Appraising the Building Cooling Load via Hybrid Framework of Machine Learning Techniques

Longlong Yue[1], Xiangli Liu[2]*, Shiliang Chang[3]
Linzhou Vocational and Technical College of Architecture, Linzhou, Henan 456500 China[1, 3]
Linzhou Audit Bureau, Linzhou, Henan 456500 China[2]

***Abstract*—The overarching objective of this study lies in the thorough evaluation of the effectiveness of K-nearest neighbors (KNN) models in the precise estimation of building cooling load consumption. This assessment holds significant importance as it pertains to the feasibility and reliability of implementing machine learning techniques, particularly the KNN algorithm, within the domain of building energy management. This evaluation process centers on scrutinizing five distinct spatial metrics closely associated with the KNN algorithm. To refine and enhance the algorithm's predictive capabilities, this endeavor incorporates utilizing test samples drawn from an extensive database. These test samples serve as valuable resources for augmenting the overall predictive accuracy of the model, ultimately leading to more robust and reliable predictions of cooling load consumption within the building systems. Ultimately, the research endeavors to contribute substantially to advancing more energy-efficient and automated cooling system control strategies. Developed models encompass a single base model, another model optimized through the application of African Vultures Optimization, and a third model optimized using the Sand Cat Swarm Optimization technique. The training dataset includes 70% of the data, with eight input variables relating to the geometric and glazing characteristics of the buildings. After validating 15% of the dataset, the performance of the remaining 15% is tested. An analysis of various evaluation metrics reveals that KNSC (K-Nearest Neighbors optimized with the Sand Cat Swarm Optimization) demonstrates remarkable accuracy and stability among the three candidate models. It achieves a substantial reduction in the prediction Root Mean Square Error (RMSE) of 32.8% and 21.5% in comparison to the other two models (KNN and KNAV) and attains a maximum R^2 value of 0.985 for cooling load prediction.**

Keywords*—K-nearest-neighbors; machine learning; cooling load prediction; African Vultures Optimization; Sand Cat Swarm Optimization

I. Introduction

A. Background

Amid growing apprehension about rising CO_2 emissions, there has been a substantial increase in energy consumption by buildings [1–3]. Numerous endeavors have been made to curtail or enhance the efficiency of building energy consumption [4]. Concentrating on passive and active design strategies, many research studies have conducted investigations aimed at augmenting energy efficiency in buildings. These efforts have encompassed initiatives to facilitate the thermal properties of building envelopes, upgrade mechanical systems to advanced technologies, and integrate renewable energy systems [5–8]. Although these design strategies have been instrumental in managing building energy, their impact on reducing building energy consumption ranges from 3% to 10% of the total energy consumption in buildings [9]. Furthermore, enhanced energy performance can be anticipated only when these design strategies are integrated during the initial phases of building design [10,11].

The swift advancement in information and communication technologies has made building energy management and prediction essential for enhancing energy efficiency and diminishing building energy consumption [12]. The incorporation of metering technologies has rendered specific data on building operations and energy consumption readily accessible, facilitating comprehensive analyses of energy consumption patterns within buildings [13]. The copious datasets gathered by specific systems, such as building automation systems and building energy management systems, offer the potential to anticipate the dynamic interplay among them, thereby influencing building energy consumption [14,15].

A crucial direction for future research involves evaluating the efficacy of these sophisticated statistical models in forecasting actual building energy performance rather than exclusively depending on simulated results. Previous studies have unveiled notable disparities between initial design simulations and real energy consumption estimations, largely arising from uncertainties linked to modeling assumptions, construction quality, weather fluctuations, operational procedures, and occupant behaviour [16]. Given the increasing accessibility of data concerning real energy consumption, a wealth of prospects exist for harnessing advanced methodologies to explore the intricate relationship between building attributes, occupant behavior, and actual energy performance. This can be achieved through the scrutiny of extensive datasets [17–19].

B. Related Work

In studies [20], [21], [22], [23] and [24–30] have focused on applying different machine learning techniques to predict building energy loads. For instance, Lin et al. [31] introduced a short-term load forecasting method using data-driven techniques to anticipate the cooling load of buildings. Their method demonstrated high precision and efficacy through rigorous evaluation, underscoring the potential of data-driven approaches for accurate cooling load forecasts. Wang et al. [32] advanced the Enhanced Harris Hawk Optimization (EHO) by developing the Improved EHO (IEHO) neural network. Their findings revealed that integrating the Back Propagation (BP) neural

network with the IEHO, forming the IEHO-BP neural network model, significantly improved the accuracy of heating and cooling load predictions. This hybrid neural network model exhibited superior robustness and precision, indicating its effectiveness for load forecasting applications. Leiprecht et al. [33] conducted a comprehensive analysis of autoregressive forecasting methods and decision trees, including adaptive boosting, for thermal load prediction. They also explored deep learning techniques such as Long Short-Term Memory (LSTM) neural networks, demonstrating the versatility and effectiveness of deep learning models in thermal load prediction. Jihad and Tahiri [34] employed Artificial Neural Networks (ANN) to forecast the energy requirements of residential structures. Their results showed high accuracy rates, with 98.7% accuracy for training data and 97.6% for test data, highlighting the ANN's capability for precise energy load predictions. Machine learning prediction has emerged as a powerful and versatile approach, recognized across various domains for its ability to analyze data, identify patterns, and make predictive decisions without explicit programming. Machine learning models, including decision trees, support vector machines, and neural networks, can train on extensive datasets, enabling the identification of intricate data relationships. These models play a pivotal role in transforming data analysis and interpretation methodologies, providing predictive insights that support informed decision-making in business and organizational contexts [20, 21]. In civil engineering, two illustrative examples showcase the application of machine learning techniques. Moradzadeh and Mohammadi-Ivatloo [22] developed an enhanced hybrid machine-learning model to predict cooling and heating loads in residential buildings. Their study involved a comprehensive analysis of diverse forecasting models, highlighting the effectiveness of hybrid models in load prediction. RC Zhao et al. [23] detailed an approach that decomposed temporal features and climatic attributes into multiple independent components. By diversifying the feature set available for model training, they provided a more nuanced depiction of pedestrian flow's impact on a building's cooling load, thereby enhancing prediction accuracy.

C. Objective

In addressing the challenges related to cooling load predictions, this study endeavors to develop a cooling load prediction model capable of application across different load modes throughout the entire cooling season. Utilizing a machine learning approach, this research exposes the potential bias in strategy guidance due to prediction inaccuracies, as evidenced by an evaluation index with practical significance, while concurrently constructing and validating the cooling load prediction model. This ongoing study draws insights from previous successful applications of KNN models in predicting cooling load for buildings, with a distinguishing feature being the incorporation of diverse datasets encompassing a broad spectrum of input variables about building geometry and glazing characteristics sourced from existing literature. The predictive capabilities of a single KNN model were rigorously assessed, and to enhance the training process, two distinct optimizers, the African Vultures Optimization (AVO) and the Sand Cat Swarm Optimization (SCSO), were introduced. Comprehensive performance evaluations of the three models, employing various metrics such as R^2, RMSE, MSE, RAE, and PI were conducted to identify the most effective hybrid model for building cooling load prediction. This study makes several significant contributions to building energy management by developing a versatile cooling load prediction model applicable across different load modes throughout the cooling season. Utilizing a machine learning approach, it addresses potential biases in strategy guidance due to prediction inaccuracies and uniquely incorporates diverse datasets with a wide range of input variables related to building geometry and glazing characteristics. The introduction of advanced optimizers, AVO and SCSO, enhances the training process of the KNN models. These findings contribute to more energy-efficient and automated cooling system control strategies, advancing sustainability in building operations.

II. Materials and Methods

A. Data Collection

Ensuring data integrity is pivotal to the methodology of this study. The dataset employed for training, the intelligent models was derived from previous research [35, 36], offering critical information necessary for the implementation and assessment of the proposed techniques in predicting building cooling loads. This dataset consists of 768 samples, each encompassing eight key input parameters: relative compactness (RC), surface area (SA), wall area (WA), roof area (RA), overall height (OH), orientation (Or), glazing area (GA), and glazing area distribution (GAD). These parameters are vital for accurate model training and evaluation. Table I comprehensively summarizes the key criteria used for statistical analysis, including data averages, standard deviations, skew, median, minimum, and maximum values. The resulting output values span a considerable range, with the minimum recorded value being 10.9 and the maximum reaching 48.03. Notably, the average value for cooling stands at 24.587. This value, representing the central tendency of the data, underscores the substantial nature of the cooling load measurements and emphasizes their significance within the scope of the research.

B. Overview of Machine Learning Method and Optimizers

1) K-Nearest Neighbor (KNN): The K-nearest neighbor (KNN) method is well-known for its simplicity, effectiveness, and ease of use [37]. KNN is versatile and can be employed for classification and regression tasks, sharing similarities with other methods such as artificial neural networks (ANN) and random forests (RF). The adoption of this technique comes with several benefits:

a) It is simple and easily understandable, rendering it suitable for practical application.

b) When applied in classification and regression tasks, it can learn non-linear decision boundaries, enhancing its versatility through the flexibility to adjust the K value for defining these boundaries.

c) In contrast to certain other algorithms, KNN does not necessitate a dedicated training phase.

d) The method relies on a single hyperparameter, denoted as K , which streamlines the fine-tuning of other hyperparameters.

TABLE I. THE STATISTICAL PROPERTIES OF THE VARIABLES

Variables		Indicators					
	Category	*Min*	*Max*	*Median*	*Avg*	*Skew.*	*St. Dev.*
RC	Input	0.62	0.98	0.75	0.764	0.496	0.106
SA	Input	514.5	808.5	673.75	671.70	-0.125	88.086
WA	Input	245	416.5	318.5	318.5	0.533	43.63
RA	Input	110.25	220.5	183.75	176.60	-0.163	45.165
OH	Input	3.5	7	5.25	5.25	-2.9E-19	1.751
Or	Input	2	5	3.5	3.5	1.45E-19	1.118
GA	Input	0	0.4	0.234	0.235	-0.060	0.133
GAD	Input	0	5	2.812	2.813	-0.089	1.55
Cooling	Output	10.9	48.03	24.588	24.587	0.396	9.51

The core concept of KNN involves pinpointing a group of K samples, typically determined through a distance function, that displays closeness to unknown samples in the training dataset. This process entails the recognition of clusters of resembling samples. Following this, KNN computes the mean of response variables and contrasts the outcomes with those obtained from a set of K samples to establish the classes of unidentified samples [38]. Hence, the KNN algorithm's choice of the K value plays a vital role in determining its efficacy [39]. For this objective, three distance functions are employed in the context of regression tasks to calculate the distances between adjacent data points, as denoted by Eq. (1) to Eq. (3):

$$F(Eu) = \sqrt{\sum_{i=0}^{f}(x_i - y_i)^2} \quad (1)$$

$$F(Ma) = \sum_{i=0}^{f}|x_i - y_i| \quad (2)$$

$$F(Mi) = \left(\sum_{i=0}^{f}(|x_i - y_i|)^d\right)^{\frac{1}{d}} \quad (3)$$

Within this framework, $F(Eu)$ corresponds to the Euclidean distance function, $F(Ma)$ signifies the Manhattan distance function, and $F(Mi)$ stands as the Minkowski distance function. The variables x_i and y_i are specifically associated with the ith dimension of data points x and y, while d is utilized as an order parameter in the calculation of distances between these points.

2) African Vultures Optimization (AVO): The African vulture optimization algorithm was introduced in a study by [40]. In the quest to identify the most proficient vultures in each category, the proposed solutions within the initial population undergo an initial assessment for suitability. The top-performing solution is the optimal choice for the group, both in the initial and subsequent iterations. It's worth emphasizing that the fitness of all populations requires periodic reassessment in each iteration. Furthermore, the remaining solutions are determined using the following approach:

$$G(i) = \begin{cases} Bestvulture_i \; if \; h_i = a \\ Bestvulture_2 \; if \; h_i = b \end{cases} \quad (4)$$

Both a and b fall within the interval of (0, 1).

Applying a roulette wheel approach is a method utilized to select a potential optimal solution. This technique provides a systematic means to pinpoint the most appropriate solution, and the process is elucidated as follows:

$$h_i = \frac{k_i}{\sum_{i=1}^{n} k_i} \quad (5)$$

In cases where b is smaller than a, implementing the AVOA may lead to a potential increase in degradation. Conversely, even when a is less than b, the AVOA could yield varying results. To transition from the exploration stage to the exploitation stage, Eq. (6) is utilized:

$$K = (2 \times rand_1 + 1) \times y \times \left(1 - \frac{Iter_i}{Max_{Iter}}\right) \quad (6)$$

B stands for the hunger level.

$Iter$ signifies the presence of multiple iterations.

$rand_1$ and y denote random numbers generated in $[0 - 1]$.

Max_{Iter} represents an integer value that indicates the maximum number of iterations.

When K falls within the range of values greater than 1 but less than 1, the African vulture optimization algorithm commences the search phase. In contrast, if K is less than 1, the AVOA algorithm transitions to the exploitation phase, resembling the behavior of a vulture scavenging for nearby food.

During the exploration phase in AVOA, the vulture employs two techniques to explore distinct regions. If the random number generated by rand_{h_1} is greater than or equal to the h_1 parameter, it opts for Eq. (7) (a). Conversely, if the random number produced by $rand_{h_1}$ is less than the h_1 parameter, it selects Eq. (8). The vulture's movement during this phase can be elucidated as follows:

$$V(i+1) = \begin{cases} G(i) - Q(i) \times \text{K} \\ if \; h_1 \geq rand_{h_1}, \; (a) \\ G(i) - K + rand_2\big((uc - lc) \times rand_3 + lc\big) \\ if \; h_1 < rand_{h_1}, \; (b) \end{cases} \quad (7)$$

$$Q(i) = |X \times G(i) - V(i)| \quad (8)$$

$V(i)$ indicates the current vector denoting the vulture's position.

$V(i+1)$ represents the vector signifying the vulture's position in the subsequent iteration.

K stands for the level of satisfaction or contentment among the vultures.

$uc\ and\ lc$ refer to the upper and lower boundaries or limits of the variable, respectively.

$rand$ represents a random number falling within the range of 0 to 1.

X symbolizes the unpredictable or random movement executed by the leading vulture.

Introducing more randomness is achieved through the utilization of $rand_2$. This results in an elevated degree of unpredictability at the environmental level, fostering diversity and safeguarding distinctive attributes across various domains.

When K falls below 1 in the AVOA algorithm, it transitions into an exploitation phase comprising two segments, each featuring two unique procedures. The choice of which procedure to employ within each segment is made in a deterministic manner, depending on the parameters h_2 and h_3. Two distinct rotation flight procedures are executed in the initial segment to avoid conflicts. Furthermore, h_2 determines the selection rate for each strategy; if $rand_{h_2}$ is greater than or equal to h_2, it carries out the stall and outbound strategy, whereas if the random number is less than the h_2 parameter, it opts for the rotational flight process.

$$V(i+1) = \begin{cases} Q(i) \times (K + rand_4) - c(t) \\ if\ h_3 \geq rand_{\mathrm{h2}}\ \ (a) \\ G(i) - V(i) \\ if\ \ h_3 < rand_{h2}\ \ (b) \end{cases} \quad (9)$$

$$c(t) = G(i) - V(i) \quad (10)$$

$G(i)$ represents the vector's position.

$rand_4$ is a random number in $[0 - 1]$.

To apply this method, the process starts with calculating the distance between the vulture and one of two vests using Eq. (8). Subsequently, a spiral equation is derived among the vultures, with their movement being directed by the parameter B, as outlined in Eq. (11):

$$K = V(i) \times \left(\frac{rand_5 \times G(i)}{2\pi}\right) \times \cos\big(h(i)\big) \quad (11)$$

$rand_5$ represents a randomly generated number.

When K drops below 0.5, the AVOA proceeds into its second exploitation phase. If the random value generated by $\mathrm{rand}_{\mathrm{h_3}}$ matches or exceeds the $\mathrm{h_3}$ the vultures engage in collection strategies, including training on various food sources. In case the random number produced by $\mathrm{rand}_{\mathrm{h_3}}$ is less than the $\mathrm{h_3}$ parameter, alternative strategies are activated as defined in Eq. (12) and Eq. (13):

$$U_1 = BestVulture_1(i) - \frac{BestVulture_1(i) \times V(i)}{BestVulture_1(i) \times V(i)^2} \times K \quad (12)$$

$$U_2 = BestVulture_2(i) - \frac{BestVulture_2(i) \times V(i)}{BestVulture_2(i) \times V(i)^2} \times K \quad (13)$$

$BestVulture_1(i)$ and $BestVulture_2(i)$ signify the top-performing vultures in the first and second groups, respectively.

In the ultimate phase of the African vulture optimization algorithm, all vultures congregate as per the steps in Eq. (14) (a). In this stage, vultures may experience conflicts and disputes as they encircle one another, as illustrated in Eq. (14) (b).

$$V(i+1) = \begin{cases} \frac{(U_1 + U_2)}{2} & if\ h_a \geq rand_{h_3}\ \ (a) \\ G(i) - |c(t)| \times K \times L(c) & if\ h_a < rand_{h_3}\ \ (b) \end{cases} \quad (14)$$

Levy Flight (L) is introduced to bolster the efficiency of the African vulture optimization algorithm, as detailed in Eq. (15). It is coupled with Eq. (14) (b) to replicate the conflicts and clashes that can take place among the vultures in the algorithm's concluding stage.

$$L(d) = 0.01 \times \frac{y \times \delta}{|w|^{1/a}} \quad (15)$$

$$\delta = \left(\frac{\tau(1+b) \times \sin\left(\frac{\pi b}{2}\right)}{\tau(1+2b) \times b \times 2 \times \frac{(b-1)}{2}}\right)^{1/b} \quad (16)$$

d represents the dimensionality of the issue, signifying the count of variables or dimensions in question.

b is a constant set at a fixed value of 1.5.

y represents random numbers in the range of 0 to 1.

The essential stages of the African vulture optimization algorithm are expounded through pseudo-code in Algorithm 1.

Algorithm 1: Pseudo-Code of AVOA Algorithm

```
Inputs: The population size N and maximum number of iterations T
Outputs: The vulture's position and its associated fitness value
Initialize the random population h_i (i = 1,2,...,N)
while (stopping condition is not met) do
Calculate the fitness values of the vulture
Set hBestVulture1 as the location of Vulture (First best location Best Vulture Category 1)
Set hBestVulture2 as the location of Vulture (Second best location Best Vulture Category 2)
for (each vulture (h_i)) do
Select G(i)
Update the K
if (|K| ≥ 1) then
if (h_1 ≥ randh_1) then
Update the location of the vulture
else
Update the location of Vulture
if (|K| < 1) then
if (|K| ≥ 0.5) then
if (h_2 ≥ randh_2) then
Update the location of the vulture
else
Update the location of the vulture
else
if (h_3 ≥ randh_3) then
Update the location of the vulture
else
Update the location of the vulture
Return hBestVulture1
```

3) Sand Cat Swarm Optimization (SCSO): The SCSO algorithm, whole in study [41], takes cues from the foraging actions of desert-dwelling sand cats. These exceptional felines have a distinctive skill for identifying low-frequency sounds, allowing them to pinpoint prey regardless of whether it is positioned above or below the surface. The algorithm's core concept focuses on identifying the best point within an exploration area, much like prey in the natural hunting environment of a sand cat. To achieve this objective, the algorithm utilizes a search agent that consistently explores the search area by periodically updating its position, gradually moving toward the estimated location of the best value. The SCSO algorithm is intricately organized, consisting of two core components: a prey-locating mechanism and a prey-capturing mechanism. The prey-locating method imitates how sand cats hunt for prey in their natural habitat, guided by a mathematical equation defining the population's search patterns. This formula reflects the combined behaviors of sand cats as they explore their surroundings in search of potential objectives, serving as the algorithm's fundamental approach to optimization and identifying the sought-after solution within the exploration area.

$$\vec{X}(t+1) = \vec{r}.\vec{X}_b(t) - rand(0,1).\vec{X}_c(t) \quad (17)$$

$\vec{X}$ depicts the search agent's positional vector.

t indicates the iteration number for the present cycle.

$\vec{X}_b(t)$ denotes the location of the top contender during iteration t

$\vec{X}_c(t)$ represent the recent place of the hunt agent at repetition t.

r signifies the scope of sand cats' receptiveness to low-pitched sounds, and this receptiveness may be elucidated as follows:

$$\vec{r} = \vec{r}_G \times rand(0,1) \quad (18)$$

$\vec{r}_G$ indicates the overall responsiveness span, which diminishes linearly from 2 to 0. Eq. (19) may be expounded upon as follows:

$$\vec{r}_G = s_M - \left(\frac{s_M \times iter_c}{iter_{max}}\right) \quad (19)$$

$iter_c$ symbolizes the present rendition, $iter_{max}$ embodies the ultimate count of iterations. Furthermore, considering that sand cats detect low frequencies of 2 kHz, the magnitude of s_M is adjusted to 2.

The SCSO procedure commences the predator assault stage upon the culmination of the prey exploration, and the sand cats' populace predator attack mechanism can be elucidated as below:

$$\vec{X}_{md} = \left|rand(0,1).\vec{X}_b(t) - \vec{X}_c(t)\right| \quad (20)$$

$$\vec{X}(t+1) = \vec{X}_b(t) - \vec{r}.\vec{X}_b(t).cos(\theta) \quad (21)$$

θ symbolizes an arbitrary angle spanning from 0 to 360 degrees. Consequently, the trigonometric function $cos(\theta)$ produces values within the interval of -1 to 1.

$\vec{X}_{md}$ alludes to the position that is created through a combination of the optimal position and the present position.

By employing this strategy, each individual within the populace can travel along unique circular paths. Every sand feline picks a haphazard azimuth, allowing them to navigate away from localized optimal snares as they draw near the quarry's position. The stochastic angle delineated in Eq. (21) is pivotal in shaping the agent's pursuit and exploration trajectory.

The SCSO algorithm balances the discovery and exploitation stages using a flexible parameter called vector r. This parameter can be expounded upon as follows:

$$R = 2 \times \vec{r}_G \times rand(0,1) - \vec{r}_G \quad (22)$$

$\vec{r}_G$ decreases gradually from 2 to 0 in a linear manner as iterations advance. The modified description of the sand cat's locations during both the investigative and exploitation stages can be articulated below:

$$\vec{X}(t+1) = \begin{cases} \vec{r}.(\vec{X}_b(t) - rand(0,1).\vec{X}_c(t)) & |R| > 1 \\ \vec{X}_b(t) - \vec{r}.\vec{X}_b(t).cos(\theta) & |R| \leq 1 \end{cases} \quad (23)$$

Within the SCSO algorithm, the exploration operative commences an assault on the target quarry once the absolute magnitude of R falls beneath or equals 1. In these cases, where $|R| > 1$,

the exploration operative transitions into a worldwide exploration mode, scrutinizing for conceivable resolutions across an extended spectrum. Notably, every unique sand cat individual holds a different exploration scope in the investigative stage, thus averting the algorithm from becoming entangled in localized optimal solutions.

Pseudo-code illustrates Algorithm 2 [42].

Algorithm 2: Pseudo-code of SCSO Algorithm

```
Commence the population setup.
Compute the fitness metric.
Commence the r; r⃗_G; R
while (t ≤ iter_max) do
for each agent do
Obtain an arbitrary angle (0 ∘ ≤ θ ≤ 360 ∘)
if (|R| ≤ 1) then
Update the search operative's location
else
Update the search operative's location
end if
end for
t = t + 1
end while
```

C. Research Methodology

This research aims to develop an accurate and reliable model for predicting building cooling load consumption using advanced machine learning techniques. The process involves several systematic steps, from data preparation and model development to performance evaluation and optimization. To

make it easier for readers to understand the methodology and replicate the results, the steps are outlined clearly and concisely as follows:

1) Data collection and preparation: Source the Dataset: Collect a dataset from previous research containing 768 samples with eight input variables related to building geometry and glazing characteristics.

Split the Dataset: Divide the dataset into training (70%), validation (15%), and testing (15%) sets.

2) Model development

Base KNN Model: Develop a basic K-nearest neighbors (KNN) model using the training dataset.

Optimize the KNN Model:

African Vultures Optimization (AVO): Apply AVO to the KNN model to create the KNAV model.

Sand Cat Swarm Optimization (SCSO): Apply SCSO to the KNN model to create the KNSC model.

3) Model training

Train the Base Model: Train the KNN model using the training dataset.

Train the Optimized Models: Train the KNAV and KNSC models using the optimized parameters derived from AVO and SCSO, respectively.

4) Model validation

Validate Performance: Use the validation dataset to evaluate the performance of the KNN, KNAV, and KNSC models.

Adjust Parameters: Fine-tune model parameters based on validation results to enhance performance.

5) Model testing and evaluation

Test the Models: Assess the performance of the KNN, KNAV, and KNSC models using the testing dataset.

Evaluate Using Metrics: Utilize various metrics such as R^2, RMSE, MSE, RAE, and PI to compare the performance of the three models.

Analyze Results: Identify the most effective model based on the evaluation metrics.

Fig. 1 illustrates the schematic presentation of the research methodology.

D. Performance Evaluation Metrics

The dataset was partitioned into three subsets: training, validation, and testing. The models' performance was rigorously evaluated through a thorough analysis of various metrics, encompassing R^2 (coefficient of determination), RMSE (Root Mean Square Error), MSE (Mean Square Error), RAE (Relative Absolute Error), and PI (Prediction Interval). The criteria for evaluating model performance are defined by the following parameters, as summarized below:

$$R^2 = \left(\frac{\sum_{i=1}^{n}(A_i - \bar{A})(B_i - \bar{B})}{\sqrt{[\sum_{i=1}^{n}(A_i - \bar{B})^2][\sum_{i=1}^{n}(B_i - \bar{B})^2]}} \right)^2 \quad (24)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(B_i - A_i)^2}{n}} \quad (25)$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(B_i - A_i)^2 \quad (26)$$

$$RAE = \sum_{j=1}^{n} \frac{|A_i - B_i|}{|A_i - \bar{A}|} \quad (27)$$

$$PI = \bar{x}_2 \pm t_{(\alpha/2,\ N-2)} * k^2 \quad (28)$$

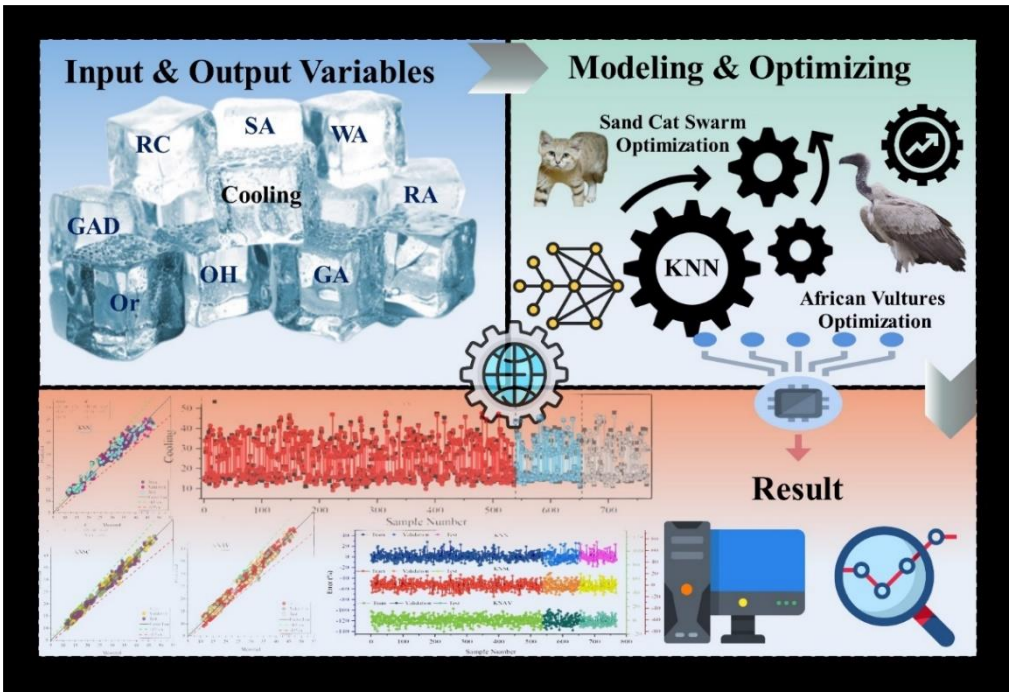


Fig. 1. Schematic presentation of the research methodology.

n: The total number of data points.

A_i: The test results for individual data points.

B_i: The predicted results for individual data points.

$\overline{A}$: The average of the test result values.

$\overline{B}$: The average of the prediction result values.

k^2: The standardized error value pooled across both groups.

$\overline{x}_1$ and $\overline{x}_2$: These are the sample means for the two groups under comparison.

$t_{(\alpha/2,\ N-2)}$: The t-value for the desired level of confidence (α) and the degrees of freedom (N-2).

III. Results and Discussion

Table II provides a comprehensive evaluation of model accuracy based on performance metrics, encompassing R^2, RMSE, MSE, RAE, and PI for all prediction models applied to cool load estimation across the training, validation, testing, and all datasets: The KNSC hybrid model demonstrates impressive performance, with maximum R^2 values of 0.985 for training and 0.983 for all data. Although R^2 values in the testing phase are slightly lower, approximately 1%, compared to training, the models still maintain strong alignment with the dataset, indicating robust predictive capabilities. The KNSC model also stands out with a minimal PI value of 0.023, reflecting reduced prediction uncertainty. In contrast, the KNN model has the highest PI value of 0.045 in the validation phase. Analysis of error values reveals the KNSC model's superior performance, with the lowest RMSE = 1.142, MSE = 1.304, values observed during training, and RAE = 28.03 value observed during the validation phase. This evidence underscores the KNSC hybrid model's high accuracy.

After a thorough comparison between the KNSC as the optimal model, the single KNN model, and the KNAV as another hybrid model, the observations from Fig. 2 and Fig. 3 lead to the following discernment: The KNSC model showcases superior performance by concentrating predicted cooling load values more closely around the central line. In contrast, the single KNN model displays a broader dispersion of data beyond the acceptable range of a $\pm 15\%$ underestimation and overestimation. Additionally, the KNSC model demonstrates a stronger alignment between observed and predicted cooling load values. The KNSC model, demonstrating slightly superior performance to KNAV, exhibits minimal disparity between the observed and forecasted data points. As previously discussed, this model also achieved a superior R^2 compared to other models, affirming its excellence in comparison to the other models.

In Fig. 4 and Fig. 5, two distinct visualization formats illustrate the error values of three models (KNSC, KNAV, and KNN). The histogram plot featured in Fig. 4 delineates the frequency distribution of error values across the developed models. The KNN model exhibits a notably higher concentration of errors near zero per cent, with approximately 135 values falling within this range. In contrast, the KNSC model registers 70 such values, while the KNAV model records 45. Based on the proximity of the error values to zero, it becomes evident that the KNSC model outperforms other models. In Fig. 5, each model has a unique Y graph, which is specified according to the color of the train part. Upon initial inspection, it is discernible that the single KNN model exhibits the highest error range, spanning from -20% to +30%, indicating the model's weak performance. Moreover, it is worth highlighting that the KNSC model exhibits the most minimal error values, as evidenced by metrics such as $RMSE_{Train}$ =1.142, MSE_{Train}=1.304 and $RAE_{Validation}$=28.03, as presented in Table II.

TABLE II. The Result of Developed Models for KNN

Model	Index values	Phase			
		Train	***Validation***	***Test***	***All***
KNN	RMSE	1.700	2.174	1.989	1.824
	R2	0.967	0.953	0.956	0.963
	MSE	2.891	4.727	3.957	3.326
	RAE	326.03	39.66	45.36	439.65
	PI	0.035	0.045	0.041	0.037
KNSC	RMSE	1.142	1.509	1.494	1.261
	R2	0.985	0.978	0.977	0.983
	MSE	1.304	2.278	2.231	1.589
	RAE	161.12	28.03	36.01	233.49
	PI	0.023	0.031	0.030	0.026
KNAV	RMSE	1.456	1.842	1.828	1.579
	R2	0.977	0.967	0.965	0.973
	MSE	2.120	3.393	3.343	2.494
	RAE	294.97	35.84	34.61	394.87
	PI	0.030	0.038	0.037	0.032

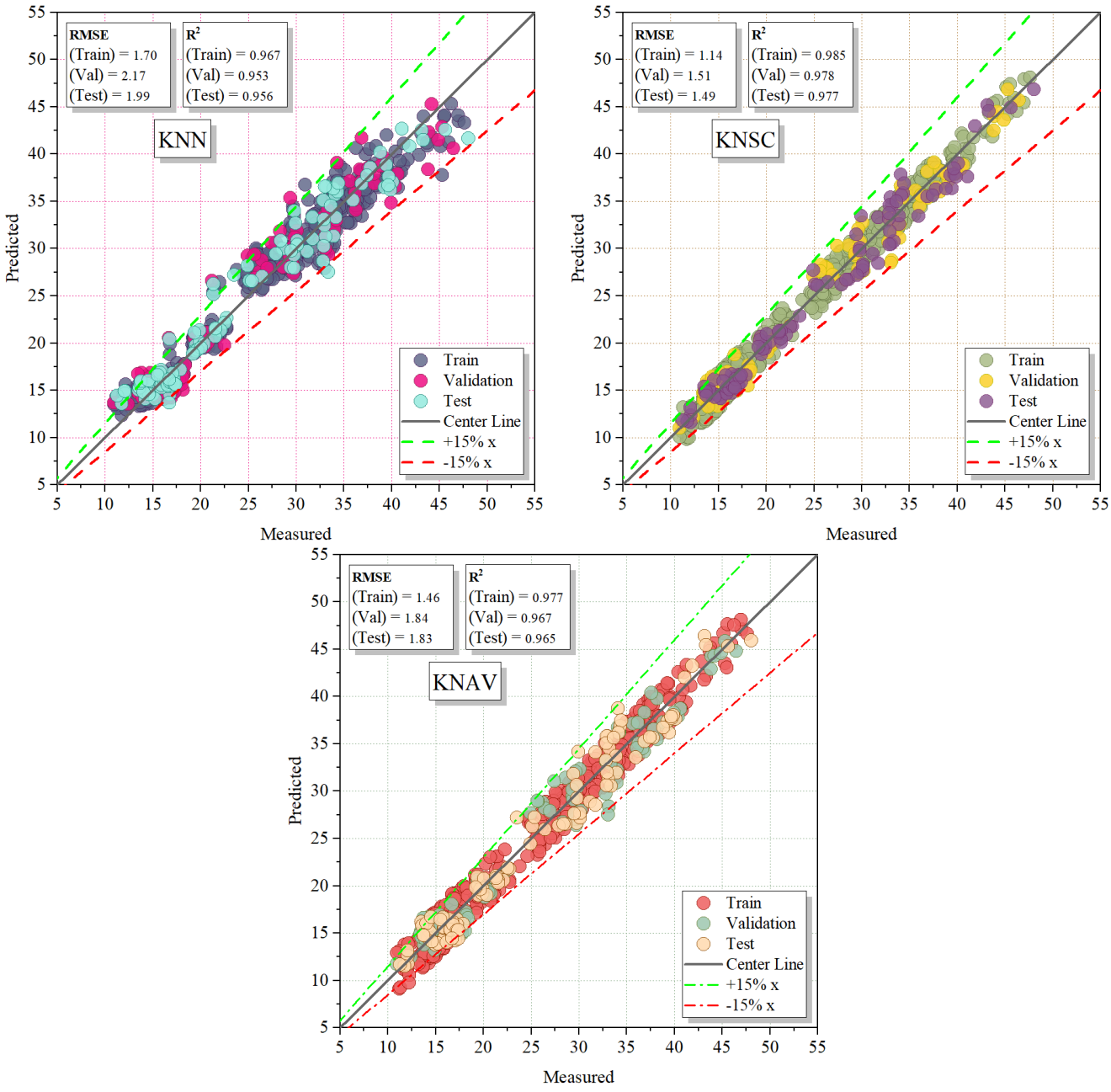
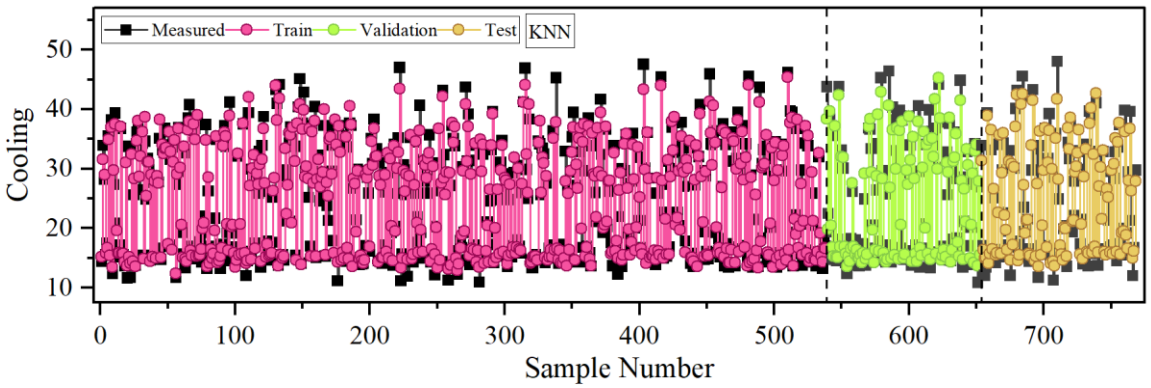


Fig. 2. Scatter plot for developed models.



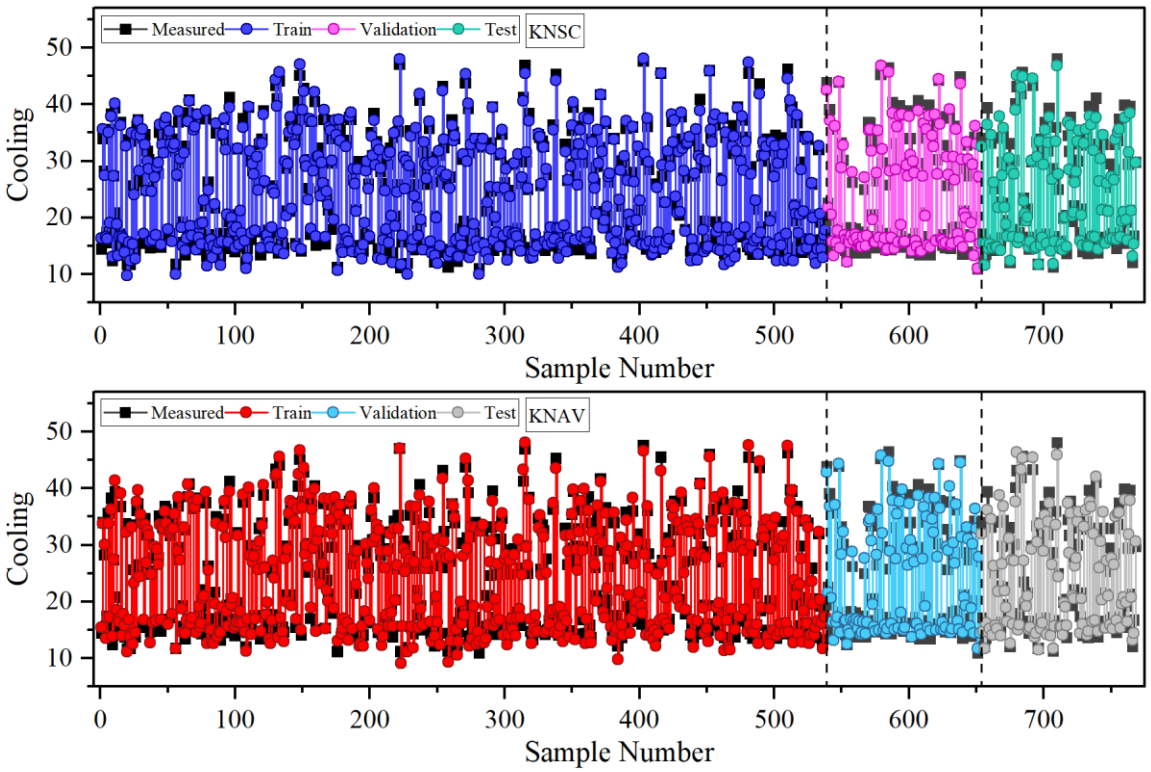


Fig. 3. Comparison of measured and predicted values.

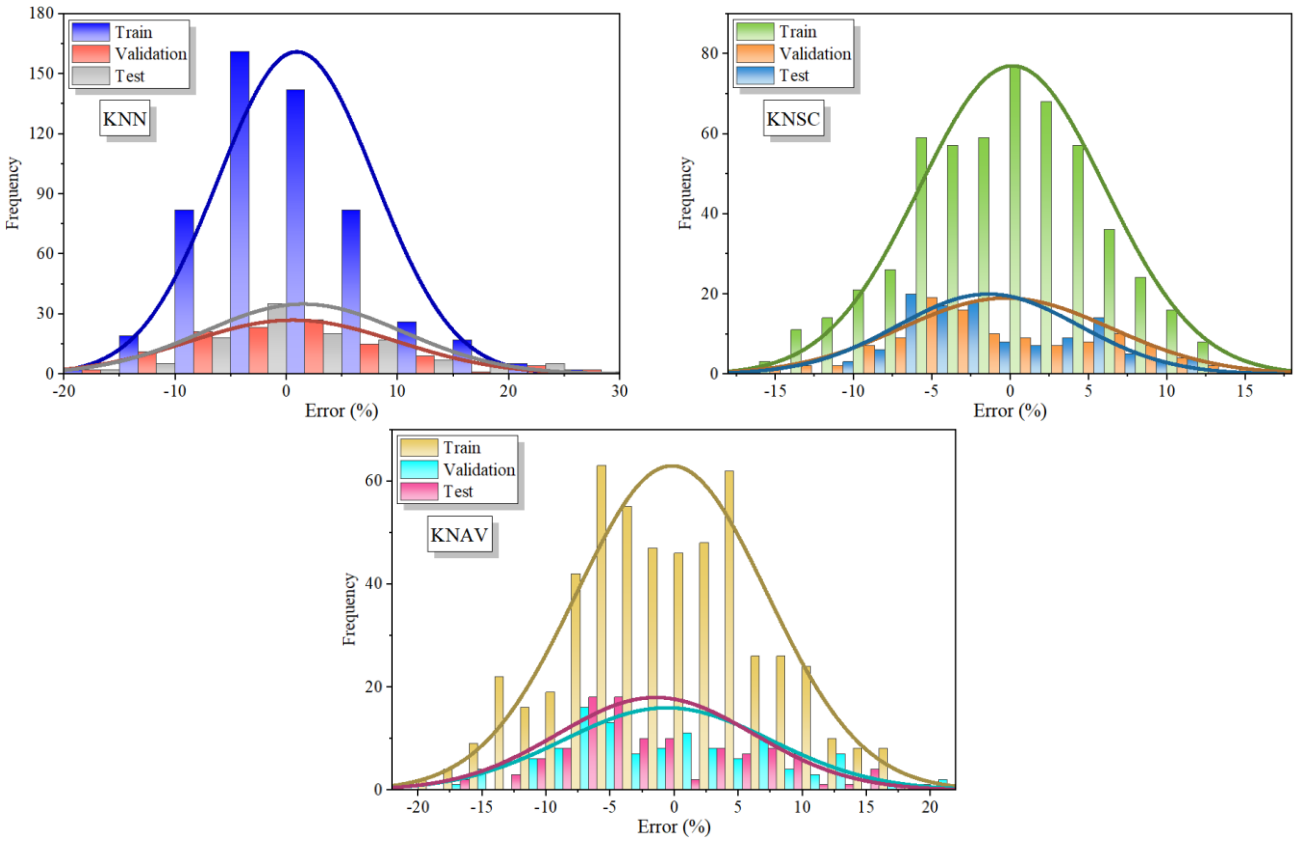


Fig. 4. Error percentage for the models based on the Histogram Distribution plot.

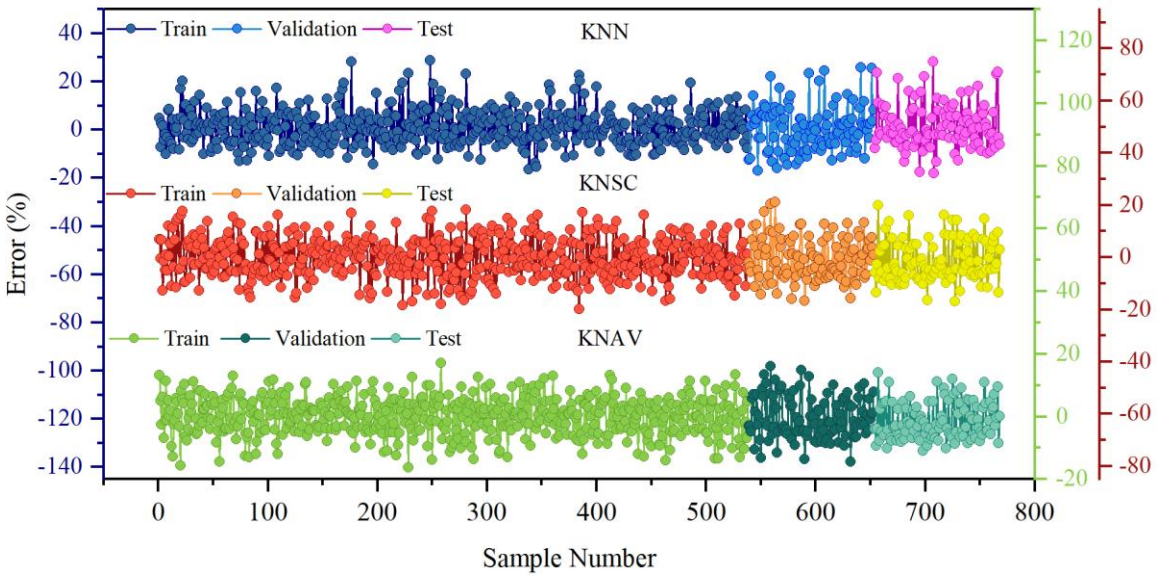


Fig. 5. Line symbol plot for errors in the developed models.

- Comparison between the results of present study and previous publications.

Several studies have investigated cooling load prediction. Afzal et al. [43] utilized the MLP model, while Gong et al. [44] applied the GPR technique. According to Table III, Gong et al. [44] demonstrated superior performance with the GBM model, achieving an R^2 value of 98.82% and an RMSE of 0.1929. In this study, the foundational framework of the KNN model was adopted and enhanced by hybridizing it with SCSO and AVO algorithms. Upon evaluation, the integration of SCSO into the KNN model proved exceptionally effective, achieving an R^2 value of 98.5% and an RMSE of 1.142, outperforming the other models in this study.

TABLE III. COMPARISON BETWEEN THE RESULTS OF THIS STUDY WITH PREVIOUS ARTICLES

Author(s)	Reference	Model	Results	
			R^2	*RMSE*
Gong et al.	[43]	GBM	98.82%	0.1929
Afzal et al.	[44]	MLP	98.06%	1.4122
Present study		**KNN+SCSO**	**98.5%**	**1.142**

IV. CONCLUSION

In summary, this research substantially contributed substantially to energy efficiency and sustainable building practices. It introduced innovative machine learning techniques, specifically incorporating K-nearest neighbors (KNN) models, which included a conventional model, an optimized version utilizing African Vultures Optimization (KNAV), and another optimized through Sand Cat Swarm Optimization (KNSC). These methodological strategies collectively addressed the crucial challenge of accurately predicting cooling load in building applications. Through a comprehensive analysis of input variables and a meticulous evaluation of model performance, the study emphasized the reliability and superiority of the KNSC hybrid model. The KNSC model achieved the highest coefficient of determination (R^2) at 0.985, surpassing the KNN and KNAV models by 1.86% and 0.81%, respectively. Additionally, it exhibited a significantly reduced root mean square error (RMSE) of 1.143, representing a 32.8% improvement compared to KNN and a 21.5% improvement compared to KNAV. These results underscored the KNSC hybrid model's capacity to revolutionize energy planning, enabling optimized energy production, distribution, and consumption within building systems. Consequently, this study propelled the field of predictive modeling for energy consumption at the time, offering a promising pathway toward more sustainable building practices and a greener future where the principles of energy efficiency and environmental preservation took precedence.

REFERENCES

[1] Pi ZX, Li XH, Ding YM, Zhao M, Liu ZX. Demand response scheduling algorithm of the economic energy consumption in buildings for considering comfortable working time and user target price. Energy Build 2021;250:111252.

[2] Oh M, Jang KM, Kim Y. Empirical analysis of building energy consumption and urban form in a large city: A case of Seoul, South Korea. Energy Build 2021;245:111046.

[3] Lei L, Chen W, Wu B, Chen C, Liu W. A building energy consumption prediction model based on rough set theory and deep learning algorithms. Energy Build 2021;240:110886.

[4] Hachem-Vermette C. Multistory building envelope: Creative design and enhanced performance. Solar Energy 2018;159:710–21.

[5] Athienitis AK, Barone G, Buonomano A, Palombo A. Assessing active and passive effects of façade building integrated photovoltaics/thermal systems: Dynamic modelling and simulation. Appl Energy 2018;209:355–82.

[6] Calero M, Alameda-Hernandez E, Fernández-Serrano M, Ronda A, Martín-Lara MÁ. Energy consumption reduction proposals for thermal systems in residential buildings. Energy Build 2018;175:121–30.

[7] Zhang R, Nie Y, Lam KP, Biegler LT. Dynamic optimization based integrated operation strategy design for passive cooling ventilation and active building air conditioning. Energy Build 2014;85:126–35.

[8] Huide F, Xuxin Z, Lei M, Tao Z, Qixing W, Hongyuan S. A comparative study on three types of solar utilization technologies for buildings:

Photovoltaic, solar thermal and hybrid photovoltaic/thermal systems. Energy Convers Manag 2017;140:1–13.

[9] Gautam KR, Andresen GB. Performance comparison of building-integrated combined photovoltaic thermal solar collectors (BiPVT) with other building-integrated solar technologies. Solar Energy 2017;155:93–102.

[10] Suh HS, Kim DD. Energy performance assessment towards nearly zero energy community buildings in South Korea. Sustain Cities Soc 2019;44:488–98.

[11] Kim D-B, Kim DD, Kim T. Energy performance assessment of HVAC commissioning using long-term monitoring data: A case study of the newly built office building in South Korea. Energy Build 2019;204:109465.

[12] Capozzoli A, Piscitelli MS, Brandi S, Grassi D, Chicco G. Automated load pattern learning and anomaly detection for enhancing energy management in smart buildings. Energy 2018;157:336–52.

[13] Liu X, Ding Y, Tang H, Xiao F. A data mining-based framework for the identification of daily electricity usage patterns and anomaly detection in building electricity consumption data. Energy Build 2021;231:110601.

[14] Yu Z, Fung BCM, Haghighat F. Extracting knowledge from building-related data—A data mining framework. Build Simul, vol. 6, Springer; 2013, p. 207–22.

[15] Ding Y, Brattebø H, Nord N. A systematic approach for data analysis and prediction methods for annual energy profiles: An example for school buildings in Norway. Energy Build 2021;247:111160.

[16] De Wilde P. The gap between predicted and measured energy performance of buildings: A framework for investigation. Autom Constr 2014;41:40–9.

[17] Park B, Srubar III W V, Krarti M. Energy performance analysis of variable thermal resistance envelopes in residential buildings. Energy Build 2015;103:317–25.

[18] Golbazi M, Aktas CB. Energy efficiency of residential buildings in the US: Improvement potential beyond IECC. Build Environ 2018;142:278–87.

[19] McLarty D, Brouwer J, Ainscough C. Economic analysis of fuel cell installations at commercial buildings including regional pricing and complementary technologies. Energy Build 2016;113:112–22.

[20] Goodfellow I, Bengio Y, Courville A. Deep learning. MIT press; 2016.

[21] Bishop CM, Nasrabadi NM. Pattern recognition and machine learning. vol. 4. Springer; 2006.

[22] Moradzadeh A, Mohammadi-Ivatloo B, Abapour M, Anvari-Moghaddam A, Roy SS. Heating and cooling loads forecasting for residential buildings based on hybrid machine learning applications: A comprehensive review and comparative analysis. IEEE Access 2021;10:2196–215.

[23] Zhao R, Wei D, Ran Y, Zhou G, Jia Y, Zhu S, et al. Building cooling load prediction based on lightgbm. IFAC-PapersOnLine 2022;55:114–9.

[24] Chen Q, Xia M, Lu T, Jiang X, Liu W, Sun Q. Short-term load forecasting based on deep learning for end-user transformer subject to volatile electric heating loads. IEEE Access 2019;7:162697–707.

[25] Roy SS, Samui P, Nagtode I, Jain H, Shivaramakrishnan V, Mohammadi-Ivatloo B. Forecasting heating and cooling loads of buildings: A comparative performance analysis. J Ambient Intell Humaniz Comput 2020;11:1253–64.

[26] Abdelkader E, Al-Sakkaf A, Ahmed R. A comprehensive comparative analysis of machine learning models for predicting heating and cooling loads. Decision Science Letters 2020;9:409–20.

[27] Mokeev V V. Prediction of heating load and cooling load of buildings using neural network. 2019 International Ural Conference on Electrical Power Engineering (UralCon), IEEE; 2019, p. 417–21.

[28] Moradzadeh A, Mansour-Saatloo A, Mohammadi-Ivatloo B, Anvari-Moghaddam A. Performance evaluation of two machine learning techniques in heating and cooling loads forecasting of residential buildings. Applied Sciences 2020;10:3829.

[29] Song J, Zhang L, Xue G, Ma Y, Gao S, Jiang Q. Predicting hourly heating load in a district heating system based on a hybrid CNN-LSTM model. Energy Build 2021;243:110998.

[30] Li X, Yao R. A machine-learning-based approach to predict residential annual space heating and cooling loads considering occupant behaviour. Energy 2020;212:118676. https://doi.org/https://doi.org/10.1016/j.energy.2020.118676.

[31] Lin X, Tian Z, Lu Y, Zhang H, Niu J. Short-term forecast model of cooling load using load component disaggregation. Appl Therm Eng 2019;157:113630.

[32] Wang H-J, Jin T, Wang H, Su D. Application of IEHO–BP neural network in forecasting building cooling and heating load. Energy Reports 2022;8:455–65.

[33] Leiprecht S, Behrens F, Faber T, Finkenrath M. A comprehensive thermal load forecasting analysis based on machine learning algorithms. Energy Reports 2021;7:319–26.

[34] Jihad AS, Tahiri M. Forecasting the heating and cooling load of residential buildings by using a learning algorithm "gradient descent", Morocco. Case Studies in Thermal Engineering 2018;12:85–93.

[35] Zhou G, Moayedi H, Bahiraei M, Lyu Z. Employing artificial bee colony and particle swarm techniques for optimizing a neural network in prediction of heating and cooling loads of residential buildings. J Clean Prod 2020;254:120082.

[36] Pessenlehner W, Mahdavi A. Building morphology, transparence, and energy performance. na; 2003.

[37] Wu X, Kumar V, Ross Quinlan J, Ghosh J, Yang Q, Motoda H, et al. Top 10 algorithms in data mining. Knowl Inf Syst 2008;14:1–37.

[38] Akbulut Y, Sengur A, Guo Y, Smarandache F. NS-k-NN: Neutrosophic set-based k-nearest neighbors classifier. Symmetry (Basel) 2017;9:179.

[39] Qian Y, Zhou W, Yan J, Li W, Han L. Comparing machine learning classifiers for object-based land cover classification using very high resolution imagery. Remote Sens (Basel) 2014;7:153–68.

[40] Abdollahzadeh B, Gharehchopogh FS, Mirjalili S. African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems. Comput Ind Eng 2021;158:107408.

[41] Seyyedabbasi A, Kiani F. Sand Cat swarm optimization: A nature-inspired algorithm to solve global optimization problems. Eng Comput 2023;39:2627–51.

[42] Li Y, Wang G. Sand cat swarm optimization based on stochastic variation with elite collaboration. IEEE Access 2022;10:89989–90003.

[43] Gong M, Bai Y, Qin J, Wang J, Yang P, Wang S. Gradient boosting machine for predicting return temperature of district heating system: A case study for residential buildings in Tianjin. Journal of Building Engineering 2020;27:100950.

[44] Afzal S, Ziapour BM, Shokri A, Shakibi H, Sobhani B. Building energy consumption prediction using multilayer perceptron neural network-assisted models; comparison of different optimization algorithms. Energy 2023:128446. https://doi.org/10.1016/j.energy.2023.128446.