INNOVATIVE ALGORITHMS FOR ANALYZING HEAT INDEX DATA: A SUSTAINABLE APPROACH FOR INDUSTRIAL IOT APPLICATIONS

Thota Maria Dwijitha[1], Elluru Datta Sai Charan [2], Dyagala Raghu [3], Mr. K. Harikrishna[4]
[1,2,3] UG Scholar, Dept. of ECE, St. Martin's Engineering College, Secunderabad, Telangana, India, 500100.
[4]Assistant Professor, Dept. of ECE, St. Martin's Engineering College, Secunderabad, Telangana, India, 500100.
tmariadiju@smec.ac.in

Abstract:

The increasing demand for energy efficiency and safety in industrial environments has led to the integration of Internet of Things (IoT) technologies to monitor and analyze environmental conditions, such as temperature and humidity. One such critical metric is the Heat Index (HI), which combines air temperature and humidity to represent the perceived temperature by humans. In industrial settings, high heat levels can significantly affect worker safety, equipment performance, and energy consumption. This paper proposes innovative algorithms for analyzing heat index data as a sustainable approach for Industrial IoT applications. By leveraging advanced data fusion, machine learning models, edge computing, and time series forecasting, these algorithms aim to optimize environmental management, predictive maintenance, and operational efficiency in industries such as manufacturing, agriculture, and warehousing. The paper discusses the history and evolution of heat index monitoring in industrial sectors, identifies limitations in traditional methods, and highlights the need for more efficient, scalable, and real-time solutions. Moreover, it emphasizes the significance of these algorithms in achieving sustainability goals, reducing operational costs, and improving safety standards. The concept of the Heat Index (HI) dates back to the 1970s when it was first introduced to quantify the effect of humidity on temperature, affecting human comfort levels. As industrialization expanded, the need to monitor environmental conditions to ensure worker safety and equipment reliability became critical. Traditional systems for monitoring temperature and humidity typically relied on basic instrumentation with manual data collection, leading to inefficiencies in identifying extreme heat conditions and responding proactively. Over time, technological advancements in sensors, IoT, and data analytics have enabled more sophisticated and real-time monitoring of heat index data, paving the way for its application in industrial IoT systems.

Keywords: Industrial IoT (IIoT), Heat Index (HI), temperature and humidity, worker safety, energy efficiency, environmental monitoring, predictive maintenance, operational efficiency, data fusion, machine learning models, edge computing, time series forecasting, real-time monitoring, sustainability goals, manufacturing, agriculture, warehousing, sensor technology, data analytics.

1. INTRODUCTION

The rapid growth of the Internet of Things (IoT) has revolutionized various industries by enabling real-time data collection and analysis through interconnected devices. In the context of environmental monitoring, IoT sensors play a crucial role in measuring parameters like temperature, humidity, and pressure, facilitating smart decision-making in domains such as industrial automation, urban planning, and energy management. This project focuses on using IoT temperature data to classify locations as indoor ("in") or outdoor ("out") using machine learning techniques. Traditional systems for such classification often rely on static thresholds or manual processes, which are inadequate for handling the complexity and volume of IoT data. To address these challenges, the project leverages advanced algorithms like Decision Tree Classifier and XGBoost Classifier to automate the classification process and provide a scalable, accurate solution.

The classification of environmental data into indoor and outdoor categories is a critical requirement for various IoT-based applications such as environmental monitoring, smart energy systems, and industrial automation. Traditional systems rely on static thresholds or manual processes to distinguish between indoor and outdoor environments, which often lead to inaccuracies due to the complexity of real-world data. Moreover, the increasing volume of IoT data makes manual methods impractical and inefficient. The key challenge lies in handling multi-dimensional data generated by IoT sensors, where relationships between variables such as temperature, time, and location may not be linear or intuitive. The lack of adaptive and scalable systems for data classification results in suboptimal utilization of IoT infrastructure, leading to wasted energy, compromised environmental monitoring, and limited automation capabilities. This project addresses these challenges by implementing machine learning algorithms, specifically Decision Tree Classifier and XGBoost Classifier, to accurately classify IoT temperature data as indoor or outdoor. The problem emphasizes the need for an automated, efficient, and scalable solution to handle large datasets while delivering accurate and reliable results.

As IoT adoption continues to grow, the ability to process and derive insights from sensor data becomes increasingly essential. IoT-based temperature monitoring has applications ranging from smart homes and industrial automation to environmental conservation. Traditional methods of classification are outdated, incapable of handling high-dimensional data, and fail to adapt to dynamic environments. The motivation for this research stems from the need to harness the full potential of IoT data through machine learning models that can automate and improve classification tasks. By replacing manual and static methods with intelligent algorithms, the project aims to pave the way for smarter and more sustainable IoT systems. Additionally, the research seeks to demonstrate the power of feature engineering and advanced algorithms like Decision Tree Classifier and XGBoost Classifier in solving real-world classification problems. This motivation aligns with the broader goals of improving efficiency, reducing resource wastage, and enabling smarter decision-making in IoT-driven ecosystems.

This project is significant for its potential to transform IoT-based environmental monitoring and classification systems. By automating the classification of indoor and outdoor environments, it enables more efficient energy management, precise environmental monitoring, and better resource allocation. The use of machine learning algorithms ensures adaptability to changing conditions, improving the reliability of predictions over time. The project's emphasis on scalability makes it highly relevant for industries managing large-scale IoT deployments. For example, in smart cities, accurate classification of environmental data can optimize energy consumption in public spaces. Similarly, in industrial settings, it can enhance predictive maintenance by understanding environmental changes that might impact equipment. Another key significance is the potential to reduce manual intervention, saving time and minimizing errors. The insights derived from this project can also drive innovation in IoT-based applications, encouraging the adoption of data-driven decision-making in diverse industries. Moreover, it contributes to sustainable development goals by optimizing resource utilization and reducing energy waste. It can also be integrated into security systems for enhanced identification at entry points in sensitive areas such as airports, military installations, and government buildings. Additionally, it can be used in surveillance systems to accurately identify individuals under various lighting conditions, improving the reliability of monitoring and tracking efforts.

The classification of indoor and outdoor environments using IoT data has a wide range of applications. In smart energy management, it helps optimize heating, ventilation, and air conditioning (HVAC) systems in buildings by identifying indoor and outdoor conditions, reducing energy consumption, and improving sustainability in smart homes and offices. In environmental monitoring, it enhances air quality monitoring systems by categorizing data based on location and supports urban planning by providing detailed environmental insights. In industrial automation, it improves predictive maintenance by accounting for environmental factors that affect equipment performance and enhances safety systems by monitoring temperature variations in specific zones. In smart cities, it enables efficient resource allocation in public spaces by understanding environmental conditions and improves the management of shared infrastructure such as parks, streets, and transportation systems. Finally, in health and safety, it helps monitor indoor and outdoor environments to ensure safe conditions for workers in industrial setups and assists in designing better ventilation systems in enclosed spaces.

2. LITERATURE SURVEY

Maintaining a stable and comfortable indoor climate in buildings is essential for the health and well-being of occupants while also playing a critical role in optimizing energy consumption and reducing environmental impact [1]. This is particularly significant in the context of rising energy prices and the pressing need to combat global warming. As a result, the construction industry is shifting toward "smart" buildings, which leverage automation to achieve maximum energy efficiency and occupant comfort [2]. Additionally, new regulations and standards worldwide demand improved energy efficiency and indoor environmental quality in modern buildings [3]. In this scenario, surrogate modeling has emerged as a transformative tool for enhancing building energy management and climate control. Surrogate models (SMs), or metamodels, are simplified representations of complex systems that approximate the behavior of detailed models at a fraction of the computational cost. These models enable faster forecasting and real-time decision-making, crucial for smart building applications. By integrating flexibility and adaptability, SMs ensure relevance and accuracy in dynamic environments while reducing the computational burden, making them essential components of modern building management systems [4]. Several types of surrogate models (SMs) exist, each with distinct characteristics and applications. These include polynomial regression, Gaussian processes, radial basis functions (RBFs), support vector machines (SVMs), among others [5]. Polynomial regression uses polynomial functions to model dependencies between variables, being simple and straightforward, but less accurate for very complex functions[6]; Gaussian Process provides flexible modeling of complex dependencies using probabilistic approaches to estimate forecast uncertainty[7]; RBFs use distance-to-center functions to build smooth interpolation models, which is useful for problems with irregular data[8]; SVMs use hyperplanes to separate data in high-dimensional spaces where nonlinear kernels can model complex relationships [9]. Studies like those by Villano, Mauro, and Pedace [10] have extensively reviewed ML and deep learning (DL) methods for building energy management, emphasizing their potential and limitations. While CNN-LSTM models, as proposed by Elmaz et al. [11], have demonstrated high accuracy ($R^2 > 0.9$) in IAT prediction, their architectural complexity necessitates substantial computational resources, making them less practical for resource-constrained settings. Two strategies were developed in Mtibaa et al. [12] LSTM-MISO for multi-input single-output and LSTM-MIMO for multi-output prediction. The performance of these strategies was evaluated based on two real smart buildings with variable (VAV) and constant (CAV) air volume systems. The experimental results showed a significant advantage of the LSTM model over multilayer perceptron models. Hamayat et al. [13], in turn, propose the use of a certain modification of Bi-LSTM, which is a black box model based on artificial intelligence and data-driven approaches. The results of the numerical experiments showed an improvement in IAT prediction of up to 10% using the proposed model compared to the standard LSTM model for IAT prediction. The study by Liang et al. [14] employs a surrogate modeling approach similar to ours, to replace complex physical models with simplified yet accurate data-driven models. However, their work has limitations, including the lack of consideration for time dependencies due to the use of the K-nearest neighbors (KNN) algorithm, the specificity of their model to a single region (Shanghai, China), and the restriction of input parameters to only seven key building characteristics. Zouloumis et al. [15] present a model for predicting the required thermal capacity of buildings using multilinear regression and analyzing heat loss, heat capacity, and air infiltration. While this traditional approach demonstrates satisfactory accuracy, it has notable limitations. Langtry et al. [16] examined the use of data to improve the accuracy of building condition forecasting models and the effectiveness of a model predictive control (MPC) scheme in a distributed generation and storage power system. Their findings revealed that a simple linear multilayer perceptron model could achieve forecast accuracy comparable to more advanced machine learning models, with the added benefits of requiring less data and computation. Kontopoulou et al. [17], conducted a comparative analysis of ARIMA models alongside machine learning methods such as neural networks, support vector machines, decision trees, linear models, and deep learning. The effectiveness of using ARIMA and SARIMA models in forecasting is described by Petropoulos et al. [18]. Additionally, the use of the Holt-Winters method19 accounts for trends and seasonality in time series, making it effective for predicting building performance when clear cyclic patterns are present. Each of the discussed methods has its advantages and limitations, and they can be used individually or in combination to obtain more accurate and reliable building performance forecasts.

3. PROPOSED METHODOLOGY

This project focuses on analyzing IoT-based temperature data to classify whether a location is indoor or outdoor using machine learning algorithms. By preprocessing the data, engineering relevant features, and training models, the goal is to make accurate and reliable predictions that can support industries such as environmental monitoring, smart city development, and building management systems. The process begins with data cleaning and preparation, ensuring missing values and duplicates are handled while converting datetime information into structured features. Feature engineering involves extracting meaningful components such as day, month, year, hour, and minute from the timestamp to enhance the dataset. Correlations between features are explored to identify key drivers influencing the classification of locations as indoor or outdoor.

For model development, two machine learning algorithms—XGBoostClassifier and DecisionTreeClassifier—are trained and evaluated. The trained models are saved for future use and efficient deployment. Performance evaluation is conducted using metrics like Mean Squared Error (MSE), Mean Absolute Error (MAE), and R^2 score, with visualizations such as scatter plots comparing actual and predicted values. The trained models are then tested on new data, where predictions are saved along with input values for validation. To ensure scalability and reusability, the models are saved using joblib, allowing them to be reused without retraining. The workflow is modularized for potential integration with real-time systems.

The structured workflow begins with data loading, where the dataset (IOT-temp.csv) is imported using pandas for manipulation and exploration. Exploratory Data Analysis (EDA) follows, involving basic statistics, checking for missing values, and identifying duplicate entries. Data cleaning steps include removing duplicate rows, converting timestamps into datetime objects, and extracting date-time components for feature engineering. Data visualization techniques such as heatmaps are applied to explore feature correlations. Preprocessing includes encoding categorical variables like indoor/outdoor classification using LabelEncoder. The dataset is then split into training (80%) and testing (20%) subsets.

During model training, both XGBoostClassifier and DecisionTreeClassifier are implemented. If pre-trained models are available, they are loaded; otherwise, new models are trained and saved for future use. Performance evaluation is conducted through a custom metrics function that computes MSE, MAE, and R^2 scores, with scatter plots visualizing actual vs. predicted values. The trained models are then tested on a randomly selected sample of 20 rows from the dataset, with predictions saved into testdata.csv. The final output consists of predictions alongside insights from the models, with trained models stored for future deployment.

DecisionTreeClassifier is a key algorithm used in this project. It is a machine learning model that builds a decision tree, where each internal node represents a feature, each branch represents a decision rule, and each leaf node represents a class label. The dataset is recursively split based on the most significant features, allowing the tree-like structure to make predictions by following paths from root to leaf. In the project, DecisionTreeClassifier classifies whether a location is indoor or outdoor based on IoT temperature data. The workflow for this classifier involves preparing the dataset by extracting features such as date, month, year, hour, and minute, followed by splitting the dataset into training (80%) and testing (20%) sets.

The model is trained using the training dataset, learning decision rules that maximize the separation between target classes. Performance evaluation is conducted using MSE, MAE, and R^2 score, followed by testing on a sample dataset for prediction. The trained model is saved as a .pkl file (decision_tree_classifier.pkl) to facilitate reuse without retraining. This classifier is suitable for the project due to its simplicity, ability to handle numerical and categorical data, and automatic feature selection. It also serves as a baseline model for comparison with advanced algorithms like XGBoostClassifier.

Applications:

This project has several real-world applications across different industries:

- **Environmental Monitoring**:

  Enhances air quality analysis and pollution tracking.

  Supports climate studies and weather station data classification.

- **Smart Cities**:

  Optimizes heating, cooling, and ventilation systems.

  Contributes to energy-efficient urban infrastructure.

- **Building Management Systems**:

  Enables automated temperature control for cost-effective operations.

  Improves HVAC (Heating, Ventilation, and Air Conditioning) efficiency.

- **Real-time IoT Deployments**:

  Helps classify temperature data in weather stations and sensor-based monitoring.

  Enhances safety applications with environmental condition tracking.

- **Industrial Applications**:

  Supports supply chain logistics for temperature-sensitive goods.

  Ensures optimal storage and transportation conditions.

- **Healthcare & Wellness**:

  Monitors indoor and outdoor temperature exposure for health analysis.

  Aids in personalized climate adaptation strategies.

Advantages:

- **Easy to Understand & Interpret** – The visual representation makes it intuitive and accessible, even for non-technical users.
- **Handles Both Numerical & Categorical Data** – Works efficiently with different data types, requiring minimal preprocessing.
- **No Assumption on Data Distribution** – Performs well with real-world datasets without needing a predefined distribution.
- **Feature Importance Analysis** – Identifies the most relevant features, offering insights into key classification factors.
- **Scalability & Efficiency** – Can process large datasets effectively and be deployed for real-time applications.
- **Minimal Data Preprocessing Required** – Does not require scaling or normalization, saving time in model development.
- **Captures Complex Relationships** – Effectively models non-linear patterns and learns from historical trends.
- **Robust to Outliers** – Less sensitive to extreme values due to logical data splitting.

- **Works Well for IoT Temperature Classification** – Efficiently processes timestamp and numerical attributes, serving as a strong baseline model before exploring advanced algorithms like XGBoost

4. EXPERIMENTAL ANALYSIS

The project focuses on analyzing IoT temperature data using machine learning models to classify whether a location is indoor or outdoor. It involves data preprocessing, feature engineering, machine learning model training, evaluation, and prediction. The implementation begins with importing essential libraries such as pandas and numpy for data manipulation, matplotlib and seaborn for visualization, and sklearn for preprocessing and model evaluation. Additionally, joblib is used for

Fig: Displaying the regression report of AdaBoost model ed to *IOT-temp.csv*, is loaded into a pandas DataFrame, and various exploratory steps are performed, including checking its shape, displaying summary statistics, and identifying missing or duplicate values.

Data cleaning involves removing duplicate rows and converting the *noted_date* column into a datetime format. New features such as date, month, year, hours, and minutes are extracted from this column for enhanced analysis, and the original *noted_date* column is dropped. A heatmap is generated to visualize correlations between numerical features, helping identify relationships within the dataset. Categorical variables like *location* are then encoded using Label Encoding to convert them into numerical values. The dataset is split into features (X) and the target variable (y), where *out/in* serves as the classification target. The data is then divided into training and testing sets, with 80% allocated for training and 20% for testing.

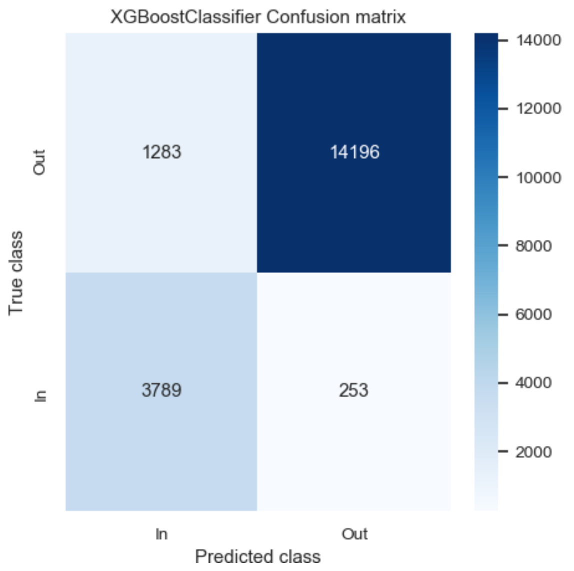

Fig 4.1: Illustration of confusion matrix obtained using XGBoost model.

```
DecisionTreeClassifier Accuracy     : 97.24911633625327
DecisionTreeClassifier Precision    : 94.88541079034094
DecisionTreeClassifier Recall       : 97.04064107029942
DecisionTreeClassifier FSCORE       : 95.91067969877511

 DecisionTreeClassifier classification report
               precision    recall  f1-score   support

          In       0.97      0.91      0.94      4311
         Out       0.97      0.99      0.98     15210

    accuracy                           0.97     19521
   macro avg       0.97      0.95      0.96     19521
weighted avg       0.97      0.97      0.97     19521
```

A custom function is implemented to calculate and display key performance metrics, including Mean Squared Error (MSE), Mean Absolute Error (MAE), and R^2 Score. A scatter plot visualizes actual versus predicted values with a trend line to assess model performance. The machine learning models used for classification include XGBoost and Decision Tree classifiers. If a pre-trained XGBoost model is available, it is loaded for prediction; otherwise, a new model is trained, evaluated, and saved. The same process is followed for the Decision Tree classifier. Random sampling is performed on 20 rows of the dataset, which are saved as *testdata.csv*. The Decision Tree classifier then predicts the class labels for this sample data, and the results are stored within the test dataset.

The dataset consists of IoT-based temperature readings collected over time, aiming to classify environments as indoor or outdoor. Key features include *noted_date*, which is split into separate components for improved analysis, *temperature* to differentiate between indoor and outdoor settings, *humidity* to measure moisture content in the air, *location* to specify where the sensor was placed, and *pressure* to capture atmospheric conditions. The *out/in* column serves as the target variable for classification. Several preprocessing steps are applied, including handling missing values, removing duplicates, performing feature engineering, and encoding categorical variables. The dataset provides valuable insights, enabling the development of machine learning models that automate indoor/outdoor classification using temperature, humidity, and time-based attributes.

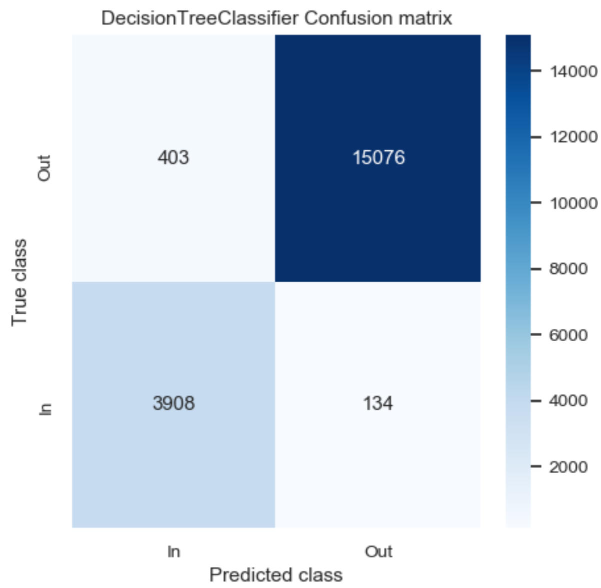

Fig 4.2: Illustration of confusion matrix obtained using Decision Tree model.

Model name	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
XGBoost	92.13	86.47	92.72	89.00
Decision tree	97.24	94.88	97.04	95.91

Fig: Comparison of all models

The results indicate strong classification performance. The XGBoost classifier achieves an accuracy of 92.13%, demonstrating high precision and recall, with an F1-score of 89.01%. Meanwhile, the Decision Tree classifier outperforms XGBoost, achieving an impressive accuracy of 97.2%, with high precision and recall values, leading to a well-optimized classification model.

5. CONCLUSION

The project demonstrates an effective approach to classifying indoor and outdoor environments using IoT temperature data and machine learning algorithms like Decision Tree Classifier and XGBoost Classifier. By automating the classification process, it overcomes limitations of traditional systems, such as manual intervention, static thresholds, and low scalability. The use of advanced algorithms ensures higher accuracy, adaptability, and efficiency, making it suitable for large-scale IoT deployments.

The integration of feature engineering, model evaluation, and performance metrics highlights the importance of data preprocessing and algorithm selection in achieving reliable results. Additionally, the ability to save trained models for future use allows for seamless integration into real-time IoT applications.

The future scope of the project is vast, with several promising extensions and applications. One major advancement would be integrating the model with real-time IoT systems, enabling continuous data streaming from sensors for dynamic classification. Incorporating additional environmental features such as humidity, air pressure, and light intensity could further enhance the model's accuracy and adaptability.

Another key improvement involves developing hybrid models that combine machine learning with deep learning techniques. This would allow the system to handle complex data more effectively while improving classification accuracy. To ensure scalability and efficiency, deploying the project on cloud platforms or edge devices would enable real-time processing in industrial IoT applications.

Predictive analytics could also be integrated to anticipate environmental changes or anomalies, supporting proactive decision-making in smart systems. Additionally, incorporating Explainable AI (XAI) would enhance transparency by providing insights into the model's decision-making process. This would be particularly valuable for critical applications like healthcare and industrial safety.

REFERENCES

[1] Kim, D. et al. Design and implementation of Smart buildings: a review of Current Research Trend. Energies 15 (12), 4278. https://doi.org/10.3390/en15124278 (2022).

[2] Aliero, M. S., Asif, M., Ghani, I., Pasha, M. F. & Jeong, S. R. Systematic review analysis on Smart Building: challenges and opportunities. Sustainability 14 (5), 3009. https://doi.org/10.3390/su14053009 (2022).

[3] Liao, H., Ren, R. & Li, L. Existing building renovation: a review of barriers to Economic and Environmental benefits. Int. J. Environ. Res. Public Health. 20 (5), 4058. https://doi.org/10.3390/ijerph20054058 (2023).

[4] Pan, X., Xu, Y. & Hong, T. Surrogate Modelling for Urban Building Energy Simulation based on the bidirectional long short-term memory model. J. Build. Perform. Simul. 1–19. https://doi.org/10.1080/19401493.2024.2359985 (2024).

[5] Franzoi, R. E., Kelly, J. D., Menezes, B. C., Christopher, L. E. & Swartz An adaptive sampling Surrogate Model Building Framework for the optimization of reaction systems. Comput. Chem. Eng. 152 (September), 107371. https://doi.org/10.1016/j.compchemeng.2021.107371 (2021).

[6] Ogren, A. C., Feng, B. T., Bouman, K. L. & Ch. Daraio Gaussian process regression as a surrogate model for the computation of Dispersion relations. Comput. Methods Appl. Mech. Eng. 420 (February), 116661. https://doi.org/10.1016/j.cma.2023.116661 (2024).

[7] Arora, G., KiranBala, H. E. & Kh. Masoumeh A review of radial basis function with applications explored. J. Egypt. Math. Soc. 31 (1), 6. https://doi.org/10.1186/s42787-023-00164-3 (2023).

[8] Gu, J. et al. Surrogate model-based Multiobjective optimization of high-speed PM Synchronous Machine: construction and comparison. IEEE Trans. Transp. Electrification. 9 (1), 678–688. https://doi.org/10.1109/TTE.2022.3173940 (2023).

[9] Choi, Y., Doosam, S., Sungmin, Y. & Junemo, K. Comparison of Factorial and Latin Hypercube sampling designs for Meta-models of Building Heating and cooling loads. Energies 14 (2), 512. https://doi.org/10.3390/en14020512 (2021).

[10] Villano, F., Mauro, G. M. & Pedace, A. A review on Machine/Deep learning techniques Applied to Building Energy Simulation, optimization and management. Thermo 4 (1), 100–139. https://doi.org/10.3390/thermo4010008 (2024).

[11] Elmaz, F. et al. CNN-LSTM Architecture for predictive indoor temperature modeling. Build. Environ. 206 (December), 108327. https://doi.org/10.1016/j.buildenv.2021.108327 (2021).

[12] Mtibaa, F. et al. LSTM-Based indoor air temperature prediction Framework for HVAC systems in Smart buildings. Neural Comput. Appl. 32 (23), 17569–17585. https://doi.org/10.1007/s00521-020-04926-3 (2020).

[13] Hamayat, F., Ullah, M. O., Zubair, S. & Anwar, S. M. An Indoor Air Temperature Prediction Framework for Model Predictive Control in HVAC Systems. In 2022 27th International Conference on Automation and Computing (ICAC), 1–6. Bristol, United Kingdom: IEEE. (2022). https://doi.org/10.1109/ICAC55051.2022.9911173

[14] Liang, Y., Pan, Y., Yuan, X., Jia, W. & Zh. Huang Surrogate modeling for long-term and high-resolution prediction of Building Thermal load with a Metric-optimized KNN Algorithm. Energy Built Environ. 4 (6), 709–724. https://doi.org/10.1016/j.enbenv.2022.06.008 (2023).

[15] Zouloumis, L., Stergianakos, G., Ploskas, N. & Panaras, G. Dynamic Simulation-based surrogate model for the Dimensioning of Building Energy systems. Energies 14 (21), 7141. https://doi.org/10.3390/en14217141 (2021).

[16] Langtry, M. et al. Impact of data usage for forecasting on performance of Model Predictive Control in buildings with Smart Energy Storage. Energy Build. 320, 114605. https://doi.org/10.1016/j.enbuild.2024.114605 (2024).

[17] Kontopoulou, V. I., Panagopoulos, A. D., Kakkos, I. & Matsopoulos, G. K. A review of ARIMA vs. Machine Learning approaches for Time Series forecasting in Data Driven Networks. Future Internet. 15 (8), 255. https://doi.org/10.3390/fi15080255 (2023).

[18] Petropoulos, F. et al. Forecasting: theory and practice. Int. J. Forecast. 38 (3), 705–871. https://doi.org/10.1016/j.ijforecast.2021.11.001 (2022).