denote the present location of an artificial fish, and F_v signify the viewpoint of the artificial fish at a specific moment. The vision range of each individual is depicted, with F_a and F_b denoting fish within the visual scope of F_i. Step indicates the maximum movement of the fake fish, while σ signifies the congestion factor of the fish swarm. The concentration of food is directly proportional to the fitness function fit(F). The behavioral tendencies demonstrated by fish swarms can be articulated as follows:

Swarming behavior is activated when fit(F_c) exceeds fit(F_i), with F_c denoting the central location inside the visual range of position F_i. Let F_c be represented as F_v. The fish at F_i will approach the position at F_c by taking a step.

Chasing behavior transpires when the objective function value at point F_{max}, the optimal point in the Visual, exceeds the objective function value at point F_i, provided that the Visual of F_i is not congested. The chasing action is performed in this instance. Let F_{max} be represented as F_v. The fish at F_i will approach the point F_{max}.

Preying behavior is evident in the following circumstances: when fit(F_c) < fit(F_i), fit(F_{max}) < fit(F_i), and the Visual is not congested, and when the Visual is congested.

This algorithm arbitrarily finds a point F_j within the visual proximity of point F_i. The program performs the predatory behavior if the objective function value at F_j surpasses that at F_i. The fish at F_i subsequently advances to F_j, adopting F_j as its new location. If the objective function value at F_j does not exceed that at F_i, the fish at F_i travels randomly within its visual range. Each repetition designates the optimal option as a "board." Upon reaching a predetermined number of iterations, the search process concludes, and the solution on the "board" is deemed the final result. The position update for artificial predatory fish can be articulated like follows:

$$F_{next} = F_i + rand \times \frac{step \times \left(F_j - F_i\right)}{norm\left(F_j - F_i\right)} \quad (7)$$

F_{next} denotes the subsequent position of the artificial fish; F_i signifies the present location of the artificial fish; F_j indicates the position with a superior objective function value. rand is a stochastic variable inside the interval of -1 to 1, and norm(F_j - F_i) denotes the distance between the two positional vectors.

The position updating for artificial swarming fish can be articulated as follows:

$$F_{next} = F_i + rand \times \frac{step \times (F_c - F_i)}{norm(F_c - F_i)} \quad (8)$$

The position update for artificial fish pursuit might be stated like follows:

$$F_{next} = F_i + rand \times \frac{step \times (F_{max} - F_i)}{norm(F_{max} - F_i)} \quad (9)$$

4 Proposed power load forecasting

This section explains the proposed power load forecasting. It includes data collection, data pre-processing and LSTM-FSO based forecasting. Figure 3 shows the general architecture of proposed work.

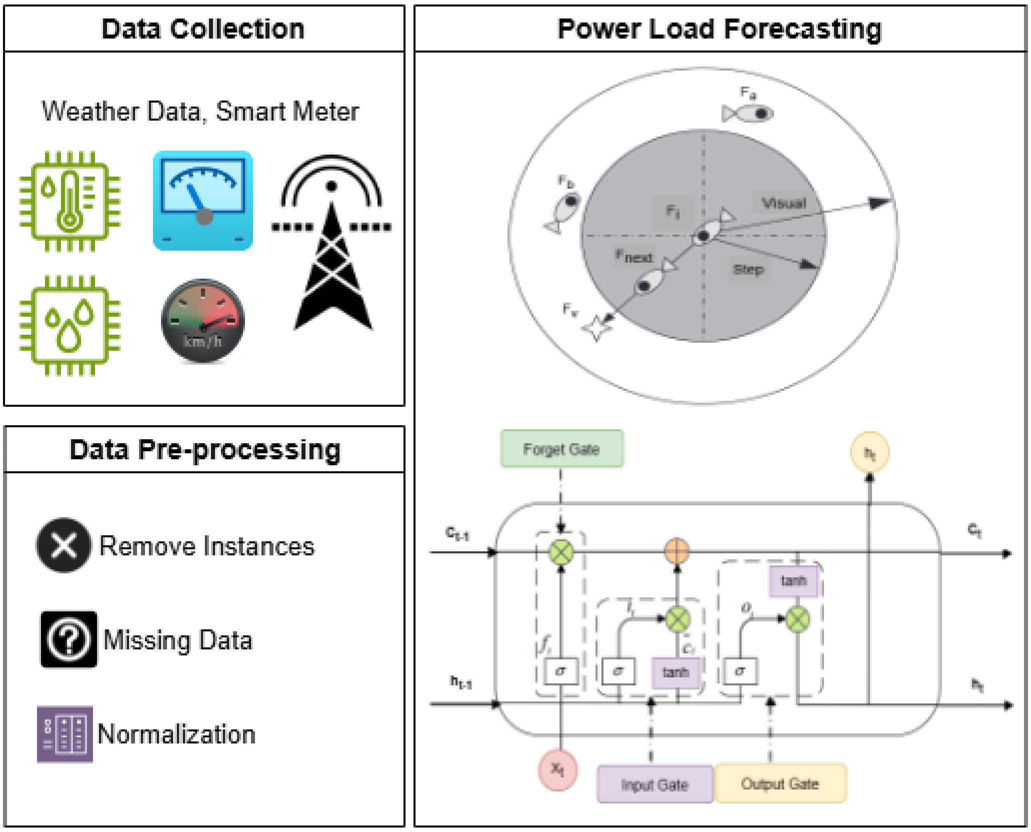

Figure 3: General architecture

The raw sequential data must first be cleansed and missing values addressed by interpolation (e.g., linear interpolation with a predetermined window size), subsequently followed by normalization techniques like Min-Max scaling to transform values into the [0,1] range prior to model training. The hyper-parameter (Learning rate, epoch, batchsize, hidden layer) for the LSTM is most effectively determined using the FSO technique.

a. Power load data collection and pre-processing

Generally, data collection involves the integration of wires throughout the building floors into a single edge connected to the main board, where a meter equipped with several sensors is installed to monitor and calculate the energy consumption throughout the building. The data is often gathered at minute resolutions.

All data sources utilized for this research are publically accessible, with hourly information obtainable from January 2018 to June 2020 [33]. The composition of the data is as follows:

Historical power load is accessible in daily post-dispatch reports, whereas historical weekly projections can be found in weekly pre-dispatch reports.

Climatic factors, including temperature, humidity levels, precipitation, and wind velocity, among others.

Table 1 shows the description of the dataset.

Table 1: Dataset description

Variable	Description	Unit of Measurement
National Load	National electricity load, excluding exports	MWh
Holiday	Holiday binary indicator	-
School	School period binary indicator	-
Temperature	Air temperature	ºC
Humidity	Specific Humidity	%
Wind	Wind speed	m/s
Precipitation	Total perceptible liquid water	l/m^2
Load Forecast	Historical national load forecast	MWh

The data gathered from sensors and meters is often significantly influenced by climatic conditions, tenant behavior, redundancy, wire breaks, or short circuits, resulting in irregularities, outliers, and noise in the variable values. Addressing this issue is essential for precise forecasting; so, enhance the data before forecasting.

During data collection, there are instances when we may not obtain the values between two data points. Consequently, the absent data are interpolated over these individual missing values. Interpolation is the method of estimating missing data by utilizing the preceding and succeeding values surrounding the gaps [34]. The subsequent formula is employed for interpolation.

$$m_i = m_{i+1} + \frac{m_{i-1} - m_{i+1}}{t_{i-1} - t_{i+1}} \quad (10)$$

Where m_i represents the missing value, m_{i-1} denotes the value preceding m_i, and m_{i+1} signifies the value succeeding m_i. t_i, t_{i-1}, and t_{i+1} represent the timestamps corresponding to m_i, m_{i-1}, and m_{i+1}, respectively.

Data are normalized to mitigate the predominance of large features and enhance convergence; if smart meters exhibit analogous patterns with varying magnitudes, this individual normalization will render these load profiles more comparable and ease training. The dataset is not similar due to its diverse units and orders of magnitude; it has been standardized and normalized in preparation.

$$\tilde{z} = \frac{x_t - \mu}{\sigma} \quad (11)$$

$$\tilde{x} = \frac{x_t - \min(x)}{\max(x)} \quad (12)$$

Where x_t is the original value. μ, σ, min(x), max (x) indicates mean, standard deviation, minimum and maximum value. $\tilde{z}$ and $\tilde{x}$ represents the standardized and normalized value.

b. LSTM-FSO power load forecasting

Upon completion of data preprocessing, it proceeds to the training phase. The LSTM model is employed to predict the power load in this phase. The LSTM possesses numerous factors that must be meticulously configured to achieve an accurate model. The essential hyperparameters are the number of epochs, batch size, time steps, and the quantity of neurons in the LSTM layers. Additionally, a range of activation and optimization functions must be chosen to yield improved outcomes. Manually configuring these parameters is tedious particularly for the time intervals where this parameter is crucial for prediction. This work utilize the Fish Swarm Optimization technique to determine the optimal time steps and the number of hidden units, or neurons, in the layers. Metaheuristic algorithms are recognized for not providing global optimum results; yet, they are proficient in identifying near-optimal solutions. This yields multiple LSTM configurations derived from FSO. The optimal model configuration is determined based on the RMSE values of the fitness function for FSO for each configuration. Algorithm-1 explains the proposed LSTM-FOS approach.

Algorithm-1 LSTM-FSO Approach

Input: Power Load Dataset PLD
Output: Best Hyper-parameter for LSTM, Load Prediction Result
Step1: Initialize parameters for FSO (population size, visual, step, crowd_factor, maxIter)
Step2: Randomly generate initial population FP (Hyper parameter for LSTM: Learning rate, epoch, batchsize, hidden units)
Step3: For each population FP[i]
Step4: Train LSTM with hyper parameters FP[i] on PLD
Step5: Compute fitness[i] based on RMSE
Step6: End
Step7: bestSoln = min(fitness[i])
Step8: For t = 1 to maxIter
Step9: For each population FP[i]
Step10: Select fish behavior
Step11: If behavior = Prey: Update FP[i] using eq. (13)
Step12: If behavior = Swarm: Update FP[i] using eq. (14)
Step13: If behavior = Chase: Update FP[i] using eq. (15)
Step14: Update bestSoln
Step15: End For
Step16: End For
Step17: Apply LSTM with best hyper-parameters
Step18: Predict the Load
Step19: Evaluate metrics

In Agorithm-1, the parameters population size (30), visual (1.5), step (0.5), crowd_factor (0.61), maxIter, MaxIter (200) are initialized and randomly generate population for LSTM (Learning rate [0.0001,0.001], epoch [10,200], batchsize [32,64,128,256], hidden units [32,64,128,256,512]). The fitness value (RMSE) can be computed using eq. (18). The lowest fitness value is assigned as a best solution. In this paper, FSO algorithm is modified based on the position updation of different behavior. For preying behavior,

$$F_{next} = X_i + (rand - 0.5) \times step \times (F_j - F_i) \quad (13)$$

For swarming behavior

$$F_{next} = F_i + (rand - 0.5) \times step \times (F_c - F_i) \times \rho \quad (14)$$

For chasing behavior:

$$F_{next} = F_i + (rand - 0.5) \times step \times (F_{max} - F_i) \times \rho \quad (15)$$

Figure 4 shows the work flow of proposed LSTM-FSO approach.

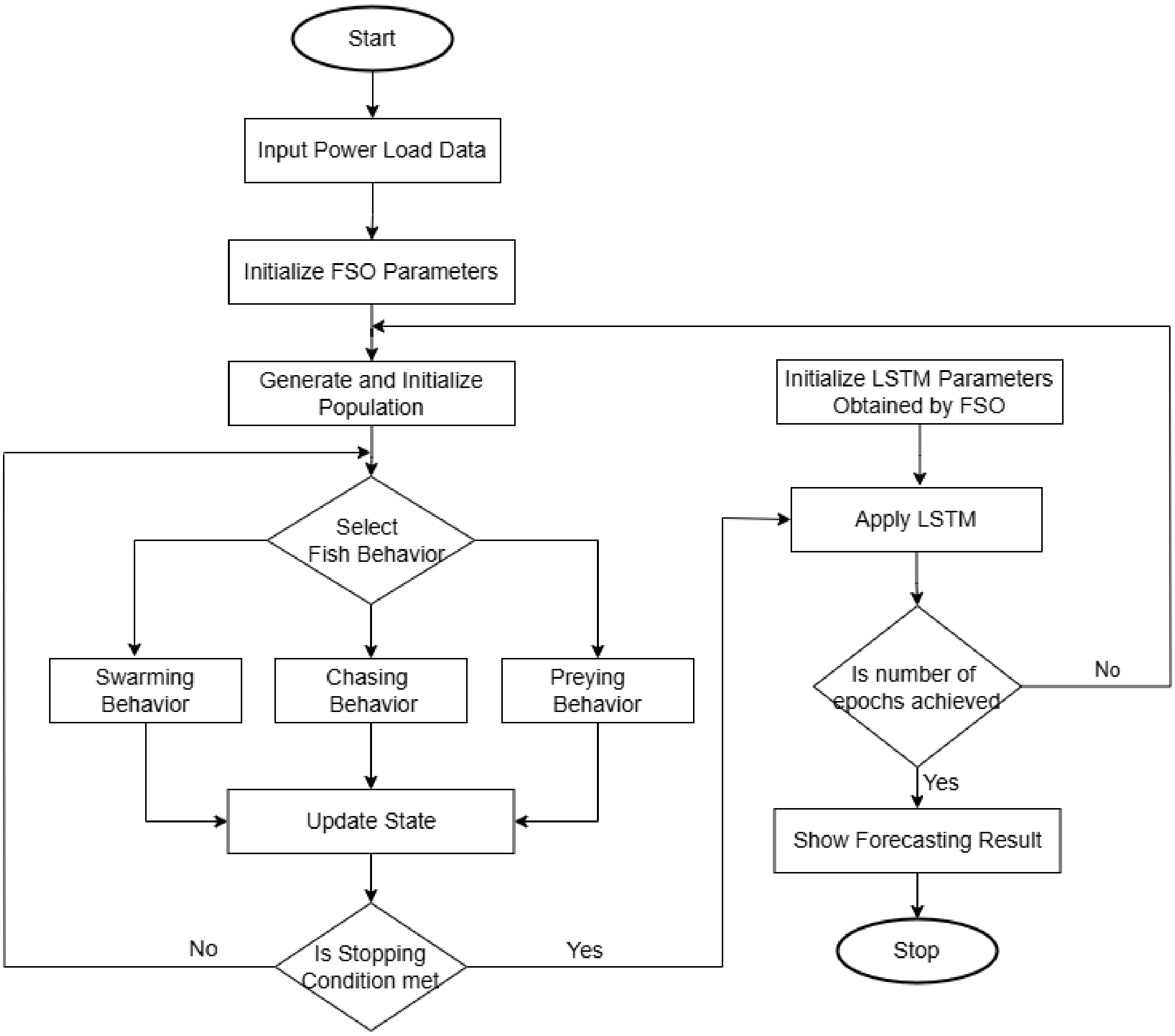

Figure 4: Flow of LSTM-FSO

5 Experimental results

This section analyzes the performance of the proposed work through experimental results. This investigation was conducted using a PC equipped with an i5-9300H processor and 8 gigabytes of RAM. Colab [45] provides a hosted solution for Jupyter notebooks. All experiments were conducted using Python. The evaluation measures selected for the model include mean absolute error (MAE), mean absolute percentage error (MAPE), root mean square error (RMSE), and coefficient of determination (R2). The RMSE effectively represents the dispersion of errors, whereas R2 signifies the linear correlation between expected and actual values, approaching 1 as the predicted and actual values converge. The formulas for each error indicator are presented in the subsequent equations:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (16)$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \quad (17)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (18)$$

$$R^2 = 1 - \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \div \sum_{i=1}^{n}\left(y_i - \frac{1}{m}\sum_{i=1}^{m} y_i\right)^2 \quad (19)$$

where y_i is the original load value, $\hat{y}_i$ is the forecasted load value, and m is the number of forecast points. The proposed approach is compared with traditional algorithms LSTM, CNN, Attention-LSTM [28] , AC-BiLSTM [29] and LSTM-PSO.

Table 2 shows the model default parameter for FSO and LSTM. The following parameters are set to the FSO algorithm: Maximum iteration = 200, visual = 1.5, step = 0.5, crowd factor = 0.61 and number of population = 30. For LSTM, the default parameter for learning rate = 0.001, epoch = 50, batchsize = 32 and hidden units 64.

Table 2: Default model parameter

Model	Parameter	Default Value
FSO	Maximum Iteration Visual Step Crowd Factor No. of Population	200 1.5 0.5 0.61 30
LSTM	Learning rate epoch batchsize hidden units	0.001 50 32 64

Table 3 shows performance metrics for different approaches.

Table 3: Performance metrics for different approaches

Model	MAE	MAPE	RMSE	R^2
LSTM	132.048	1.5662	185.82	0.952
CNN	113.331	1.3249	164.45	0.975
Attention-LSTM [28]	171.67	1.229	233.651	0.911
AC-BiLSTM [29]	89.9	1.05	132.1	0.99
LSTM-PSO	94.521	1.1460	115.46	0.982
LSTM-FSO	86.524	0.9452	102.56	0.991

Figure 5 shows the performance of comparison of different approach. The proposed approach reduces MAE and RMSE compared to other methods. From the result, the proposed LSTM-FSO, reduces MAE upto 34.47% over LSTM, 23.65% over CNN, 49.59% over Attention-LSTM [28], 3.75% over AC-BiLSTM [29] and 8.46% over LSTM-PSO. Similarly, the proposed LSTM-FSO, reduces MAPE upto 39.65% over LSTM, 28.66% over CNN, 23.09% over Attention-LSTM [28], 9.98% over AC-BiLSTM [29] and 17.52% over LSTM-PSO.

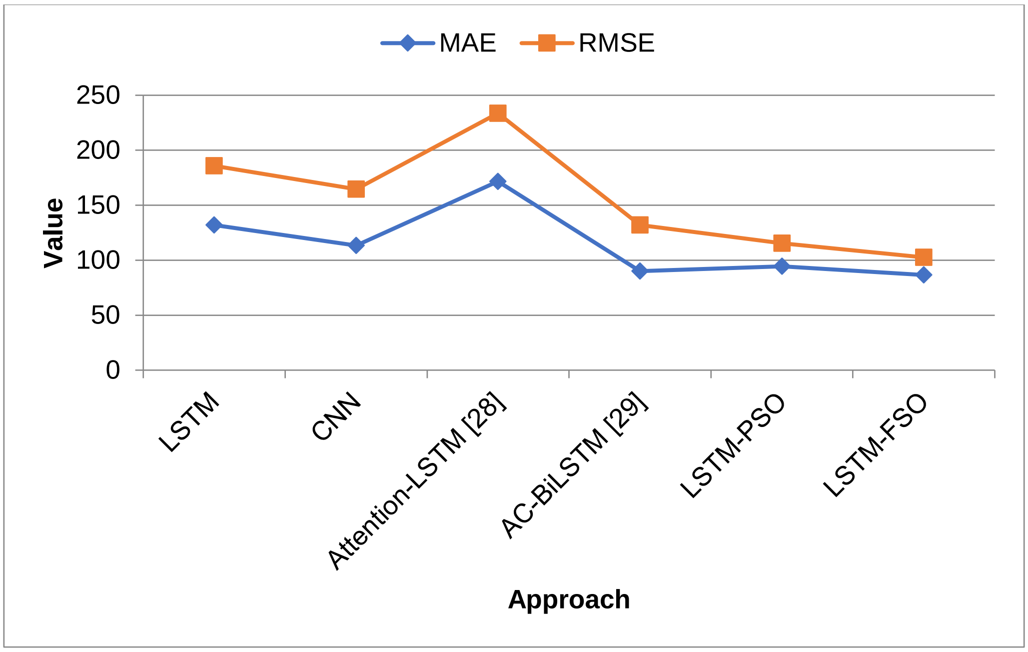

Figure 5: MAE and RMSE comparison

Figure 6 shows the comparison of MAPE and R^2 comparison.

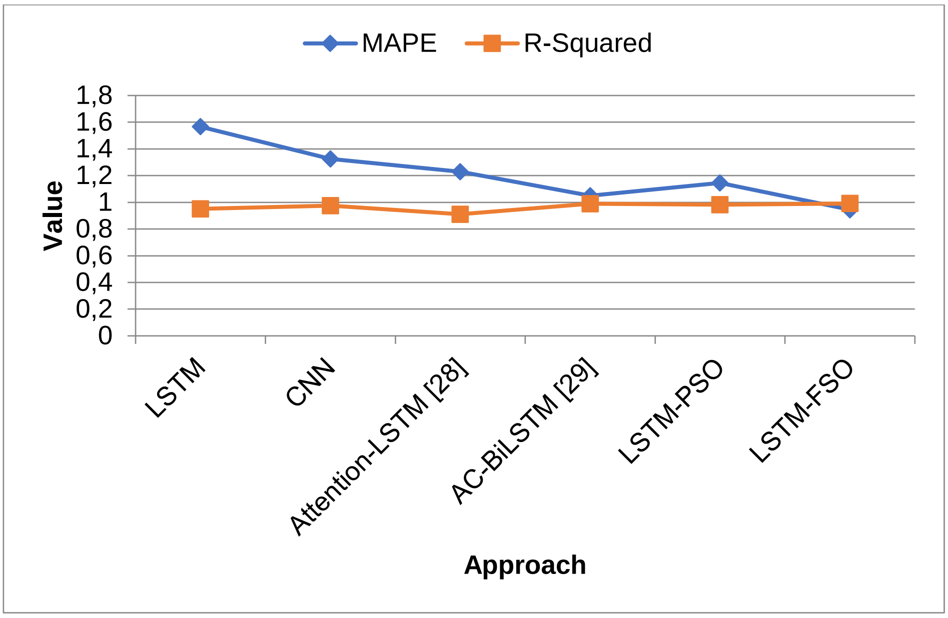

Figure 6: MAPE and R^2 comparison

From the result, the proposed LSTM-FSO, reduces RMSE upto 44.81% over LSTM, 37.63% over CNN, 56.10% over Attention-LSTM [28], 22.36% over AC-BiLSTM [29] and 11.17% over LSTM-PSO. Similarly, the proposed LSTM-FSO, increases R^2 upto 3.93% over LSTM, 1.61% over CNN, 8.07% over Attention-LSTM [28], 0.1% over AC-BiLSTM [29] and 0.90% over LSTM-PSO.

The optimal hyper parameter for LSTM is shown in Table 4.

Table 4: Optimal Hyper parameter for LSTM

Parameter	Default Value
Learning rate	0.001
epoch	150
batchsize	64
hidden units	256

Figure 7 shows the forecasted results for different models.

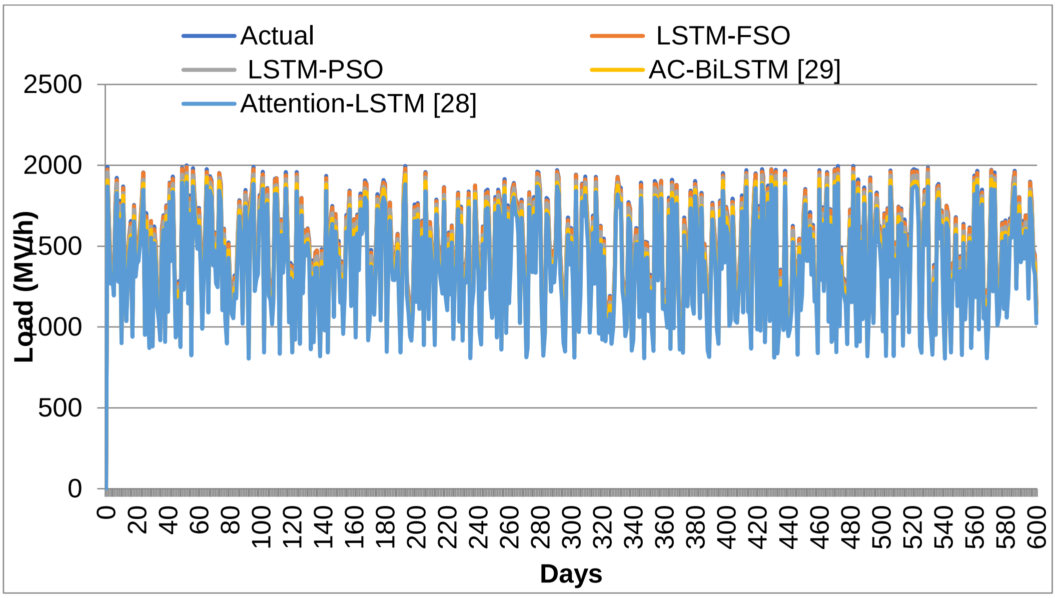

Figure 7: Forecasted result

6 Conclusion

Electrical load forecasting is a critical component of power systems, significantly impacting the national economy and the long-term stability of society. This paper introduces a power load prediction model that integrates the Fish Swarm Optimization algorithm with the Long Short-Term Memory neural network. The FSO algorithm was employed to maximize the number of time steps and the neurons inside the LSTM layers. The model's hyperparameters were established following extensive testing of various values to enhance performance outcomes. The experimental outcomes of this hybrid model were juxtaposed with the results of other existing models to assess its efficacy. The hybrid model demonstrated superior performance relative to other models based on the assessment criteria RMSE and MAE.

References

[1] Wang, Z., Zhou, X., Tian, J., & Huang, T. (2021). Hierarchical parameter optimization-based support vector regression for power load forecasting. Sustainable Cities and Society, 71, 102937. DOI: 10.1016/j.scs.2021.102937

[2] Debnath, K. B., & Mourshed, M. (2018). Forecasting methods in energy planning models. Renewable and Sustainable Energy Reviews, 88, 297-325. DOI: 10.1016/j.rser.2018.02.002

[3] Shang, C., Gao, J., Liu, H., & Liu, F. (2021). Short-term load forecasting based on PSO-KFCM daily load curve clustering and CNN-LSTM model. Ieee Access, 9, 50344-50357. DOI:10.1109/ACCESS.2021.3067043

[4] Pavlatos, C., Makris, E., Fotis, G., Vita, V., & Mladenov, V. (2023). Enhancing Electrical Load Prediction Using a Bidirectional LSTM Neural Network. Electronics, 12(22), 4652. https://doi.org/10.3390/electronics12224652

[5] Zhang, X., & Wang, J. (2018). A novel decomposition ensemble model for forecasting short-term load-time series with multiple seasonal patterns. Applied Soft Computing, 65, 478-494. https://doi.org/10.1016/j.asoc.2018.01.017

[6] Wang, R., Wang, J., & Xu, Y. (2019). A novel combined model based on hybrid optimization algorithm for electrical load forecasting. Applied Soft Computing, 82, 105548. DOI: 10.1016/j.asoc.2019.105548

[7] Chodakowska, E., Nazarko, J., & Nazarko, Ł. (2021). Arima models in electrical load forecasting and their robustness to noise. Energies, 14(23), 7952. https://doi.org/10.3390/en14237952

[8] Bassi, S., Gomekar, A., & Murthy, A. V. (2019). A learning algorithm for time series based on statistical features. International Journal of Advances in Engineering Sciences and Applied Mathematics, 11(3), 230-235. DOI:10.1007/s12572-019-00253-6

[9] Kottath, R., Singh, P., & Bhowmick, A. (2023). Swarm-based hybrid optimization algorithms: an exhaustive analysis and its applications to electricity load and price forecasting. *Soft computing*, *27*(19), 14095-14126 DOI:10.1007/s00500-023-07928-0

[10] Dai, Y., & Zhao, P. (2020). A hybrid load forecasting model based on support vector machine with intelligent methods for feature selection and parameter optimization. Applied energy, 279, 115332. DOI: 10.1016/j.apenergy.2020.115332

[11] Guo, W., Che, L., Shahidehpour, M., & Wan, X. (2021). Machine-Learning based methods in short-term load forecasting. The Electricity Journal, 34(1), 106884. DOI: 10.1016/j.tej.2020.106884

[12] Moon, J., Kim, J., Kang, P., & Hwang, E. (2020). Solving the cold-start problem in short-term load forecasting using tree-based methods. Energies, 13(4), 886. https://doi.org/10.3390/en13040886

[13] Abumohsen, M., Owda, A. Y., & Owda, M. (2023). Electrical load forecasting using LSTM, GRU, and RNN algorithms. Energies, 16(5), 2283. https://doi.org/10.3390/en16052283

[14] Al Mamun, A., Sohel, M., Mohammad, N., Sunny, M. S. H., Dipta, D. R., & Hossain, E. (2020). A comprehensive review of the load forecasting techniques using single and hybrid predictive models. IEEE access, 8, 134911-134939. DOI:10.1109/ACCESS.2020.3010702

[15] Li, J., Deng, D., Zhao, J., Cai, D., Hu, W., Zhang, M., & Huang, Q. (2020). A novel hybrid short-term load forecasting method of smart grid using MLR and LSTM neural network. IEEE Transactions on Industrial Informatics, 17(4), 2443-2452. DOI:10.1109/TII.2020.3000184

[16] Bashir, T., Haoyong, C., Tahir, M. F., & Liqiang, Z. (2022). Short term electricity load forecasting using hybrid prophet-LSTM model optimized by BPNN. Energy reports, 8, 1678-1686. DOI: 10.1016/j.egyr.2021.12.067

[17] Saoud, A., & Recioui, A. (2023). Load Energy Forecasting based on a Hybrid PSO LSTM-AE Model. Algerian Journal of Environmental Science and Technology, 9(1). https://doi.org/10.3390/en13236241

[18] Wei, T., & Pan, T. (2021). Short-term power load forecasting based on LSTM neural network optimized by improved PSO. Journal of System Simulation, 33(8), 1866-1874. https://doi.org/10.1016/j.egyr.2024.01.026

[19] Jiang, A., Qin, Z., Faulder, D., Cladouhos, T. T., & Jafarpour, B. (2022). Recurrent neural networks for short-term and long-term prediction of geothermal reservoirs. Geothermics, 104, 102439.

[20] Dong, X., Deng, S., & Wang, D. (2022). A short-term power load forecasting method based on k-means and SVM. Journal of Ambient Intelligence and Humanized Computing, 13(11), 5253-5267. DOI:10.1007/s12652-021-03444-x

[21] Li, K., Huang, W., Hu, G., & Li, J. (2023). Ultra-short term power load forecasting based on CEEMDAN-SE and LSTM neural network. Energy and Buildings, 279, 112666. DOI: 10.1016/j.enbuild.2022.112666

[22] Veeramsetty, V., Reddy, K. R., Santhosh, M., Mohnot, A., & Singal, G. (2022). Short-term electric power load forecasting using random forest and gated recurrent unit. Electrical Engineering, 104(1), 307-329. DOI:10.1007/s00202-021-01376-5

[23] Wan, A., Chang, Q., Khalil, A. B., & He, J. (2023). Short-term power load forecasting for combined heat and power using CNN-LSTM enhanced by attention mechanism. Energy, 282, 128274. https://doi.org/10.1016/j.energy.2023.128274

[24] Veeramsetty, V., Chandra, D. R., & Salkuti, S. R. (2021). Short-term electric power load forecasting using factor analysis and long short-term memory for smart cities. International Journal of Circuit Theory and Applications, 49(6), 1678-1703. DOI:10.1002/cta.2928

[25] Xiang, W., Xu, P., Fang, J., Zhao, Q., Gu, Z., & Zhang, Q. (2022). Multi-dimensional data-based medium-and long-term power-load forecasting using double-layer CatBoost. Energy Reports, 8, 8511-8522. DOI: 10.1016/j.egyr.2022.06.063

[26] Ullah, F. U. M., Ullah, A., Khan, N., Lee, M. Y., Rho, S., & Baik, S. W. (2022). Deep Learning Assisted Short-Term Power Load Forecasting Using Deep Convolutional LSTM and Stacked GRU. Complexity, 2022(1), 2993184. DOI:10.1155/2022/2993184

[27] Xu, H., Fan, G., Kuang, G., & Song, Y. (2023). Construction and application of short-term and mid-term power system load forecasting model based on hybrid deep learning. Ieee Access, 11, 37494-37507. DOI: 10.1109/ACCESS.2023.3266783

[28] Dai, Y., Zhou, Q., Leng, M., Yang, X., & Wang, Y. (2022). Improving the Bi-LSTM model with XGBoost and attention mechanism: A combined approach for short-term power load prediction. Applied Soft Computing, 130, 109632. https://doi.org/10.1016/j.asoc.2022.109632

[29] Liu, F., & Liang, C. (2024). Short-term power load forecasting based on AC-BiLSTM model. Energy Reports, 11, 1570-1579. https://doi.org/10.1016/j.egyr.2024.01.026

[30] Schmidhuber, J., & Hochreiter, S. (1997). Long short-term memory. Neural Comput, 9(8), 1735-1780. DOI:10.1162/neco.1997.9.8.1735

[31] Qian, LX., Shao, Z., & Xin, J. (2002). An optimizing method based on autonomous Animats: fish-swarm algorithm. Syst Eng Theor Pract, 22(11):32 DOI:10.21307/ijssis-2017-474

[32] Peng, Z., Dong, K., Yin, H., & Bai, Y. (2018). Modification of fish swarm algorithm based on levy flight and firefly behavior. Computational Intelligence and Neuroscience, 2018(1), 9827372. DOI:10.1155/2018/9827372

[33] Aguilar Madrid, E., & Antonio, N. (2021). Short-term electricity load forecasting with machine learning. Information, 12(2), 50.; https://doi.org/10.3390/info12020050

[34] Aldegheishem, A., Anwar, M., Javaid, N., Alrajeh, N., Shafiq, M., & Ahmed, H. (2021). Towards sustainable energy efficiency with intelligent electricity theft detection in smart grids emphasising enhanced neural networks. IEEE Access, 9, 25036-25061.
DOI:10.1109/ACCESS.2021.3056566