Hybrid Optimized Machine Learning Models for Forecasting Electricity Demand and Generation: A Comparative Analysis

Liandian Jiang[1], Pengfei Huang[1], Junyang Tian[1], Haiyong Li[1], Bin Liu[1], Shan Liang[1], Jian Wang[2,*]
[1]Guangxi Power Grid Co., Ltd, Nanning 530023, China
[2]Shandong University, Jinan 250100, China
E-mail: wjianvip@126.com
[*]Corresponding author

Keywords: electricity demand forecasting, maximum electricity generation, support vector regression, catboost model, metaheuristic optimization algorithm

Received: December 3, 2024

Increasingly complicated and volatile electricity demand considerably complicates the job of operators and planners of energy systems. For enhancing grid reliability and efficiency in resource allocation, realistic forecasting is thus in high demand regarding overall electricity demand and peak generation capacity. The study will introduce a new approach by adopting ML techniques, namely CatBoost and support vector regression, for predicting electricity demand and maximum generation capacity. Optimization algorithms for metaheuristics, such as phasor particle swarm optimizer and chaos game optimizer, are used to improve model performance. To this end, four hybrid models were developed: namely, CatBoost-CGO, CatBoost-PPSO, SVR-CGO, and SVR-PPSO. The performance was evaluated using statistical indices such as MAE, RMSE, MAPE, and R^2. Among the models, CatBoost-PPSO demonstrated the highest accuracy with $R^2 = 0.955$ for total demand and $R^2 = 0.958$ for maximum generation in the test set, outperforming other combinations. These results validate the effectiveness of hybrid optimization in improving prediction accuracy.

Povzetek: Razviti in testirani so štirje hibridni modeli (CatBoost/SVR z optimizatorjema PPSO/CGO) za napovedovanje električnega povpraševanja in največje proizvodnje. Najbolje se izkaže CatBoost-PPSO, ki doseže najnižje napake.

1 Introduction

Electricity demand forecasting helps in keeping the operation and planning of power systems effective. Correct forecasts of overall electricity demand and peak generation of electricity are required in order for the utility to be aware, in terms of decisions on resource planning, grid management, and infrastructure plans[1]. Due to increasing complexity and dynamism among energy markets, strong and reliable forecasting models are increasingly in demand, which, upon adaptation to changed conditions, will deliver proper predictions effectively[2].

In the last couple of years, ML methodologies have emerged as proficient tools in forecasting electricity demand, being able to uncover intrinsic patterns and interrelations present in the historical data. Of these, SVR and CatBoost have grown as two well-recognized approaches applied to problems related to extreme non-linearity and high-dimensionality. The performance of these models can be further improved using optimization techniques that are suitably tuned to improve their predictive accuracies. For example, PPSO and CGO represent the optimization methods giving the paths for improving parameters in ML models, thus enhancement of predictive accuracy and efficiency.

These comparative insights have very important implications for energy providers, policy analysts, and researchers alike. Identifying the best combination of algorithms and optimization techniques greatly enhances the reliability of forecasting in terms of electricity demand, optimizes resource utilization, and contributes toward a much more robust and sustainable energy infrastructure.

1.1 Related studies

The importance of forecasting electricity and energy consumption has attracted several researchers to forecasting using various methods and models. A wide range of ML-based models has been applied in different forms with supervised learning algorithms, including decision trees, SVM, linear, and logistic regression, along with neural networks.

[3] Proposed an NN-based grey forecasting model, namely the NNGM, to project electricity consumption patterns. The present study investigates the prediction performance of the NNGM for electricity consumption using datasets from the Turkish governmental database. From the empirical results, one could perceive that the NNGM performed remarkably well in the forecasting of electricity consumption. [4] This section presents the

application of ANN to forecast long-term energy usage in Greece, using the MLP model with different trials of architecture. Real input data and real output data are used to influence the energy consumption for training, validating, and testing purposes. The ANN model predicts energy consumption for the years ranging from 2005 to 2015. Comparison analysis underlines the accuracy of the ANN method, as compared to other methods like linear regression and SVM. The study pinpoints that correct forecasting of energy consumption is imperative for the development of an appropriate energy policy and to satisfy socio-economic factors, underlining the relevance of the method for long-term energy consumption prediction in Greece.[5] It proposes an improved ANN model for the hourly forecasting of building electricity consumption. The parameters of the ANN structure have been optimized by using an improved version of the Particle Swarm Optimization algorithm, iPSO. The important input variables are selected and the model is simplified with the help of PCA. To assess the performance of this model, 2 datasets have been used: one from the Energy Prediction Shootout contest and the other from a campus building in East China. Comparative analysis with conventional ANN and hybrid Genetic Algorithm-ANN (GA-ANN) models shows both versions of iPSO-ANN and GA-ANN with higher accuracy. Besides, it is observed that iPSO-ANN performs with a higher modeling time compared to the GA-ANN. Hopefully, the proposed model will be able to demonstrate a very feasible approach toward real-time prediction of building electricity consumption.

In a study done by [6] a novel approach to predict natural gas consumption is introduced, employing BMA to effectively handle uncertainties related to model structure and parameters, consequently improving prediction accuracy. 6 variables, such as GDP, urban population, energy consumption structure, industrial composition, energy efficiency, and exports of goods and services, are selected for prediction purposes. Comparative evaluation against the gray prediction model, Linear regression model, and ANNs demonstrates the versatility and reliability of BMA in forecasting the expected substantial increase in natural gas consumption. Studies of [7] evaluate various regression analyses using different time intervals of hourly and daily data from a research house. While multiple linear regression models incorporating outdoor temperature and solar radiation improve the R2, they also worsen the RMSE. The paper aims to serve as a comprehensive resource for future research in energy forecasting, suggesting a move towards personalized models for households based on smart meter data and user-friendly software tools. [8] explores 3 unique approaches of the ARIMA-ANFIS model to predict annual energy consumption in Iran. Initially, the ARIMA model is applied to 4 input features, and its nonlinear residuals are predicted using 6 diverse ANFIS structures. The second approach combines ARIMA predictions with the 4 input features for ANFIS prediction. To tackle data scarcity, the third approach integrates AdaBoost data diversification. Results indicate substantial enhancements in prediction accuracy across all hybrid patterns, particularly with the third approach, leveraging the diversification model. [9] presents a method for predicting electricity consumption in 28 commercial buildings in Seo-gu, Gwangju, South Korea, using publicly available monthly and daily consumption data. It addresses the challenge of accurately selecting parameters that influence energy usage, proposing a 2-step approach. Firstly, a customized electricity consumption model is developed based on individual building characteristics within the building community. Secondly, additional data such as building features and energy equipment types are integrated using dynamic-time-warping-based clustering classification. In comparison with conventional time-series analysis approaches, the suggested strategy performs better as it uses ML techniques for both model building and optimization. This development has the potential to improve the accuracy of electricity consumption estimates for commercial buildings and offer insightful information for controlling the supply and demand of power in both urban and building contexts.

[10] introduces a new stacking model aimed at predicting building energy consumption. By blending different base prediction algorithms and generating "meta-features," this model provides a comprehensive analysis of datasets from various angles. The practical application of the stacking model is demonstrated through 2 engineering case studies. Comparative evaluation against established prediction models, such as Random Forest and Gradient Boosted Decision Tree, reveals the superior performance of the stacking method in terms of accuracy, generalization, and robustness. In essence, this model diversifies available prediction techniques and enhances the algorithmic arsenal for empirical models. [11] presents a short-term electricity consumption prediction model using LSTM with an attention mechanism. Initially, the attention mechanism assigns weight coefficients to input sequence data. Following this, the output of each LSTM cell is computed using forward propagation, and errors between predicted and actual values are determined using back-propagation. Model weights are adjusted through gradient descent to minimize errors. Experimental results across various electricity consumption types demonstrate the effectiveness of the proposed model in improving prediction accuracy compared to existing approaches. [12] presents an ML-driven approach for forecasting power usage, exploring a range of techniques including linear regression, K Nearest Neighbors, XGBOOST, random forest, and ANNs. The study leverages historical electricity usage data obtained from a power utility company, which undergoes preprocessing to handle outliers and missing values. Evaluation of model performance involves various assessment measures, revealing the efficacy of the proposed method in accurately predicting power usage. Notably, the K-Nearest Neighbors model emerges as the top performer, demonstrating particularly high accuracy in predicting agricultural production. [13] explores the application of deep learning in forecasting annual energy consumption for buildings and conducts a comparative evaluation of model performance. An innovative aspect of this study is the development of a model trained on data from multiple buildings, allowing building designers to input key

features early in the development process to predict annual average energy consumption. Results show that the ANN method surpasses other models in accurately predicting annual energy consumption.

Several studies have been conducted on electricity demand and energy consumption forecasting using machine learning and statistical approaches. Table 1 presents a comparative summary of selected state-of-the-art methods, highlighting their employed techniques, reported metrics, and identified limitations. Most of these works focused either on specific case studies such as buildings or used traditional machine learning techniques without hybrid optimization. Moreover, several studies lack detailed statistical performance metrics or comparisons across multiple algorithms. This study addresses these limitations by proposing hybrid models combining advanced regressors (CatBoost and SVR) with novel metaheuristic optimizers (PPSO and CGO), demonstrating significant improvements in forecasting accuracy.

Table 1: Summary of related studies

Study	Forecasting Method	Reported Metrics	Accuracy/Highlights	Identified Limitations
Hu (2017)	Neural Network + Grey Forecasting	MAPE	Good predictive behavior	No optimization; lacks statistical metrics
Ekonomou (2010)	ANN	Error (%)	Low error for long-term prediction	No hybrid methods or optimization
Li et al. (2015)	ANN + PCA	MAE, RMSE	Improved short-term accuracy	Building-level focus only
Fumo & Biswas (2015)	Linear Regression	MAE, RMSE	Simple model; acceptable performance	Lacks adaptability to nonlinear patterns
Barak & Sadegh (2016)	ARIMA–ANFIS hybrid	MSE = 0.026	Low MSE (0.026%)	Model complexity; no R^2 given
Hwang et al. (2020)	ML approach for commercial buildings	RMSE, MAE	Effective for commercial data	Application-limited; no comparative benchmarks
Wang et al. (2020)	Model Stacking	RMSE, Accuracy	9.5%–31.6% improvement over baselines	High model complexity
Lin et al. (2020)	LSTM + Attention	RMSE	Strong performance on time series	Requires large datasets
Vijendar Reddy et al. (2023)	ML Techniques	Accuracy	Broad machine learning application	No benchmarking or R^2 data

1.2 Main novelty

Motivated by the limitations observed in earlier research, such as insufficient adaptability to nonlinear patterns and lack of effective parameter optimization, the proposed approach aims to address these gaps. The investigation focuses on evaluating the comparative performance of the hybrid models and determining which configuration offers the optimal balance between predictive accuracy and computational efficiency. This paper considers 2 metaheuristic optimizers including PPSO and CGO to integrate them with SVR and CatBoost. Thien the developed hybrid models are compared based on several metrics, in order to present an efficient ML-based algorithm to forecast the electricity consumption by computing maximum generation and total demand. The aim of this study is for the proposed model to be prominent and to overcome existing models. It is hypothesized that the integration of CatBoost with PPSO will result in superior forecasting performance due to the model's advanced feature handling capabilities and the optimizer's enhanced global search behavior.

The rest of the paper is organized as follows. Section 2, contains a brief description of the compared models such as SVR and CatBoost. Also, the utilized metaheuristic optimizers are described. then the presented hybrid models are introduced. Section 3, presents the results of the numerical analysis of the hybrid models. Section 4 includes the results of the model's comparison, based on the statistical evaluation indices. Finally, section 5 concludes the comparison and discusses the selected hybrid model for predicting electricity consumption.

2 Algorithm's description & method of study

The descriptions of the ML models utilized in this comparison including SVR and CatBoost, are provided concisely. Also, the selected optimizers are introduced along with the hybrid models developed through integrating optimizers and ML models.

2.1 CatBoost algorithm

CatBoost (short for Category Boosting) is a powerful ML algorithm developed by Yandex [14], specifically designed to tackle complex predictive modeling tasks with efficiency and accuracy. It fundamentally employs, at a

base level, the gradient boosting technique, which is one of the most popular ensembles learning methods that fits a sequence of decision trees in improving a previous model's prediction. What actually makes CatBoost different is its unique capability to handle categorical features out-of-the-box: unlike classical methods, which require explicit preprocessing steps like one-hot encoding, CatBoost handles categoricals internally during the training phase, saving time and quite often, at the same time, improving model performance. It also contains many optimization strategies to enhance learning: ordered boosting, which trains instances in order of gradient; oblivious trees, special structure of trees that simplify computations. Besides that, prevention of overfitting is implemented in CatBoost by means of regularization methods, such as L2 regularization and tree depth regularization. So, CatBoost balances out model complexity and generalization; therefore, robust models are obtained that have the capability to give accurate predictions for data that has not been seen before. Mathematically, CatBoost optimizes an objective function, usually based on MSE for regression or cross-entropy for classification, via gradient boosting. The model parameters are optimized in order to minimize an objective function L (y, F(x)) with respect to the goal variable y and the input features x. Optimization of the objective function is performed in iterations through a gradual addition of decision trees in the ensemble, with each added tree aiming at a minimal residual of the previous ensemble predictions [14], [15]. In the context of this study, since the forecasting task is regression-based, the specific loss function employed is the Mean Squared Error (MSE), formally defined as:

$$L_{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - y_i)^2 \quad (1)$$

where y_i represents the true target value, y_i is the model's predicted value, and n denotes the number of samples.

2.2 Support vector regression algorithm

SVR is one of the supervised learning algorithms for regression, introduced by [16]. It belongs to a family of SVMs, which is mainly applied to classification. However, the extension of this to the domain of regression where one tries to predict continuous outcomes, rather than discrete classes, is known as SVR.

The basic idea behind the work of SVR is to find that function which does the job of approximating mappings from input variables to continuous output variable with minimal error. To show how the relationship of input and output variables is modeled, a hyperplane or set of hyperplanes is constructed in a high-dimensional space. The hyperplane(s) is/are chosen such that it/they possess maximum margin from the data points. The margin of the hyperplane is defined as the distance between the hyperplane and the nearest data point(s).

Unlike other traditional regression methods, the objective of SVR is to capture the points that are within a boundary of a specified margin from the hyperplane(s)in other words, support vectorsrather than precisely fitting all data. Because of this, SVR is more resistant against noise and outliers. The mathematical description of the SVR can be visualized with:

Given a training dataset $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ where x_i represents the input features and y_i represents the corresponding output values, SVR aims to find a function $f(x)$ that approximates the relationship between the input features and the output values. The primal optimization problem for SVR can be formulated as follows:

$$\text{Minimize: } \frac{1}{2}\| \boldsymbol{w} \|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \quad (2)$$

Subject to:

$$\begin{aligned} & y_i - \mathrm{w}^T\phi(\mathrm{x}_i) - b \le \varepsilon + \xi_i \\ & \mathrm{w}^T\phi(\mathrm{x}_i) + b - y_i \le \varepsilon + \xi_i^* \\ & \xi_i, \xi_i^* \ge 0, i = 1,2,\ldots,N \end{aligned} \quad (3)$$

where $\boldsymbol{w}$ is the weight vector. $\phi(x)$ is a mapping function to transform the input features x into a higher-dimensional space. b denotes the bias term. ε shows the epsilon-insensitive loss function parameter, which defines the size of the margin around the predicted values where no penalty is incurred.

ξ_i and ξ_i^* are slack variables, allowing for deviations from the margin and controlling the trade-off between the margin and the training error. C is a regularization parameter that balances the trade-off between maximizing the margin and minimizing the training error. A smaller value of C encourages a larger margin but may allow more training errors, while a larger value of C penalizes errors more heavily, potentially leading to a narrower margin.

The dual optimization problem, derived from the primal problem using Lagrange duality, involves finding the Lagrange multipliers α_i and α_i^* for each training sample using the equation below:

$$f(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)\langle\phi(x_i)\cdot\phi(x)\rangle + b \quad (4)$$

where $0 \le \alpha_i, \alpha_i^* \le C$.

The final regression function is given by:

$$f(x) = W^T\,\phi(x) + b \quad (5)$$

SVR often uses kernel tricks, where the inner product $\langle\phi(x_i), \phi(x_j)\rangle$ is replaced with a kernel function $K(x_i, x_j)$ to implicitly map the input features into a higher-dimensional space without explicitly calculating $\phi(x_i)$ and $\phi(x_j)$.

By using this formulation, SVR is able to minimize training error and maximize margin while identifying a hyperplane or group of hyperplanes that best match the connection between the input features and the output values [16], [17].

2.3 Chaos game optimization algorithm

The Chaos Game Optimization is a metaheuristic optimization technique inspired by the dynamic's principles of the "chaos game" concept originating from fractal theory. Talataheri & Azizi [18], in the year 2020, proposed the CGO and developed the algorithm to emulate the dynamics of chaos games in finding the optimum solutions for optimization problems. Chaos theory focuses on regular patterns that may come out from systems' behavior, like fractals, cycles that repeat

themselves, and subsystems that repeat. The systems are shown by these patterns as dynamic, self-similar, and self-organized entities. Chaos theory explains how even small changes to a dynamic system's starting circumstances can result in radically different behavior later on. This dependency on initial conditions underscores the system's sensitivity and its potential for diverse and unpredictable trajectories.

The CGO is a method for generating geometric patterns using randomness, famously demonstrated by the Sierpinski Triangle. Each vertex of the triangle corresponds to a color on a dice, determining the starting point. Rolling the dice guides subsequent movements towards vertices, marking points halfway. Despite randomness, the pattern converges into a self-similar fractal resembling the Sierpinski Triangle, showing order emerging from randomness. Fig. 1 illustrates this process, resulting in the Sierpinski triangle's creation. The self-similarity of the triangle is evident in the final shape.

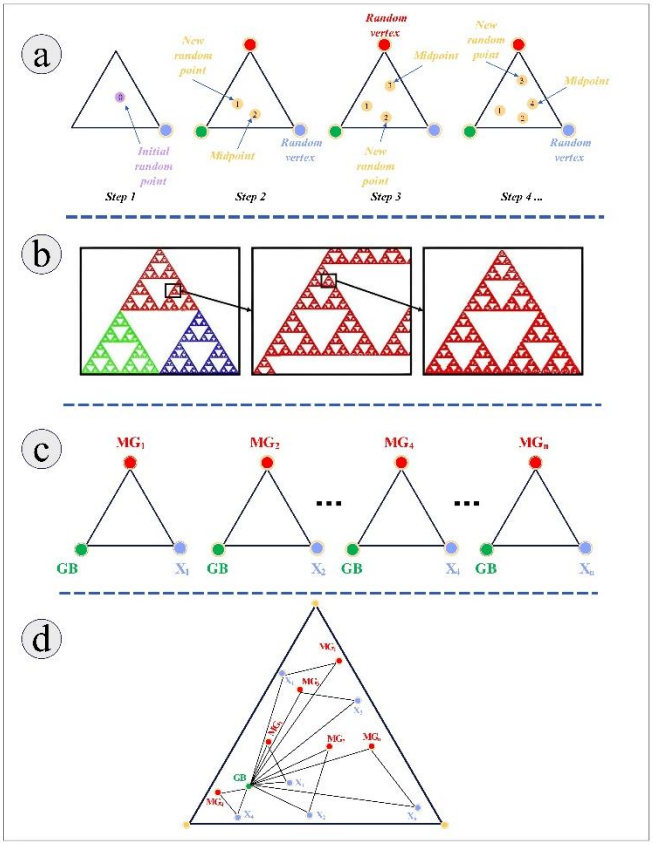

Figure 1: The chaos game process a) generating the Sierpinski triangle b) The self-similarity of the Sierpinski triangle c) the formation of temporary triangles d) multiple triangles in the search area

According to the mathematical representation of the CGO algorithm in Fig. 1, the CGO delves into a variety of potential solution (X), each representing potential seeds within the Sierpinski triangle. Each parameter x_{ij} in Eq. (5) is a decision variable inside each potential solution X_i. Also, the Sierpinsky triangle is the search space of the CGO algorithm and each potential solution inside the search space is called a qualified seed.

$$X = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_i \\ \vdots \\ X_n \end{bmatrix} = \begin{bmatrix} x_1^1 & x_1^2 & \cdots & x_1^j & \cdots & x_1^d \\ x_2^1 & x_2^2 & \cdots & x_2^j & \cdots & x_2^d \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ x_i^1 & x_i^2 & \cdots & x_i^j & \cdots & x_i^d \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ x_n^1 & x_n^2 & \cdots & x_n^j & \cdots & x_n^d \end{bmatrix} \quad (6)$$

where $\begin{cases} i = 1, 2, \ldots, n \\ j = 1, 2, \ldots, d \end{cases}$.

Also, n represents the total number of qualified seeds or potential solutions. d shows the dimension of each seed.

Each qualified seed has an incipient position where it is selected randomly using the equation below:

$$x_i^j(0) = x_{i,\,\min}^j + \mathrm{r} \,.\, \left(x_{i,\,\max}^j - x_{i,\,\min}^j\right) \quad (7)$$

where the expression $x_i^j(0)$ denotes the incipient position of each qualified seed. The minimum and maximum allowed values are $x_{i,\,\min}^j$ and $x_{i,\,\max}^j$, respectively, for the j-th decision variable and i-th potential solution. r is a number that is selected randomly in the range of 0 to 1.

Chaos theory investigates the innate patterns seen in dynamic systems, describing them as self-similar and self-organizing systems. When modeling these patterns as potential solutions (X) in optimization problems, initial

seeds are essential. Seed eligibility is determined by fitness values, with the highest indicating optimal candidates. The goal of the mathematical model is to produce suitable seeds in the search space so that a Sierpinski triangle is formed. Temporary triangles are created for each eligible seed, using Global Best (GB), Mean Group (MG_i), and chosen seed (Xi) as vertices. New eligible seeds are introduced via CGO, with movements guided by dice rolls and probabilities towards GB or MG_i. Constraints on seed movement are imposed through randomly generated factorials to adhere to CGO's principles. Mathematically, Eq. (7) demonstrates the aforesaid process for the seed number one.

$$seed_i^1 = X_i + \alpha_i \times (GB \times \beta_i - MG_i \times \gamma_i) \quad , \ i = 1 \ to \ n \tag{8}$$

where GB and MG_i are the best global solutions found so far, and the average values derived from a subset of chosen qualified seeds. α_i denotes a factorial generated randomly used to simulate movement constraints for the seeds. γ_i and β_i are 2 integer parameters for modeling the dice rolling probabilities, which can be selected between 0 and 1.

3 blue and 3 red sides on a dice are used by the second seed (GB). The seed travels in the direction of Xi (blue) or MGi (red), according to the color rolled. Movement is modeled similarly to the first seed, limited by randomly generated factors. This process is formulated using the following equation:

$$seed_i^2 = GB + \alpha_i \times (GB \times \beta_i - MG_i \times \gamma_i) \quad , \ i = 1 \ to \ n \tag{9}$$

where α_i, γ_i and β_i are the same parameters used in Eq. (7).

The third seed referred to as MG_i, employs a die featuring both blue and green faces. Its movement direction is determined by the color rolled, with blue directing it towards Xi and green towards GB. A random number-generating function is used to do this, and its result is either 0 or 1, representing the option of blue or green faces. Most importantly, the seed can move along the lines that join GB and Xi. Eq. (9) presents the mentioned procedure for the third seed:

$$seed_i^3 = MG_i + \alpha_i \times (GB \times \beta_i - MG_i \times \gamma_i) \quad , \ i = 1 \ to \ n \tag{10}$$

The 4th seed is created by a different procedure in order to incorporate mutation during position updates of qualifying seeds inside the search space. This seed's position adjustments are executed by introducing random changes to randomly selected decision variables. The mathematical expression of the aforesaid process can be demonstrated as follows:

$$seed_i^4 = X_i \left(x_i^k = x_i^k + R\right), \quad k = 1 \ to \ d \tag{11}$$

where k is an integer number selected in the range of 0 to d. R represents a random number uniformly distributed within the range of 0 to 1.

There are 4 different formulations for the variable α_i, which controls the restrictions on seed movement, in order to control and fine-tune the CGO's exploration and exploitation rates.

$$\begin{cases} \alpha_i = r \\ \alpha_i = 2 \times r \\ \alpha_i = 1 + (\delta \times r) \\ \alpha_i = \sim\varepsilon + (\varepsilon \times r) \end{cases} \tag{12}$$

where r is a random number that is distributed uniformly in the range 0 to 1, and ε and δ are 2 integer numbers selected in the range of 0 to 1.

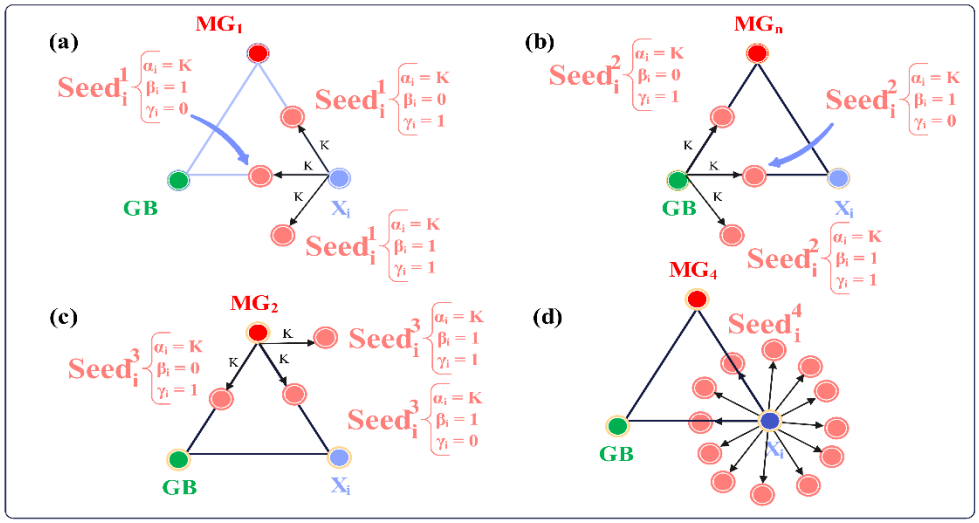

Figure 2: The process of position updating in the search space for a) first seed b) second seed c) third seed d) 4th seed.

To manage self-similarity in fractals, evaluating both existing and new seeds is crucial to determine their appropriateness for the search space. New solutions are compared with current ones, retaining superior ones while discarding those with lower fitness values, indicative of higher self-similarity. The substitution process simplifies

model complexity in mathematical approaches. In practice, all eligible seeds found are used to construct the complete Sierpinski triangle shape (Fig. 2). [18], [19].

2.4 Particle swarm optimization algorithm

PSO is a computational optimization technique inspired by the social behavior of bird flocking or fish schooling. It was originally developed by Dr. Eberhart and Dr. Kennedy in 1995 [20]. PSO is a population-based optimization algorithm that iteratively improves a set of candidate solutions (particles) according to a fitness function, aiming to find the optimal solution.

In PSO, each potential solution to the optimization problem is represented as a "particle" in an n-dimensional search space. Each particle has a position and a velocity vector, which are updated iteratively based on its own best-known position and the best-known position in the entire swarm. The position of a particle corresponds to a potential solution to the optimization problem, and the fitness of that solution is evaluated using an objective function.

At each iteration, particles adjust their velocities based on their previous velocities, where the equation below expresses it mathematically:

$$v_{id}^{f+1} = v_{id}^{f} + C_1 r_1^{f} \left(Pb_{id}^{f} - x_{id}^{f}\right) + C_2 r_2^{f} \left(Gb_{id}^{f} - x_{id}^{f}\right) \quad (13)$$

where the current velocity of the particle is shown as v_{id}^{f}, with f times of iterations in a search space with d number of dimensions, and v_{id}^{f+1} is the updated velocity of the particle. Pb_{id}^{f} is the best-known position of particle i (personal best), and Gb_{id}^{f} is the best-known position of any particle in the swarm (global best). C_1 and C_2 are denoting the acceleration coefficients, and r_1^{f} and r_2^{f} are random numbers uniformly distributed in the range of 0 to 1.

According to the PSO algorithm, each particle updates its position through the following equation:

$$x_{id}^{f+1} = x_{id}^{f} + v_{id}^{f+1} \quad (14)$$

x_{id}^{f+1} is the updated position of particle i. x_{id}^{f} is referred to the current position of particle i. v_{id}^{f+1} denotes the velocity of particle i, which can be calculated using Eq. (12) [21].

It iterates until any of the termination criteria, such as maximum number of iterations or satisfactory solution quality, are reached. The advantage of PSO is that it is very efficient in exploring complicated search areas and it is easy and convenient to use [20], [21].

2.5 Phasor particle swarm optimizer

PPSO is a PSO algorithm variant that has been proposed by Ghasemi et al. [22]. The basic idea of PPSO introduces a new mechanism that was inspired by the phasor theory in mathematics.Every particle in the PPSO algorithm is assigned an important parameter: phase angle, θ. It acts as an interface towards controlling and steering the movement of particles within the search space. In other words, the phase angle is one of the main ingredients of the whole control process in the behavior of the particles during the process of optimization.

Phase angle is one of the crucial parameters in PPSO, as it bears the signature of direction and amplitude over the particle velocities. The manipulation of phase angle may change the velocities of PPSO particles. This adjustment is typically achieved through an equation that incorporates the phase angle into the calculation of the particle velocities. [23]:

$$V_i^{T} = p(\theta_T) \times (Pb - X_T) + g(\theta_T) \times (Gb - X_T) \quad (15)$$

The mechanism of position updating in PPSO is presented using the following mathematical equation:

$$X_{T+1} = X_T + V_T \quad (16)$$

where all the parameters are similar to the parameters in the basic PSO. T is the current iteration and the parameters $p(\theta_T)$ and $g(\theta_T)$ can be calculated through the following equations:

$$p(\theta_T) = |\cos(\theta_T)|^{2\times\sin(\theta_T)} \quad (17)$$

$$g(\theta_T) = |\sin(\theta_T)|^{2\times\cos(\theta_T)} \quad (18)$$

Alongside updating the position and velocity, the phasor angle will also be adjusted using the equation below:

$$\theta_{T+1} = \theta_T + |\cos(\theta_T) + \sin(\theta_T)| \times 2 \times \pi \quad (19)$$

In other words, in PSO, the phase angle is a guiding parameter that orients the flight of particles toward regions of interest in the search space. Since the particle's velocity is dynamically updated according to this phase angle, it is then possible for the algorithm to explore complex optimization surfaces in the search for optimal or near-optimal solutions.

To clarify the relation between standard PSO and its phasor-based variant (PPSO), it is important to note that the phase angle θ in PPSO can be interpreted as an implicit generalization of the stochastic influence mechanisms present in classical PSO. While traditional PSO relies on random multipliers r_1 and r_2 to modulate cognitive and social components of particle motion, PPSO encapsulates such modulation into deterministic trigonometric functions of the phase angle, i.e., $p(\theta_T)$ and $g(\theta_T)$. These functions control the intensity and direction of movement toward both personal best (Pb) and global best (Gb) positions. Thus, the phase angle serves not as a completely external addition, but as a mathematically grounded and dynamic extension of existing update rules in PSO. This formulation offers more control over convergence behavior and exploration patterns, especially in complex or multimodal search spaces.

2.6 PAWN sensitivity index

Pianosi and Wagener (2015) [24] Introduce a new method for defining and computing a density-based sensitivity index, PAWN. If some uncertainty about an input is vanished, the output *CDF* variance represents the impact of that input. Precisely, the computation of KS between the conditional distribution and the empirical unconditional *CDF* of model output, when x_i is kept constant, is required.

From these individual results, the maximum, median, or other quantile of the sensitivity index statistic can be obtained, while the KS statistic is computed under various conditioning scenarios.

This index makes it possible to filter out non-influential inputs and rank inputs according to how much they contribute to output uncertainty.

The benefit of PAWN is that it may be used with any kind of output distribution, including ones that are extremely skewed. Also, researches show that the PAWN index is highly sensitive to design parameters in models, compared to other indices [25]. Furthermore, it is adaptable to concentrate on particular output sub-ranges, such as extreme values, which is very helpful for applications like natural hazard models [24].

2.7 Hybrid models & dataset description

Aiming to forecast electricity consumption, 2 ML-based algorithms are adopted including CatBoost and SVR. To increase the performance of the algorithms, the adopted models are integrated with 2 optimizers. Hence, optimizers such as CGO and PPSO are selected for this purpose. The hyperparameters for ML models used, with search spaces defined as follows: for CatBoost – learning rate ∈ [0.01, 0.3], depth ∈ [3, 10], iterations ∈ [100, 1000]; for SVR – C ∈ [0.1, 100], epsilon ∈ [0.01, 0.5], gamma ∈ [0.001, 1]. Finally, 4 hybrid models such as CatBoost-CGO, CatBoost-PPSO, SVR-CGO, and SVR-PPSO are created and compared.

Several input data points were selected to train and test the hybrid models. The datasets contain various types of information and raw data, including the date of data collection, weather conditions, temperature, and more. The dataset was divided into 80% training and 20% testing sets using time-based chronological splitting, ensuring no data leakage between phases and preserving the temporal dependencies essential. This information was adapted from a study conducted by Salehin et al. in 2024 [26], where the datasets were sourced from Bangladesh's governmental website. Table 2 introduces the datasets used as input for the hybrid models, along with the expected outputs required from the models.

Table 2: Input and output data description [26]

Name of the data	Description
Day	Implies the day on which the data is recorded.
Month	Denotes the month in which data is recorded.
Year	Shows the year that the data is recorded.
Temp (C)	Temperature at a specific location using the sensor. (Celsius ℃)
Temp_max (C)	The maximum temperature recorded during a particular timeframe. (Celsius ℃)
Temp_min (C)	Minimum temperature recorded during a particular timeframe. (Celsius ℃)
Temp_ave (C)	Average temperature recorded during a particular timeframe. (Celsius ℃)
Surface pressure (Pa)	The air pressure measured at ground level.
Wind_speed_max (m\s)	Maximum wind speed measured at 50 m above the earth’s surface.
Wind_speed_min (m\s)	Minimum wind speed measured at 50 m above the earth’s surface.
Wind_speed_ave (m\s)	Average wind speed measured at 50 m above the earth’s surface.
Prectotcorr	Measurement associated with precipitation or correlation involving precipitation.
Total demand (MW)	Total quantified electricity demand. (Megawatts)
Max generation (MW)	Maximum electricity generation documented. (Megawatts)

The aforementioned datasets are in different levels of correlation with each other. Some of the data are more correlated with the total demand and max generation, i.e., they have more influence on the measured amount of electricity demand and electricity generation. The result of correlation analysis on the variables of input data is illustrated as a heat map which is presented in Fig. 3:

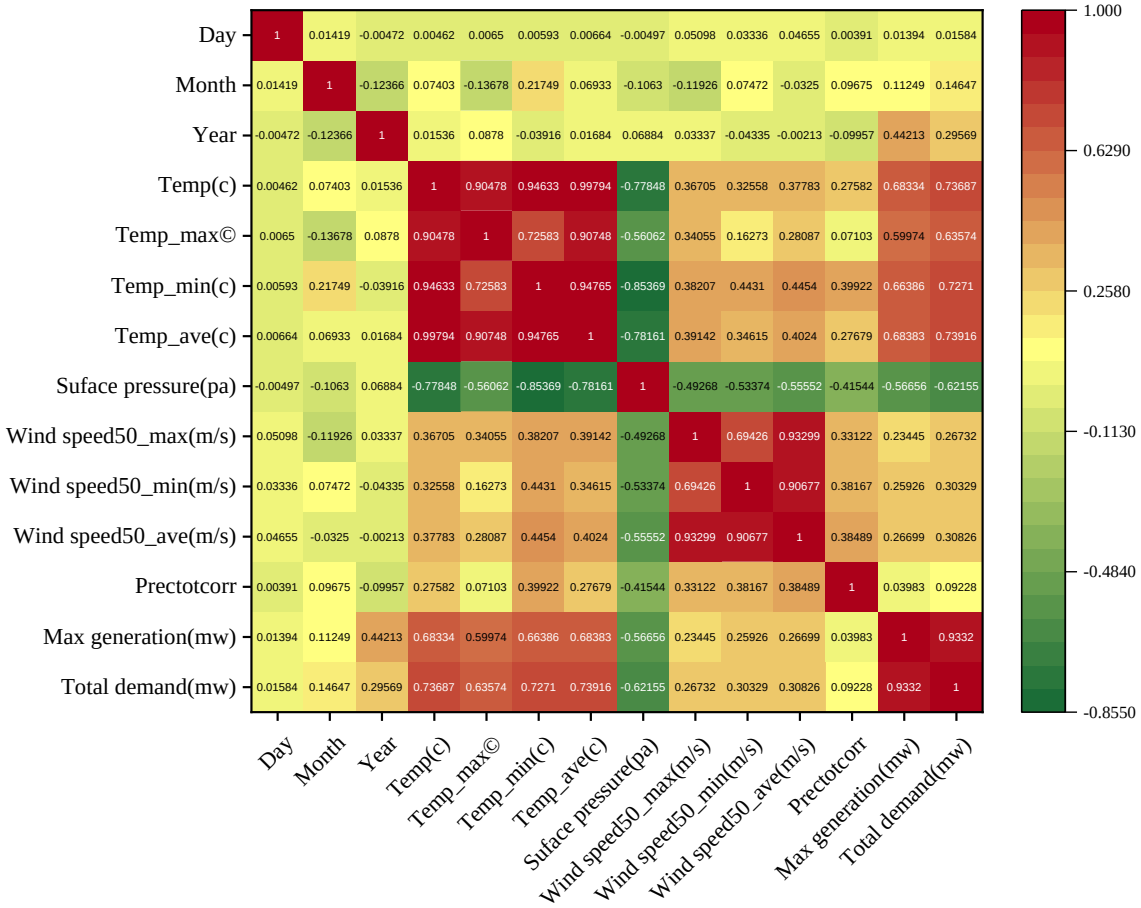

Figure 3: The correlation among variables of input data

According to Fig. 3, the heat map determines the correlation among the variables, with warmer colors denoting high correlation and colder colors denoting low correlation between the 2 variables. Based on this, maximum generation and total electricity demand are correlated highly with each other. Both maximum generation and total demand show a high correlation with variables related to temperature, while their correlation with surface pressure is relatively the lowest. As a matter of fact, the surface pressure demonstrates a low correlation with most of the variables.

Moreover, the heat map implies that wind speed has a higher correlation with electricity generation and electricity demand, compared to precipitation. Also, the correlation between total demand and temperature is partly higher than the correlation between maximum generation and temperature.

In order to determine how uncertainties in input data, affect the outputs of the forecasting models, a sensitivity analyze implemented. By executing the PAWN sensitivity analysis on the inputs of Table 1, the following results are achieved.

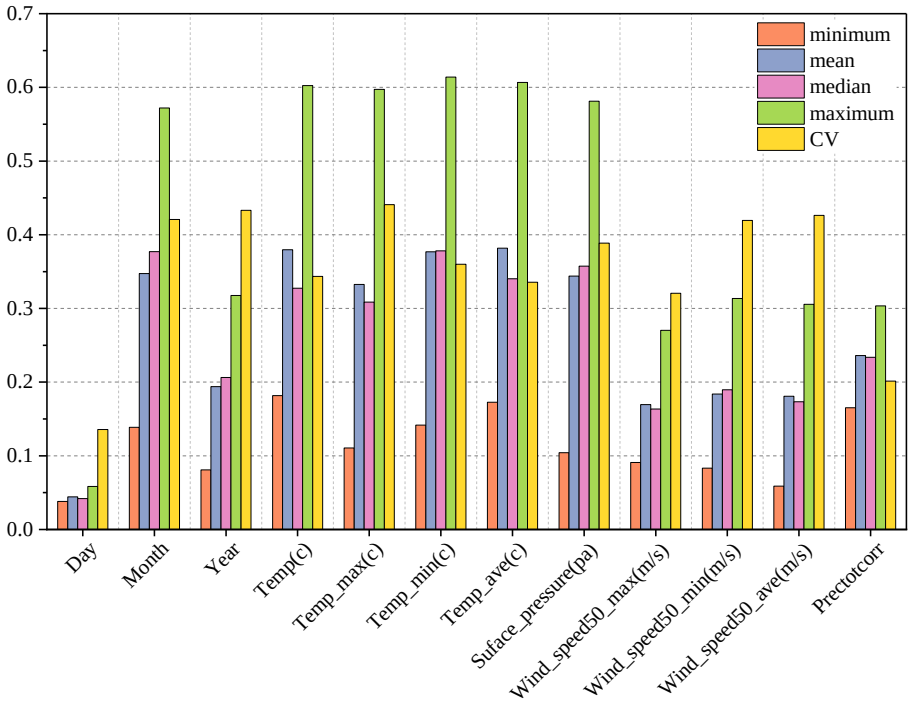

Figure 4: PAWN sensitivity analysis for predicting total demand

Fig. 4 Implies that the variables related to the temperature are showing higher sensitivity to the model, i.e. predicting total electricity demand by the hybrid models is mostly under the influence of temperature. According to the graph, the month of the recording of the data has a high impact on outputs, while the day of the recording data is not effective on the outputs.

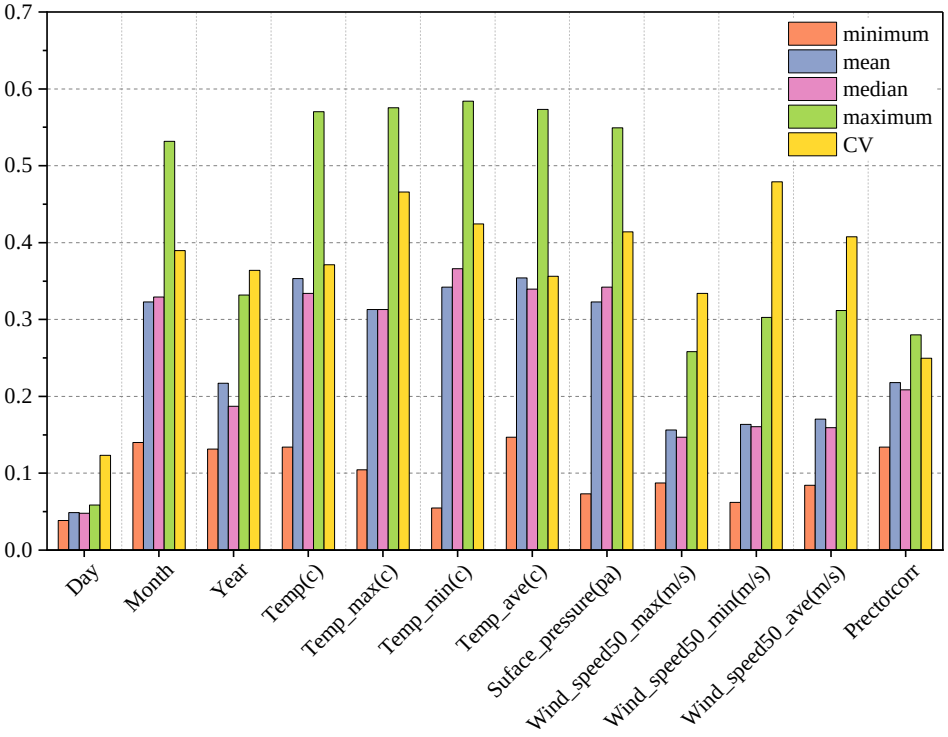

Figure 5: PAWN sensitivity analysis for predicting maximum generation

The results of the PAWN sensitivity analysis for predicting the maximum electricity generation (Fig. 5), demonstrate similar results as the total demand in Fig. 4. The variables related to the temperature, contribute more to the prediction values of prediction of maximum generation. Also, the month of the recording data has a higher impact on the outputs, rather than the variables such as recording day and year. Accordingly, the precipitation is more effective than the wind speed, to the maximum electricity generation.

3 Numerical analysis results

The results of the performance of the models in forecasting total electricity demand and maximum electricity generation are stated in the following. Furthermore, the error values of each hybrid model are compared and datasets division to test and train is noted.

3.1 Total demand forecasting results

The 4 hybrid models including CatBoost-PPSO, CatBoost-CGO, SVR-PPSO, and SVR-CGO are trained and tested through various input data, aiming to predict the electricity demand. Fig. 6 displays the comparison between the target value and predicted value accomplished by the hybrid models, along with the obtained error rates.

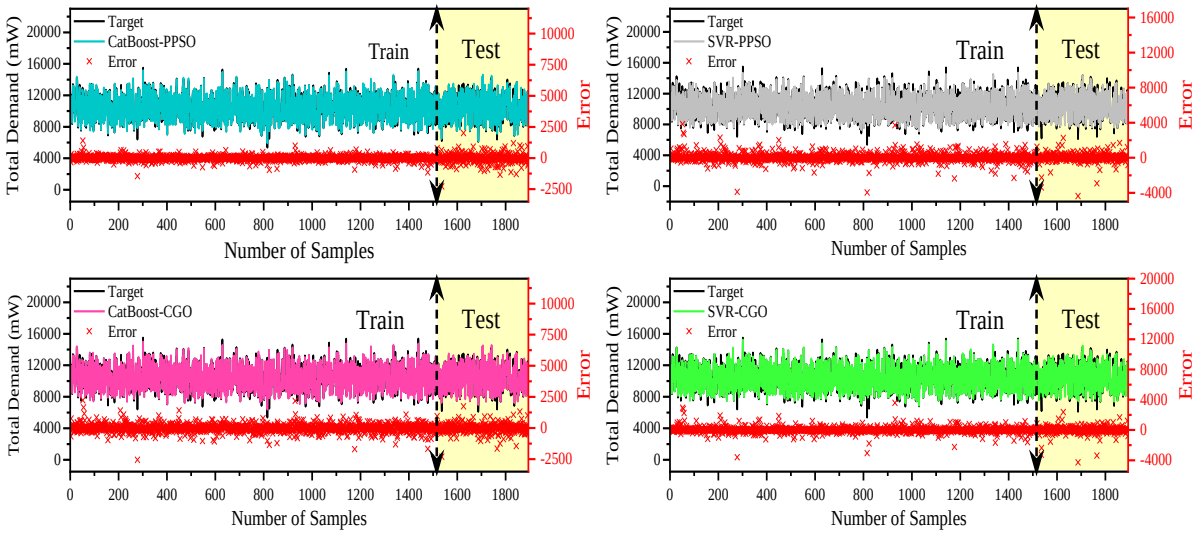

Figure 6: Comparison of target values and predicted values by hybrid models

Fig. 6 denotes how coincided are the prediction values of models are, with the expected values. This figure clearly distinguishes between the training and testing periods, and presents the predicted and actual electricity demand for all hybrid models. The result of the testing process shows that the prediction values of the CatBoost-PPSO are on the same wavelength as the target values, i.e., the CatBoost-PPSO performed more effective in forecasting the electricity demand, compared to the other hybrid models. The red curve represents the prediction error. the graphs of error rates indicate lower error values for CatBoost-PPSO for training samples, while according to Fig. 6, the error values of testing samples are almost the same (PPSO stands for Phasor Particle Swarm Optimizer, and CGO stands for Chaos Game Optimization).

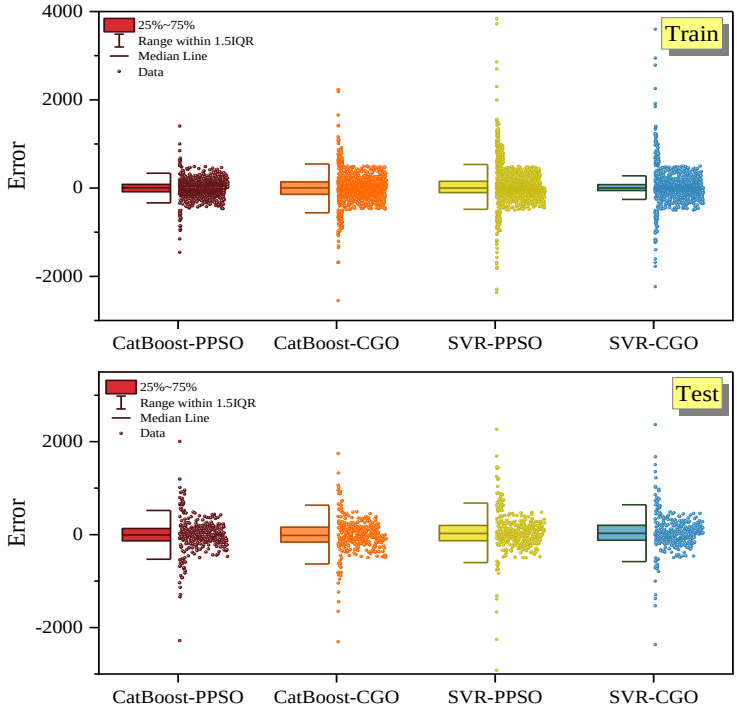

Figure 7: Range of error values for hybrid models on forecasting total demands

Fig. 7 demonstrates the dispersion of obtained error rates during the training and testing process. Accordingly, The CatBoost-PPSO reveals that the error rates of test samples are concentrated more tightly around 0, indicating more accurate forecasting by the model. CatBoost-CGO, SVR-PPSO, and SVR-CGO are in the next places, respectively. As can be seen in Fig. 7, although the SVR-CGO displays satisfying results in training, the error rates of this model in testing are comparatively higher than the other models.

The distribution of error values of each hybrid model is illustrated in more detail in Fig. 8:

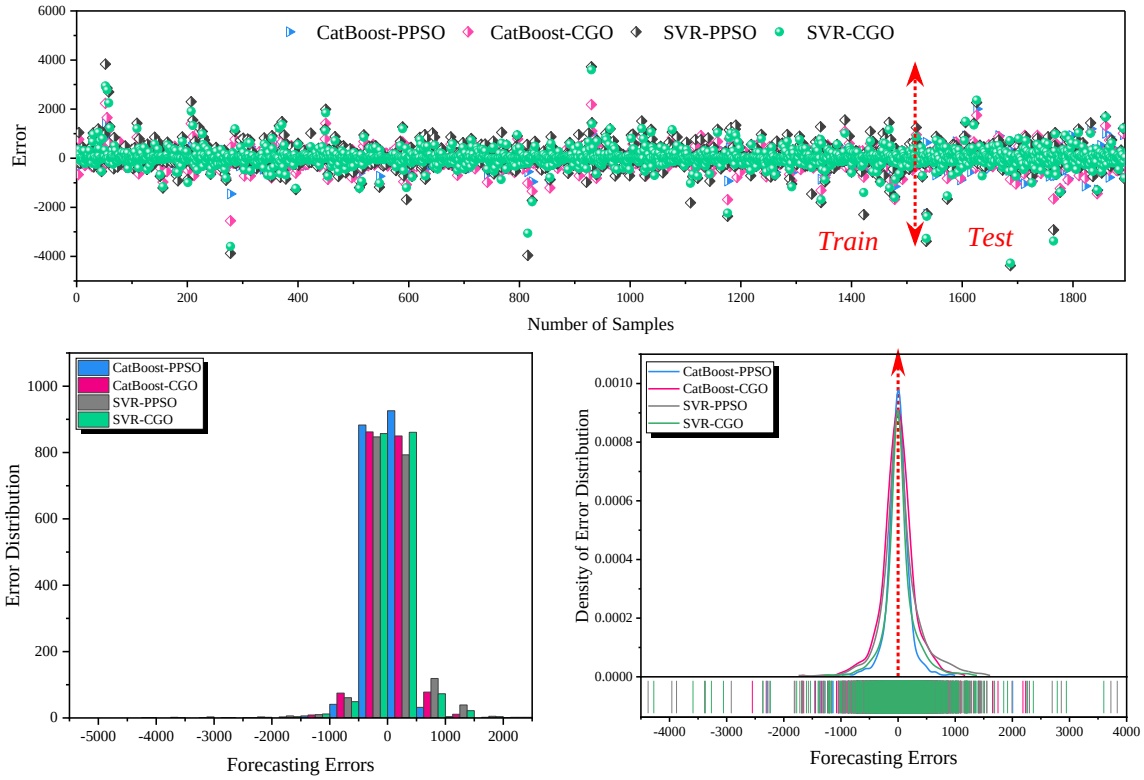

Figure 8: Comparison of error rates of the hybrid models on predicting total demand

The graphs in Fig. 8 demonstrate the comparison of the aforementioned error rates. Fig. 8 clarifies that the CatBoost-PPSO is more precise in prediction since its graph is more convergent to 0 and the obtained error values of samples are less dispersed.

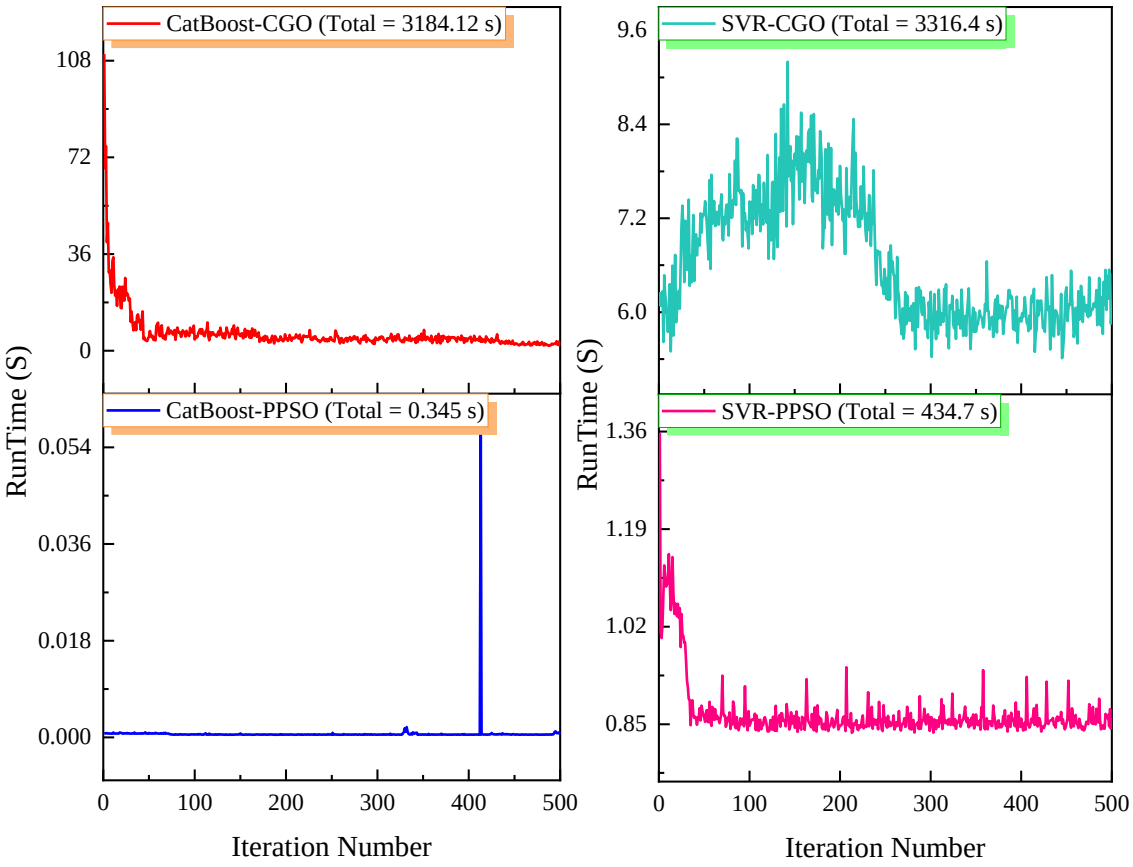

Figure 9: Comparing the runtime of the hybrid models during the prediction of total demand

According to Fig. 9, the CatBoost-PPSO performed the prediction with the shortest runtime. The models such as CatBoost-CGO and SVR-CGO, displayed disappointing results based on runtime. The runtime of the CatBoost-PPSO has remarkably outperformed the rest of the models, indicating the robustness and precision of the hybrid model.

3.2 Maximum electricity generation prediction results

The 4 aforementioned hybrid models have performed a prediction on maximum electricity generation where the results and error rates are presented in the following.

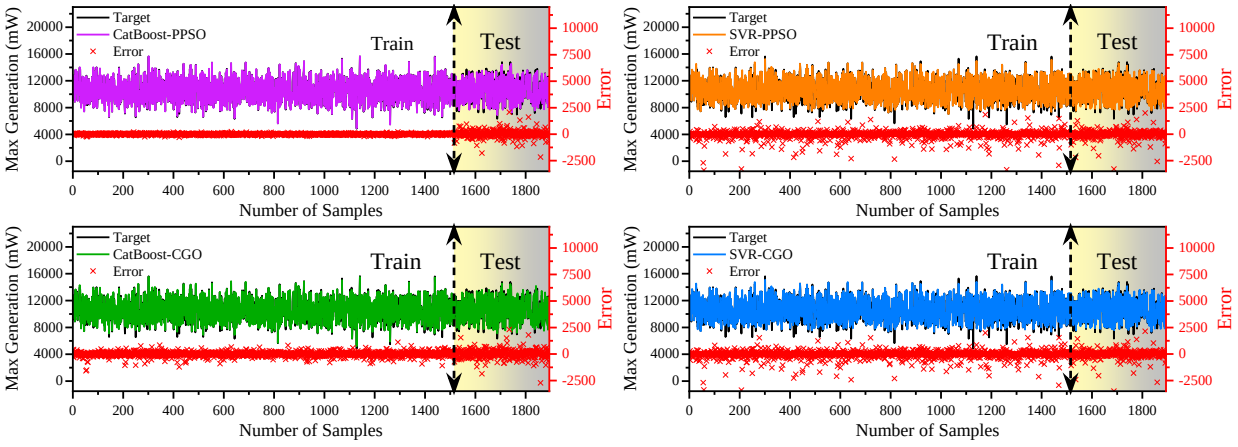

Figure 10: Comparison of target values and predicted values by hybrid models

The performance of the hybrid models in forecasting maximum generation is compared to the expected values and illustrated in Fig. 10. This visualization compares actual and predicted maximum electricity generation for different hybrid models, over both training and test datasets. Each subplot includes the target values, predicted outputs, and associated error plots. As can be seen, the CatBoost-PPSO shows notable results in predicting the training samples. Accordingly, the CatBoost-PPSO shows

comparatively lower error values during the testing process.

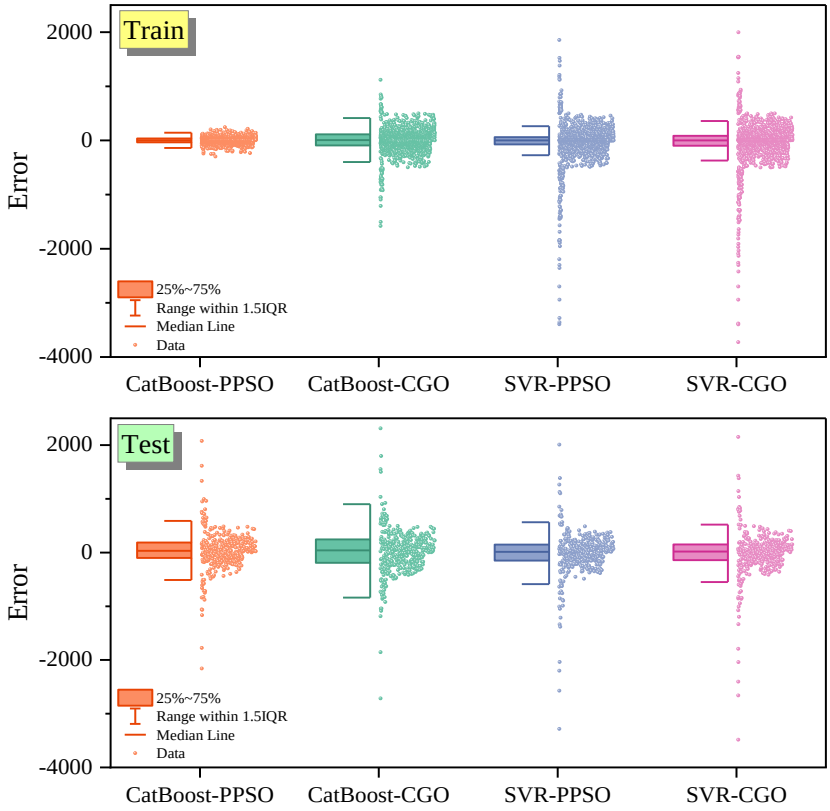

Figure 11: Range of error values for hybrid models on forecasting max generation

The dispersion of error rates of the hybrid models in Fig. 11, is indicating that the errors produced by CatBoost-PPSO during maximum generation forecasting, are so low. On the other hand, the error values of CatBoost-CGO during the test, are more dispersed and there are more outliers, indicating the low precision of CatBoost-CGO in forecasting max generation.

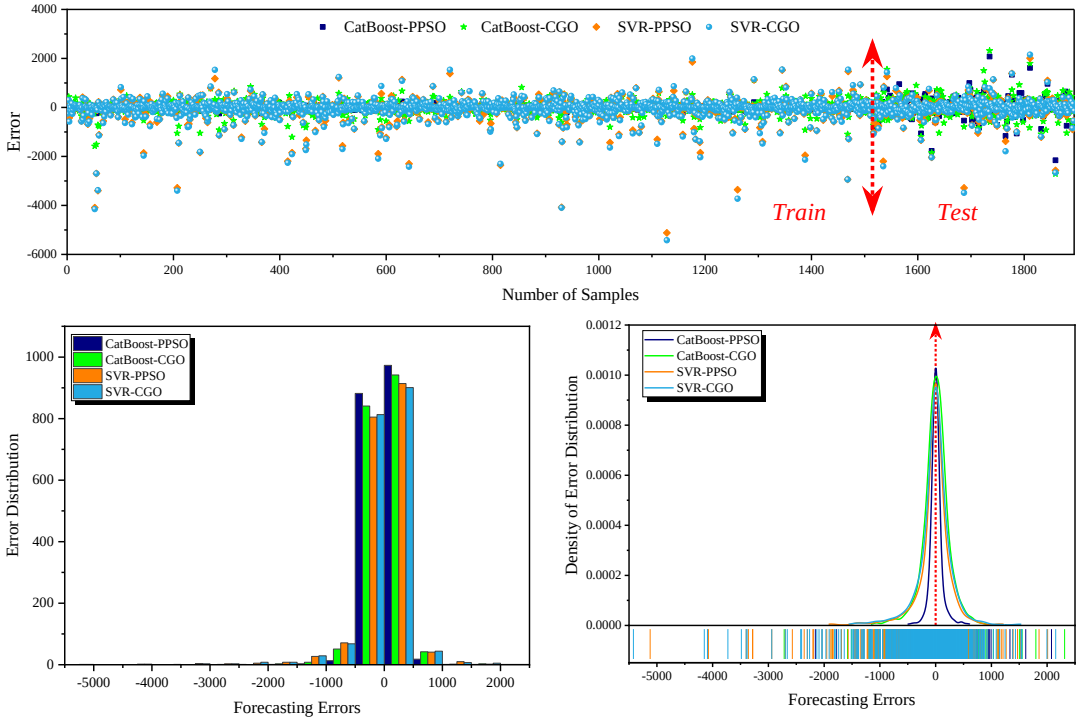

Figure 12: Comparison of error rates of the hybrid models on predicting max generation

Fig. 12 compares the distribution and density of obtained error values by the hybrid models in one graph. Evidently, the error values of CatBoost-PPSO are concentrated to 0 and less distributed, admitting the fact that the CatBoost-PPSO shows outstanding performance in forecasting the maximum electricity generation.

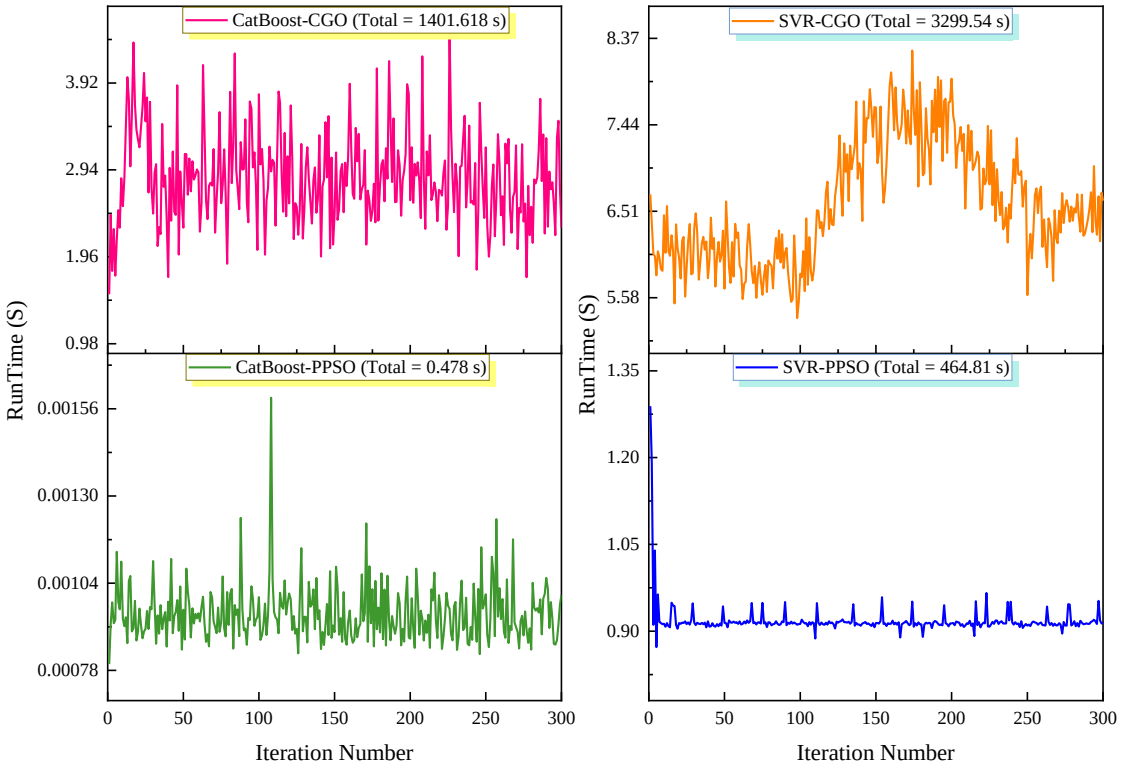

Figure 13: Comparing the runtime of the hybrid models during the prediction of max generation

Comparing the runtime of the models during the prediction of the maximum electricity generation denotes the best performance of the CatBoost-PPSO. Fig. 13 indicates that this hybrid model is significantly faster than the remaining models with a runtime of 0.478s after 300 iterations.

The models are analyzed according to several statistical evaluation indices the description of the metrics along with the results of the analysis are presented in the next section.

4 Analysis of the models based on the statistical evaluation indices

The hybrid ML-based models are assessed using several statistical evaluation indices. The accuracy of the models in predicting the total electricity demand and maximum electricity generation can be obtained using various metrics. Hence, the following indices are selected for this purpose. As shown in Table 3, the proposed statistical evaluation indices are detailed through specific equations.

Table 3: Equations of the proposed statistical evaluation indices

Statistics	Name	Equation
MSE	Mean Squared Error	$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
RMSE	Root Mean Square Error	$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}}$
NRMSE	Normalized Root Mean Square Error	$NRMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i)^2}}$
MAPE	Mean Absolute Percentage Error	$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left\|\frac{y_i - \hat{y}_i}{y_i}\right\|$
MAE	Mean Absolute Error	$MAE = \frac{\sum_{i=1}^{n}\|y_i - \hat{y}_i\|}{n}$
RAE	Relative Absolute Error	$RAE = \frac{\sum_{i=1}^{n}\|y_i - \hat{y}_i\|}{\sum_{i=1}^{n}\|y_i - \bar{y}\|}$
R2	Coefficient of Determination	$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$
n is the number of data points, y_i and $\hat{y}_i$ are i-th actual value and i-th estimated value, respectively, and $\bar{y}$ is the mean of the actual values.		

Each index, evaluates the performance of predictive models, particularly in regression analysis and forecasting. The metrics assessed the performance of the models in predicting total electricity demand and maximum electricity generation, separately.

4.1 Total demand forecasting results

To measure the difference between the predicted values of total demand and the real total demand, the R^2 is exerted. The fitness of the hybrid models to the data is illustrated in the following figure.

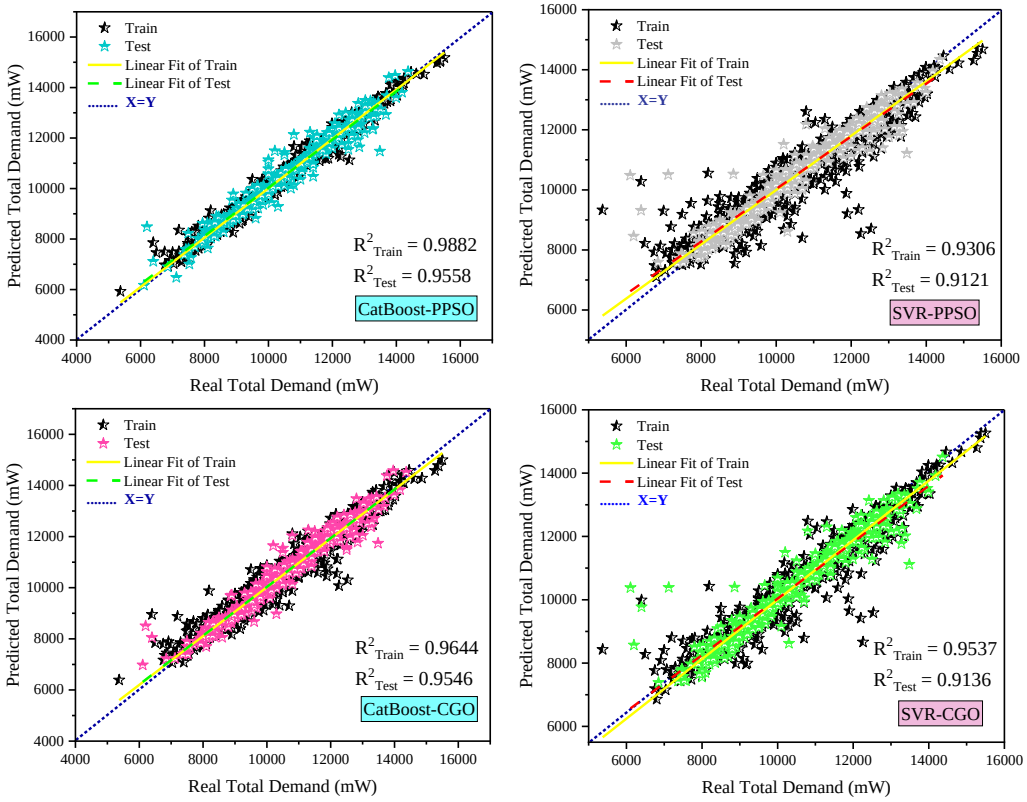

Figure 14: Comparing the goodness of fit of a model to the real data of total demand

The regression analysis using R2 in Fig. 14, indicates the promising result of CatBoost-PPSO for the data of test. The CatBoost-CGO also demonstrates satisfying results, denoting the distinction of CatBoost to the SVR in the given task.

The results of the evaluation based on MAE, MAPE, RAE, RMSE, and NRMSE are presented in Fig. 14. The obtained R2 values are also compared in a singular graph which is presented.

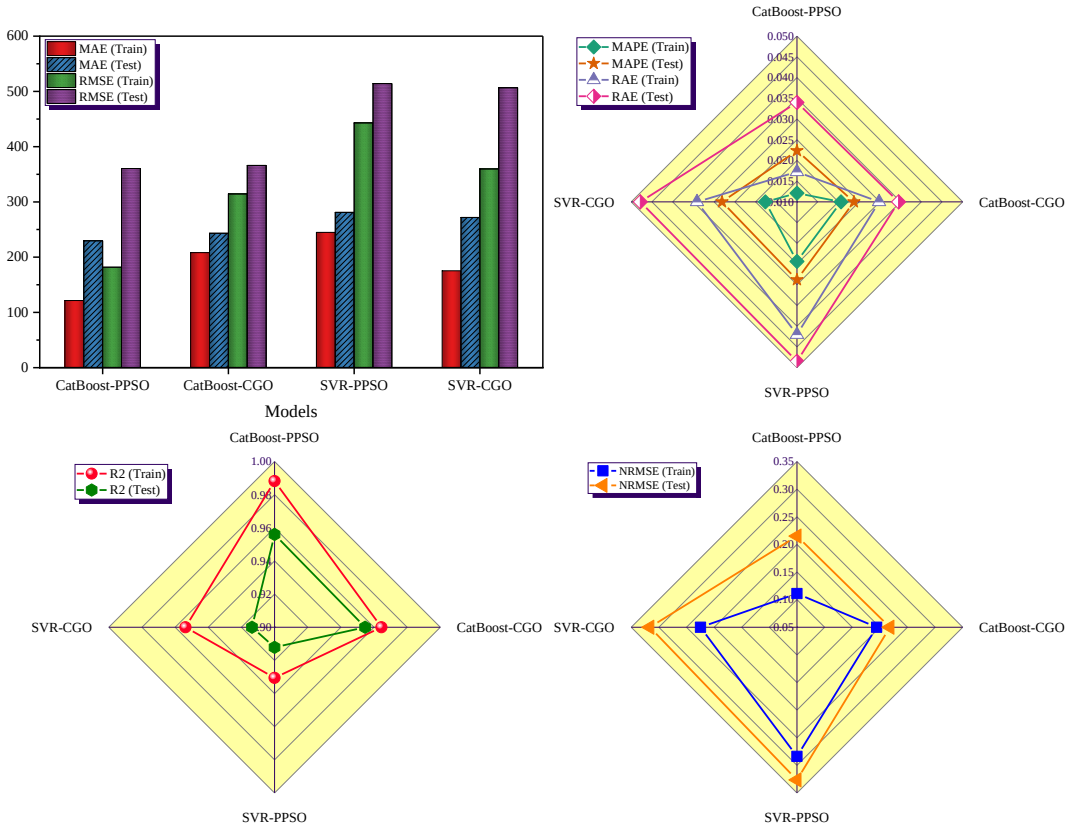

Figure 15: Comparing the obtained values of R2, MAE, MAPE, RAE, RMSE, and NRMSE for total demand forecasting

According to Fig. 15, the results of the testing process imply the low error rates and precise performance of the CatBoost compared to the SVR. Furthermore, the integration of CatBoost with PPSO outperforms the CatBoost-CGO, relatively. The CatBoost-PPSO exceeds the rest of the models in performance and accuracy.

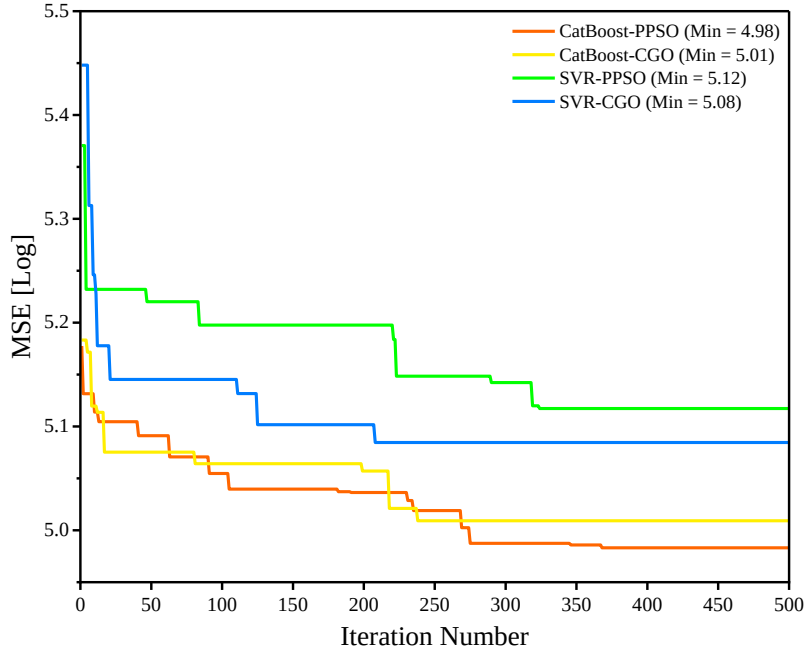

Figure 16: The obtained MSE values after 500 iterations in forecasting the total demand

The Fig. 16 shows the obtained MSE values after 500 epochs. According to the results, the CatBoost-PPSO displays fewer MSE values at earlier iterations. Hence, the CatBoost-PPSO model performs more beneficial in predicting the target values.

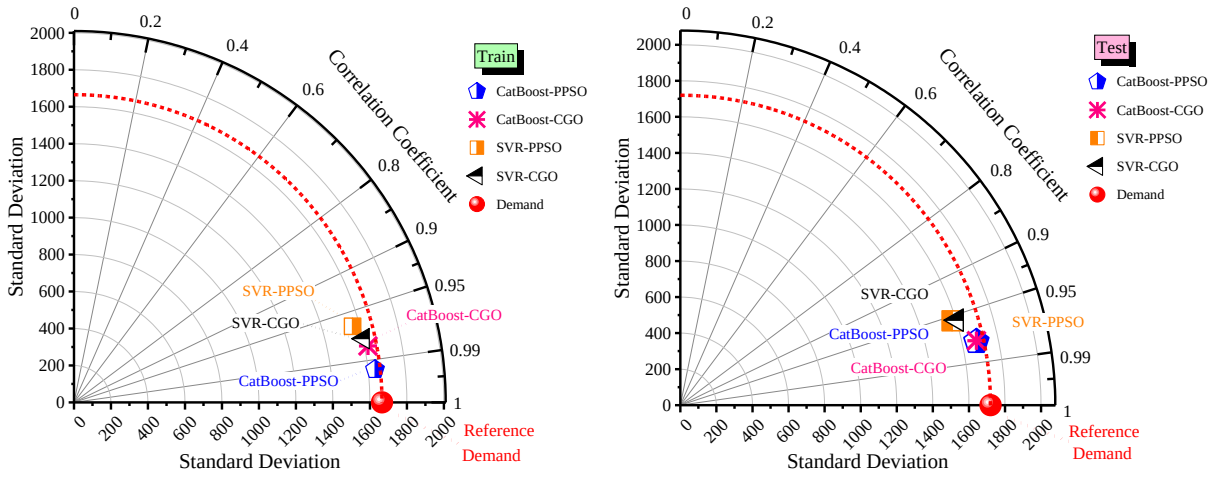

Figure 17: The Taylor diagrams of the hybrid models in predicting the total demand

Fig. 17, determines the correlation between the target value and the predicted values by the models. according to Fig. 17, the prediction values of the CatBoost-PPSO are closer to the line of target values (demand). This indicates the accuracy of the models in the prediction of the total electricity demand.

The obtained values of the evaluation indices are presented in Table 4, which summarizes the aforementioned figures.

Table 4: The obtained values of the statistical evaluation indices for predicting total demand

CatBoost-PPSO	Tain		Test		CatBoost-CGO	Tain		Test	
	MAE	121.4943	MAE	229.6541		MAE	208.2137	MAE	243.2856
	RMSE	181.7452	RMSE	360.4203		RMSE	314.3134	RMSE	366.0203
	MAPE	0.012091	MAPE	0.02234		MAPE	0.020638	MAPE	0.023736
	R2	0.988195	R2s	0.955997		R2	0.964452	R2s	0.954626
	RAE	0.017242	RAE	0.034001		RAE	0.029818	RAE	0.034529
	NRMSE	0.111016	NRMSE	0.214936		NRMSE	0.194235	NRMSE	0.218172
SVR-PPSO	Tain		Test		SVR-CGO	Tain		Test	
	MAE	244.8789	MAE	281.0936		MAE	175.2804	MAE	271.7475
	RMSE	443.0239	RMSE	514.0837		RMSE	359.6045	RMSE	506.6743
	MAPE	0.024369	MAPE	0.028811		MAPE	0.017614	MAPE	0.028079
	R2	0.930567	R2s	0.912098		R2	0.953707	R2s	0.913594
	RAE	0.042029	RAE	0.048497		RAE	0.034115	RAE	0.047798
	NRMSE	0.283801	NRMSE	0.326346		NRMSE	0.224504	NRMSE	0.316315

To assess the significance of model performance differences in the Total Demand objective, the Wilcoxon signed-rank test was applied using Absolute Error (AE) values. As shown in Table 5, the comparison between Catboost-PPSO and SVR-PPSO revealed a statistically significant advantage for Catboost-PPSO ($p = 0.0014$), indicating better predictive accuracy. Conversely, the comparison between Catboost-CGO and SVR-CGO did not result in a statistically significant difference ($p = 0.9496$), suggesting similar performance between the two models for this objective.

Table 5: Wilcoxon signed-rank test results for model comparison in total demand

Algorithm	Model 1	Model 2	Median Difference (AE_M1 - AE_M2)	P-value	Conclusion
PPSO	Catboost-PPSO	SVR-PPSO	-28.7344	0.0014	Catboost-PPSO Significantly Better (Lower AE)
CGO	Catboost-CGO	SVR-CGO	12.3338	0.9496	No Significant Difference

4.2 Maximum generation forecasting results

The foregoing statistical evaluation indices evaluate the maximum electricity generated along with the total demand. The results of the prediction accomplished with the hybrid models are compared based on the error rates and the difference between the prediction and the real values.

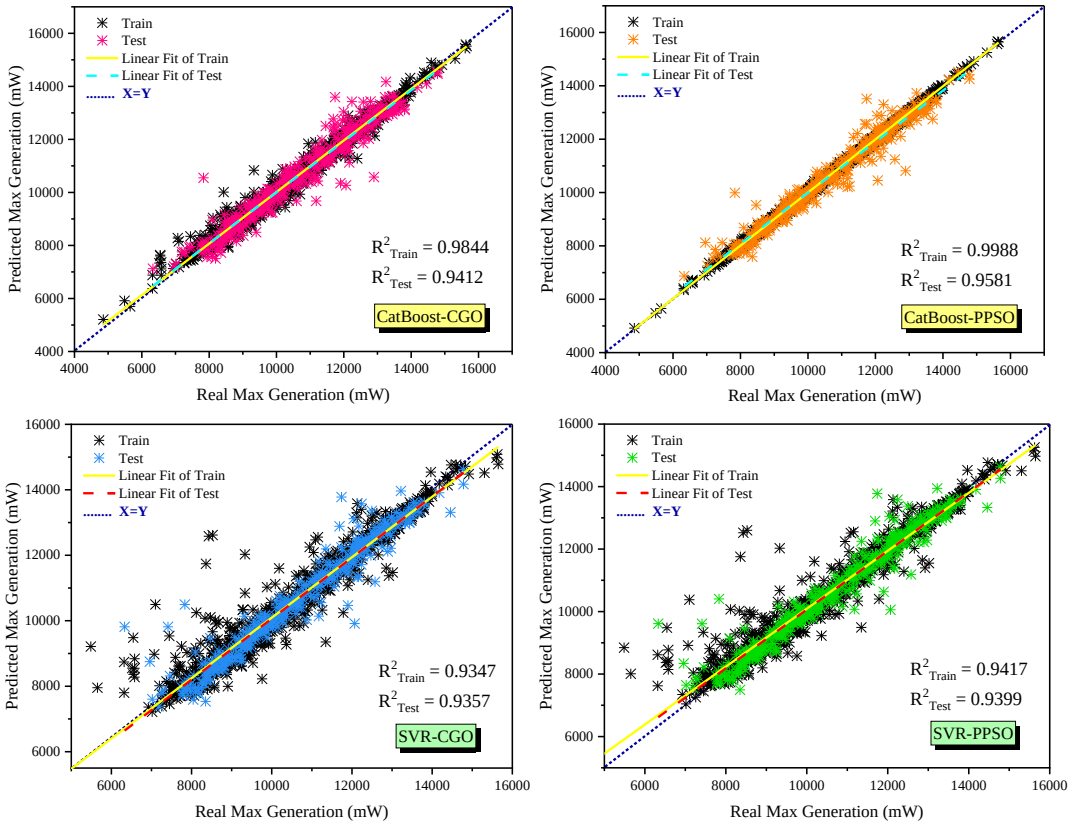

Figure 18: Comparing the goodness of fit of a model to the real data of the max generation

The regression graphs in Fig. 18 demonstrate that the predicted max generation by the CatBoost- PPSO is coincident with the real values with an R2 test value of 0.9581 which outperforms the other models.

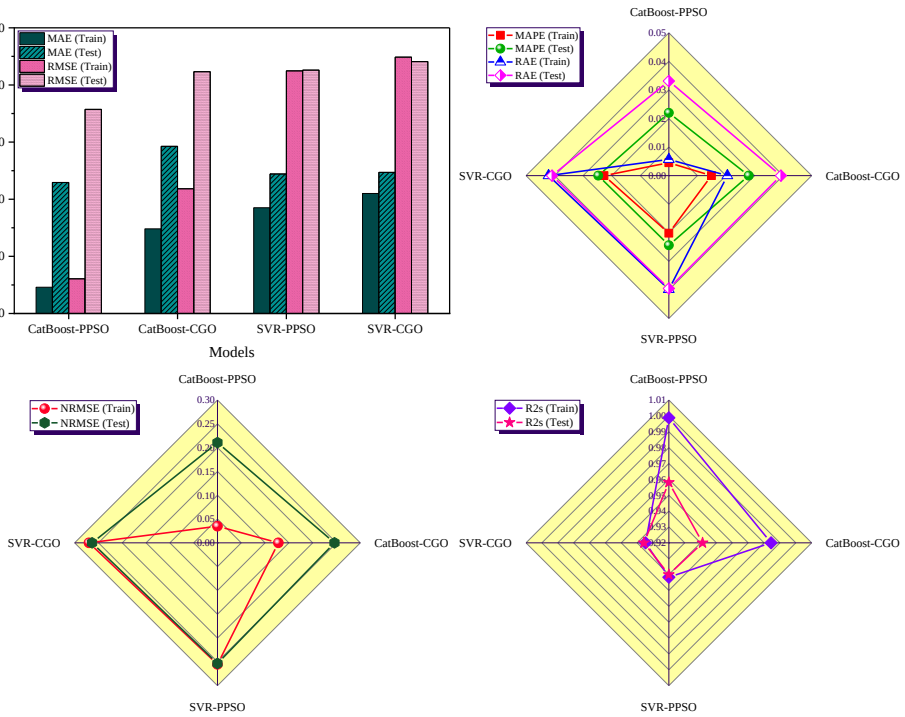

Figure 19: Comparing the obtained values of R2, MAE, MAPE, RAE, RMSE, and NRMSE for the max generation forecasting

By comparing the error values of the hybrid models illustrated in Fig. 19, the CatBoost-PPSO model has lower error values during the prediction of the maximum electricity generation. In addition to CatBoost-PPSO, the SVR-PPSO also showed promising results, particularly in terms of MAE. However, the R2 value for SVR-PPSO is relatively low.

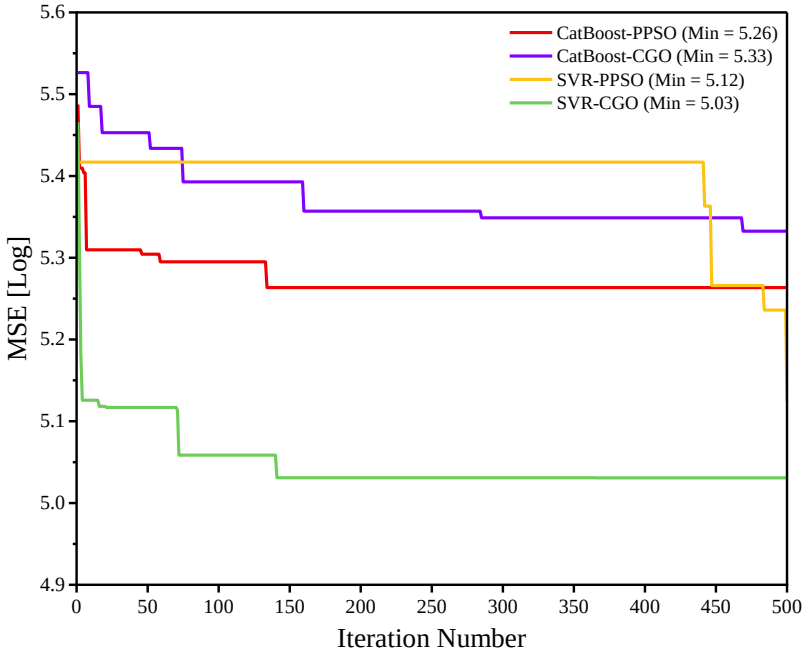

Figure 1: The obtained MSE values after 500 iterations in forecasting the max generation

By comparing the obtained MSE values in Fig. 20, the SVR-CGO displays notably lower MSE rates after 500 iterations. In addition, SVR-PPSO and then CatBoost-PPSO are in the next places.

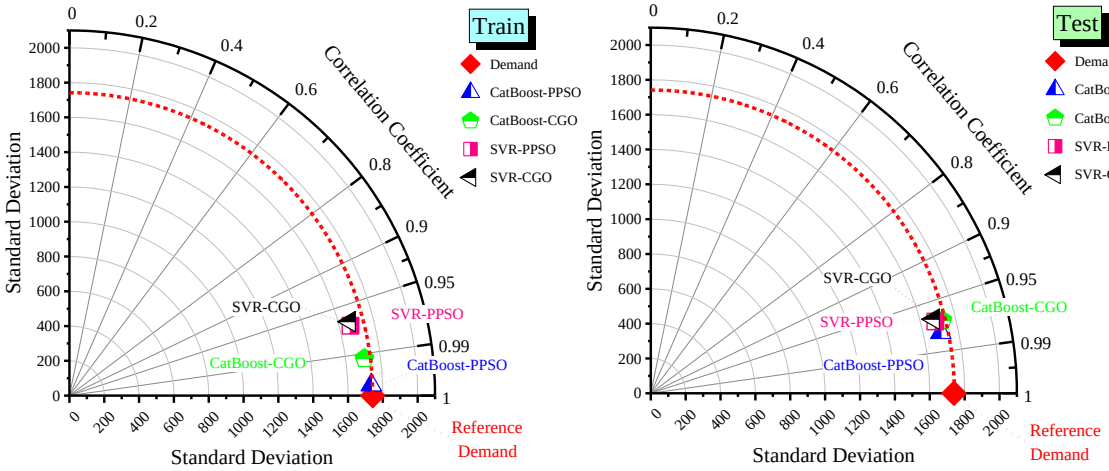

Figure 2: The Taylor diagrams of the hybrid models in predicting the max generation

According to Fig. 21, the standard deviation graphs denoting the performance of the models on predicting the max generation indicate better performance of the CatBoost-PPSO on data of train. The results of the testing process show that the models almost performed similarly and there is no noticeable difference, as can be deduced from the graph. Although, the numerical values in Table 5, denote better results for CatBoost-PPSO.

Table 5 presents the obtained values of the indices and summarizes the figures above:

Table 5: The obtained values of the statistical evaluation indices for predicting the maximum generation

CatBoost-PPSO	Tain		Test		CatBoost-CGO	Tain		Test	
	MAE	45.9977	MAE	229.3393		MAE	148.1471	MAE	292.5621
	RMSE	60.69213	RMSE	357.2295		RMSE	218.3039	RMSE	423.2141
	MAPE	0.004515	MAPE	0.02197		MAPE	0.014922	MAPE	0.02796
	R2s	0.998796	R2s	0.958087		R2s	0.984419	R2s	0.94124
	RAE	0.005696	RAE	0.033127		RAE	0.020489	RAE	0.039245

	NRMSE	0.034997	NRMSE	0.210455		NRMSE	0.127899	NRMSE	0.245781
SVR-PPSO	Tain		Test		SVR-CGO	Tain		Test	
	MAE	185.0795	MAE	244.2409		MAE	210.0944	MAE	247.0029
	RMSE	424.6411	RMSE	425.9695		RMSE	448.6983	RMSE	440.8853
	MAPE	0.020161	MAPE	0.024377		MAPE	0.022707	MAPE	0.024697
	R2	0.941689	R2s	0.940021		R2	0.934701	R2s	0.93571
	RAE	0.039855	RAE	0.039501		RAE	0.042112	RAE	0.040884
	NRMSE	0.25535	NRMSE	0.253392		NRMSE	0.269826	NRMSE	0.263338

Tables 3 and 4 indicate the substantial performance of the CatBoost-PPSO model, with the highest R^2 values on the test set and the lowest error metrics among the compared hybrid configurations. This superior performance can be attributed to the gradient boosting structure of CatBoost, which effectively handles complex feature interactions and minimizes overfitting, in conjunction with the exploration-exploitation balance provided by the phasor-enhanced swarm optimization mechanism of PPSO. The consistent reduction in RMSE and MAPE across both total demand and maximum generation forecasts further reinforces its robustness. In addition to predictive accuracy, runtime performance also plays a crucial role in practical deployments. Models with faster convergence and lower computational cost, such as CatBoost-PPSO, are more suitable for real-time or large-scale forecasting environments where rapid decision-making is essential. Hence, CatBoost-PPSO is selected as the preferred model for the given task, balancing both accuracy and applicability in real-world energy systems.

For the Max Generation objective, the Wilcoxon signed-rank test was conducted to evaluate the statistical differences in model performance. As shown in Table Y, SVR-CGO significantly outperformed Catboost-CGO ($p < 0.0001$), achieving lower AE values. However, in the PPSO-based comparison, no statistically significant difference was observed between Catboost-PPSO and SVR-PPSO ($p = 0.7646$).

Table 7: Wilcoxon signed-rank test results for model comparison in max generation

Algorithm	Model 1	Model 2	Median Difference (AE_M1 - AE_M2)	P-value	Conclusion
PPSO	Catboost-PPSO	SVR-PPSO	4.0609	0.7646	No Significant Difference
CGO	Catboost-CGO	SVR-CGO	48.8007	1.356e-06	SVR-CGO Significantly Better (Lower AE)

5 Discussion

The literature on electricity demand and generation forecasting reveals several recurring limitations. Many prior studies have relied on conventional machine learning or statistical methods (e.g., ANN, ARIMA, linear regression) without incorporating robust optimization techniques. Consequently, these models often suffer from limited adaptability to nonlinear demand patterns, lack of generalizability across contexts, and suboptimal parameter tuning resulting in reduced predictive accuracy. Furthermore, a substantial number of prior works failed to provide comprehensive statistical benchmarking using indices like R^2 or NRMSE, limiting comparability.

To address these shortcomings, we proposed four hybrid machine learning models that integrate CatBoost and Support Vector Regression (SVR) with advanced metaheuristic optimizers—Phasor Particle Swarm Optimizer (PPSO) and Chaos Game Optimizer (CGO). These hybridizations were designed to enhance model accuracy, stability, and generalization by leveraging both the learning capabilities of the base models and the exploratory efficiency of the optimizers. All experiments were conducted on a personal computing system running Windows 10 Pro (64-bit, build 19045.2311), equipped with a 12th Gen Intel® Core™ i5-12400F CPU @ 2.50 GHz and 32.0 GB DDR4 RAM. The experimental results, presented in Tables 4 and 5, strongly support the effectiveness of the proposed framework. Among all models, CatBoost-PPSO consistently delivered the best performance, achieving the lowest test RMSE values (360.42 for demand, 357.23 for generation) and the highest R^2 scores (0.956 for demand and 0.958 for generation). Notably, its test MAPE values (0.02234 for demand and 0.02197 for generation) were also significantly lower than those of all competing models. By comparison, even other hybrid combinations such as SVR-PPSO or CatBoost-CGO showed relatively higher error margins. For example, SVR-CGO exhibited a test RMSE of 506.67 for total demand and 440.88 for maximum generation, alongside lower R^2 values (0.913 and 0.935 respectively), confirming the superior synergy between CatBoost and PPSO.

In conclusion, the results indicate that the integration of CatBoost with PPSO not only mitigates the limitations of earlier models—such as underfitting, local optima, and high sensitivity to data noise—but also offers a more scalable and accurate solution for forecasting in complex, real-world electricity networks. Despite the results derived from the suggested hybrid models are, there are some limitations to consider. For one, the models were trained and validated using a defined dataset that is unique to a specific geographic and temporal context, limiting the applicability of the results to other areas or periods.

Second, while metaheuristic optimization methods like PPSO and CGO enhance performance, they are computationally intensive and will not perform well in real-time or high-frequency forecasting setups. Additionally, the models were evaluated in relation to benchmark statistical measures, but additional experiments in extreme demand settings or unusual events could provide more insight regarding robustness and pragmatic reliability. These findings underscore the value of hybrid optimization strategies and provide a strong foundation for future research in energy analytics.

6 Conclusion

Electricity demand and peak generation are some of the major factors that need to be forecasted, as this helps optimize resource allocation, infrastructure planning, and policymaking in energy management. The effective prediction, therefore, of utilities and policy-makers finds its basis in handling issues related to supply-demand imbalance, integration of renewable energy sources, and stability of the grid. Such forecasting is performed by using modern approaches like ML and metaheuristics algorithms that help in attaining better efficiency and robustness in energy systems for supporting sustainable energy transitions.

Precisely, this paper performed a deep forecast of total electricity demand and highest electricity generation using a hybrid approach by combining ML models with metaheuristic algorithms. More precisely, CatBoost and SVR have been combined with PPSO and CGO in order to build 4 hybrid models: namely Catboost-CGO, CatBoost-PPSO, SVR-CGO, and SVR-PPSO.

Further, performance evaluation in an extensive statistical basis using metrics like MSE, MAE, MAPE, RAE, RMSE, NRMSE, and R2 was performed.

It is clearly obvious that the hybrid CatBoost-PPSO model presents a higher predictive ability than the other examined hybrid models in all the considered indices. CatBoost-PPSO achieved the lowest test RMSE (360.42 for total demand and 357.23 for maximum generation), along with the highest R^2 scores (0.955997 and 0.958087, respectively), confirming its superior predictive performance over the other hybrid configurations. This is indicative of how effective the incorporation of CatBoost and PPSO will be in terms of forecasting electricity demand and generation. The obtained error values are considerably lower for the hybrid CatBoost-PPSO metrics.

These results not only underscore the importance of leveraging hybrid approaches but also highlight the significance of selecting appropriate combinations of ML models and metaheuristic algorithms for enhanced forecasting accuracy. Such insights could prove invaluable for practitioners and researchers in the field of energy forecasting, facilitating more precise decision-making and resource allocation in electricity management systems.

Nomenclature

Abbreviation	Description	Abbreviation	Description
SVR	Support Vector Regression	d	Dimension of seeds
SVM	Support Vector Machine	$x_i^j(0)$	The incipient position of each qualified seed
PSO	Particle Swarm Optimizer	$x_{i,\min}^j$	The minimum allowed value for the j-th decision
PPSO	Phasor Particle Swarm Optimizer	$x_{i,\max}^j$	The minimum allowed value for the j-th decision
CGO	Chaos Game optimizer	r	Randomly selected number
ML	Machine Learning	GB	Global best
NNGM	Neural Network-based grey forecasting model	MG_i	Average value
ANN	Artificial Neural Network	γ_i and β_i	Integer parameters for modeling the dice rolling probabilities
BMA	Bayesian Model Averaging	X_i	The chosen seed
LSTM	Long Short-Term Memory	R	Uniformly distributed random number
x_i	Input feature of SVR	v_{id}^f	The current velocity in PSO
y_i	Corresponding output of SVR	v_{id}^{f+1}	Updated velocity of the particle
w	Weight Factor	Pb_{id}^f	The best-known position of the particle so far
$\phi(x)$	Mapping function	Gb_{id}^f	Global best position of the particle
ε	epsilon-insensitive loss function	C_1 and C_2	Acceleration coefficients
b	Bias term	x_{id}^f	Current position of the particle
ξ_i and ξ_i^*	Slack variables	x_{id}^{f+1}	The updated position of the particle
C	Regularization parameter	θ	Phase angle of the PPSO
α_i and α_i^*	Lagrange multiplier	T	The current iteration

$K(x_i, x_j)$	Kernel function	PAWN	Probability of Absolute Worthlessness of Nodes
n	Total number of qualified seeds of CGO		

Acknowledgment

This work was supported by research and engineering applications of panoramic display and decision analysis technology for power grid fault relay protection based on artificial intelligence.

Authorship contribution statement

Jian Wang: Writing-Original draft preparation, Conceptualization, Supervision, Project administration.
Liandian Jiang: Methodology
Pengfei Huang: Validation.
Junyang Tian: Language review
Haiyong Li: Software
Bin Liu: Formal Analysis
Shan Liang: Data Curation

Conflicts of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Author Statement

The manuscript has been read and approved by all the authors, the requirements for authorship, as stated earlier in this document, have been met, and each author believes that the manuscript represents honest work.

Ethical approval

All authors have been personally and actively involved in substantial work leading to the paper, and will take public responsibility for its content.

References

[1] J. Tian and S. Cui, "Analysis and Calculation of Marginal Electricity Price of Nodes with Network Loss from the Perspective of Intelligent Robot Considering Digital Signal Processing Technology," *Informatica (Slovenia)*, vol. 48, no. 14, pp. 97–106, Oct. 2024, doi: 10.31449/inf.v48i14.6066.

[2] J. Zupančič, M. Gams, and J. Stefan, "Dynamic Protocol for the Demand Management of Heterogeneous Resources with Convex Cost Functions," 2017.

[3] Y. C. Hu, "Electricity consumption prediction using a neural-network-based grey forecasting approach," *Journal of the Operational Research Society*, vol. 68, no. 10, pp. 1259–1264, Oct. 2017, doi: 10.1057/s41274-016-0150-y.

[4] L. Ekonomou, "Greek long-term energy consumption prediction using artificial neural networks," *Energy*, vol. 35, no. 2, pp. 512–517, 2010, doi: 10.1016/j.energy.2009.10.018.

[5] K. Li, C. Hu, G. Liu, and W. Xue, "Building's electricity consumption prediction using optimized artificial neural networks and principal component analysis," *Energy Build*, vol. 108, pp. 106–113, Dec. 2015, doi: 10.1016/j.enbuild.2015.09.002.

[6] W. Zhang and J. Yang, "Forecasting natural gas consumption in China by Bayesian Model Averaging," *Energy Reports*, vol. 1, pp. 216–220, Nov. 2015, doi: 10.1016/j.egyr.2015.11.001.

[7] N. Fumo and M. A. Rafe Biswas, "Regression analysis for prediction of residential energy consumption," 2015, *Elsevier Ltd*. doi: 10.1016/j.rser.2015.03.035.

[8] S. Barak and S. S. Sadegh, "Forecasting energy consumption using ensemble ARIMA-ANFIS hybrid algorithm," *International Journal of Electrical Power and Energy Systems*, vol. 82, pp. 92–104, Nov. 2016, doi: 10.1016/j.ijepes.2016.03.012.

[9] J. Hwang, D. Suh, and M. O. Otto, "Forecasting electricity consumption in commercial buildings using a machine learning approach," *Energies (Basel)*, vol. 13, no. 22, Nov. 2020, doi: 10.3390/en13225885.

[10] R. Wang, S. Lu, and W. Feng, "A novel improved model for building energy consumption prediction based on model integration," *Appl Energy*, vol. 262, Mar. 2020, doi: 10.1016/j.apenergy.2020.114561.

[11] Z. Lin, L. Cheng, and G. Huang, "Electricity consumption prediction based on LSTM with attention mechanism," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 15, no. 4, pp. 556–562, Apr. 2020, doi: 10.1002/tee.23088.

[12] G. Vijendar Reddy, L. J. Aitha, C. Poojitha, A. Naga Shreya, D. Krithika Reddy, and G. Sai Meghana, "Electricity Consumption Prediction Using Machine Learning," in *E3S Web of Conferences*, EDP Sciences, Jun. 2023. doi: 10.1051/e3sconf/202339101048.

[13] R. Olu-Ajayi and H. Alaka, "Building energy consumption prediction using deep learning."

[14] A. V. Dorogush, V. Ershov, and A. Gulin, "CatBoost: gradient boosting with categorical features support," Oct. 2018, [Online]. Available: http://arxiv.org/abs/1810.11363

[15] G. Huang *et al.*, "Evaluation of CatBoost method for prediction of reference evapotranspiration in humid regions," *J Hydrol (Amst)*, vol. 574, pp. 1029–1041, Jul. 2019, doi: 10.1016/j.jhydrol.2019.04.085.

[16] H. Drucker·, C. J. C. Burges, L. Kaufman, A. Smola··, and V. Vapoik, "Support Vector Regression Machines."

[17] B. Üstün, W. J. Melssen, and L. M. C. Buydens, "Facilitating the application of Support Vector Regression by using a universal Pearson VII function-based kernel," *Chemometrics and Intelligent Laboratory Systems*, vol. 81, no. 1, pp. 29–40, Mar. 2006, doi: 10.1016/j.chemolab.2005.09.003.

[18] S. Talatahari and M. Azizi, "Chaos Game Optimization: a novel metaheuristic algorithm," *Artif Intell Rev*, vol. 54, no. 2, pp. 917–1004, Feb. 2021, doi: 10.1007/s10462-020-09867-w.

[19] A. Ramadan, S. Kamel, M. M. Hussein, and M. H. Hassan, "A new application of chaos game optimization algorithm for parameters extraction of three diode photovoltaic model," *IEEE Access*, vol. 9, pp. 51582–51594, 2021, doi: 10.1109/ACCESS.2021.3069939.

[20] J. Kennedy, R. Eberhart, and bls gov, "Particle Swarm Optimization."

[21] D. N. Wilke, M. Khajehzadeh, and Q. Bai, "Analysis of Particle Swarm Optimization Algorithm Swarm Int elligence based Soft Comput ing Techniques for t he Solut ions t o Mult iobject ive Op… hifza afaq A Survey of t he St at e of t he Art in Part icle Swarm Opt imizat ion A Review of Populat ion-based Met a-Heurist ic Algorit hms Analysis of Particle Swarm Optimization Algorithm."

[22] M. Ghasemi, E. Akbari, A. Rahimnejad, S. E. Razavi, S. Ghavidel, and L. Li, "Phasor particle swarm optimization: a simple and efficient variant of PSO," *Soft comput*, vol. 23, no. 19, pp. 9701–9718, Oct. 2019, doi: 10.1007/s00500-018-3536-8.

[23] khatsu Siseyiekuo, Srivastava Abhishek, and Kumar Das Dushimanta, "An Adaptive Phasor Particle Swarm Optimization to Solve Economic Load Dispatch and Combined Emission Economic Load Dispatch Problem," 2019.

[24] F. Pianosi and T. Wagener, "A simple and efficient method for global sensitivity analysis based oncumulative distribution functions," *Environmental Modelling and Software*, vol. 67, pp. 1–11, May 2015, doi: 10.1016/j.envsoft.2015.01.004.

[25] A. Puy, S. Lo Piano, and A. Saltelli, "A sensitivity analysis of the PAWN sensitivity index," *Environmental Modelling and Software*, vol. 127, May 2020, doi: 10.1016/j.envsoft.2020.104679.

[26] I. Salehin, S. M. Noman, and M. M. Hasan, "Electricity energy dataset 'BanE-16': Analysis of peak energy demand with environmental variables for machine learning forecasting," *Data Brief*, vol. 52, Feb. 2024, doi: 10.1016/j.dib.2023.109967.