Journal of Building Engineering 75 (2023) 107016

Contents lists available at ScienceDirect

Journal of Building Engineering

journal homepage: www.elsevier.com/locate/jobe

Charging water load prediction for a thermal-energy-storage air-conditioner of a commercial building with a multilayer perceptron

Mirza Rayana Sanzana [a,*], Mostafa Osama Mostafa Abdulrazic [b], Jing Ying Wong [a], Tomas Maul [b], Chun-Chieh Yip [c]

[a] Department of Civil Engineering, Faculty of Science and Engineering, University of Nottingham Malaysia, 43500, Semenyih, Hulu Langat District, Selangor, Malaysia

[b] Department of Computer Science, Faculty of Science and Engineering, University of Nottingham Malaysia, 43500, Semenyih, Hulu Langat District, Selangor, Malaysia

[c] Department of Civil Engineering, University Tunku Abdul Rahman, Bandar Sungai Long, 43000, Cheras, Kajang, Selangor, Malaysia

A R T I C L E I N F O

Keywords:
Deep learning (DL)
Facility management (FM)
Facility management and maintenance (FMM)
Heating
Ventilation
And air conditioning (HVAC)
Deep neural networks
Multilayer perceptron

A B S T R A C T

This research focuses on the development of a machine learning model for predicting the water volume that needs to be chilled in Thermal-Energy-Storage-Air-Conditioning (TES-AC) systems. TES-AC technology uses thermal energy storage tanks to store and distribute chilled water during peak hours, reducing the need for the continuous operation of chillers and resulting in significant cost savings and a reduction in carbon emissions. However, determining the optimal amount of chilled water to generate and store each day can be challenging. The aim of this research is to design a machine learning model that takes input variables about the next day's weather, which day of the week it is, and occupancy data and outputs a predicted water volume that needs to be chilled. It utilizes a Multilayer Perceptron for charging water load prediction in TES-AC systems to assist facility managers in making informed decisions minimizing disruptions. By fine-tuning the hyperparameters of the deep learning model and evaluating different metrics, the model was trained sufficiently and optimized. The model provides a specific water range as a target output, giving facility managers a small set of ranges to choose from, minimizing errors, while the accuracy achieved was 93.4%. The developed model can be retrained for other TES-AC plants, without requiring specific sensor input that might not be available in different TES-AC systems. That makes the developed solution more flexible and can encourage more stakeholders to use TES-ACs which in turn would lead to greener buildings that would benefit the environment.

1. Introduction

The climate crisis has become a concern with global warming being evident across the world. With heat waves, more humidity, and extreme temperatures being felt, the usage of Air-Conditioner (AC) has become a necessity rather than an amenity [1]. Generally, a conventional AC functions by giving a desirable temperature for the indoor space by conditioning the air and releasing the unwanted heat and humidity outdoors. In this process, a conventional AC also emits lots of greenhouse gases and as such contributing negatively

* Corresponding author.
E-mail addresses: sanzanarayana@gmail.com, khcy5mrs@nottingham.edu.my (M.R. Sanzana).

https://doi.org/10.1016/j.jobe.2023.107016
Received 24 April 2023; Received in revised form 17 May 2023; Accepted 4 June 2023
Available online 5 June 2023

to the environment. By constantly releasing the unwanted heat to the outdoors, an urban heat island effect is created which causes an increase of more undesirable weather conditions [2]. For much-needed environment-friendly and sustainable technology, Thermal-Energy-Storage Air-Conditioner (TES-AC) systems are being focused on (A [3–5]. A TES-AC system functions by simply transferring the charging load from on-peak to off-peak hours, reducing building energy consumption besides reducing greenhouse gas emissions [6,7].

Moreso, a TES-AC system stores thermal energy in the form of chilled water at night-time to cool the building the next day without dispersing unwanted heat and humidity outdoors as it constantly circulates the stored energy within its system [8,9]. The responsible entities for managing and maintaining this facility are facility managers who ensure the daily operations in a building go smoothly and minimize disruptions. However, the Facility Management and Maintenance (FMM) task for a TES-AC system is essentially complicated as an incorrect charging load can increase the building energy consumption when facility managers are required to charge during on-peak hours. Hence, predicting charging load to allow facility managers to prepare in advance during off-peak hours will be a useful contribution to the FMM and the environment with an increased shift to sustainable TES-AC systems. With a proper predictive maintenance with deep learning, facility managers will be more inclined to the shift towards sustainable technology as it will not cause occupancy discomfort or building management costs to increase [10].

Facility Management is the costliest phase of a building lifecycle in the Building Information Modeling (BIM) methodology, as after the construction of a building is completed, it requires monitoring, operations, and maintenance [11,12]. By integrating the appropriate management type for FMM, building management costs can be reduced, and satisfaction for both facility managers, and occupants can be increased. According to Industry 4.0, digital technologies are being implemented to improve the building lifecycle management [13,14]. As Deep Learning, a subset of Machine Learning within the Artificial Intelligence domain, is known for its prediction capability in identifying complex patterns by utilizing the deep layers of its models, it is suitable for charging water load prediction of a TES-AC [15,16].

TES-AC falls under the Heating, Ventilation, and Air-Conditioning (HVAC) of building facilities and deep learning can also be useful in FMM of HVAC components including an increased occupancy satisfaction. However, not much research has focused on utilizing deep learning techniques for predictive maintenance of HVAC though it can be a powerful tool [17]. Recently, a research study demonstrated that external weather factors and the number of occupants at a building at a certain time has an impact in the water consumption of TES-AC systems [18]. It is important to consider the external factors impacting the water consumption as this can affect the charging water load prediction. Hence, this research focuses on using deep learning techniques to predict the charging water load required for TES-AC systems solely depending on the external factors of weather, which day of the week it is, if that day is a holiday, and occupancy. Previously research studies have focused on predicting energy consumption of Thermal-Energy-Storage Air--Conditioners (TES-ACs) with deep learning, but importance was not given to the crucial prediction of charging water load [19–22]. As in a study by Ref. [17] comparing various Machine Learning Algorithms, Multilayer Perceptron (MLP) has shown satisfactory results for prediction of features similar to charging water load prediction, this research focuses on utilizing MLP.

The aim of this research is to predict the charging water load for TES-AC systems with MLP based on external factors such as weather data, occupancy number, and the days of the week. Besides, by simply considering the external factors and without the requirement of additional sensors, more facility managers may be inclined to benefit from the optimized TES-AC system with deep learning. For this research, data was collected from a commercial building for the charging water load, and occupancy concerning the type of day, and weather related to the commercial building. The facility managers of a commercial building have a complicated task of charging water load prediction and maintaining occupancy comfort besides not raising the management costs which is one of the key motivations for this research. The solution is intended to be useful for facility managers in making decisions regarding FMM of TES-AC, as to ensure low building energy consumption and management costs. Through an extensive study, a deep learning technique is suggested with satisfactory prediction of charging water load only based on external factors.

By removing the reliance on data that comes from specific sensors in specific TES-AC plants it is possible to apply this research in other TES-AC plants and train new models that can predict the water charging load needed for the next day. Furthermore, by demonstrating how variables like weather provide acceptable prediction results, this work can be adopted and used on TES-AC plants in different weather conditions.

The research questions are as follows.

RQ1: How to fine tune the hyperparameters for the MLP predicting charging the water load of the TES-AC system?

RQ2: What are the best methods to evaluate the performance of the developed MLP models?

The following objectives were carried out to address the above research questions:

RO1: To implement agrid search algorithm to determine the best hyperparameters for the MLP in predicting charging water load of the TES-AC system.

RO2: To implement and compare different evaluation metrics for the prediction.

2. Literature review

2.1. Overview of multilayer perceptron (MLP)

Deep Learning is a field that encompasses neural networks with many layers that allow the deep neural network to learn from large amounts of data with improved accuracy [23,24]. The additional layers of the neural network make the neural network deep as well as optimize and refine the prediction accuracy in big data i.e., large amounts of data [25,26]. The Multilayer Perceptron (MLP) is a deep learning method which was first introduced in 1958 by Frank [27] where the basic structure consists of an input layer, hidden layer(s), and an output layer. Being a type of feedforward neural network, the neurons in an MLP are typically trained with the backpropagation algorithm. The input layer processes the input signal, an arbitrary number of hidden layers are placed in between the input and output layer, and then the output layer carries out the required task such as the prediction [28]. MLP is one of the most common neural networks and is used in various disciplines for both classification and regression problems due to its architecture [29,30].

The reason why MLP falls under a feedforward architecture is because the inputs are combined with initial weights in a weighted sum and then subjected to the activation function and each neuron output is then propagated forward to the next layer. This way, every layer is feeding the next layer with the result of their computation which is their internal representation of the data, and this process is carried out through the hidden layers to the output layer. Backpropagation is implemented to iteratively adjust the weights in the network to minimize the loss or error. The activation function that is applied to the weighted sum needs to be differentiable, which is a hard requirement of the backpropagation algorithm. Such functions e.g., Rectified Linear Unit (ReLU) need to have a bounded derivative as Gradient Descent is the optimization function for MLPs normally. During each iteration, after the weighted sums have been forwarded through all layers with their activation functions, the gradient of the Mean Squared Error (MSE), for example, is calculated across all input-output pairs. Then during backpropagation, the weights of all hidden layers get updated according to the gradients computed for a particular mini-batch. This process continues until one or more convergence criteria are met (e.g. maximum number of epochs reached or validation error stagnation).

Examples of activation functions for the hidden layers include the Identity, Logistic, Tanh and ReLU functions [31,32]. The activation function, ReLU has been widely adopted because of its enhanced optimization, with Stochastic Gradient Descent, improved efficiency in computation, and its scale-invariance [33].

The four types of activation functions are as follows.

1. Identity: no-op activation, useful for linear bottleneck, has the formula, $f(x) = x$.
2. Logistic: the logistic sigmoid function, has the formula, $f(x) = 1/(1 + \exp(-x))$.
3. Tanh: the hyperbolic tan function, has the formula, $f(x) = \tanh(x)$.
4. ReLU: the rectified linear unit function, has the formula, $f(x) = \max(0, x)$.

Mathematically, even one hidden layered MLP model is able to predict the mapping of any continuous function [34]. For all neural networks, the input vector's dimension determines the number of neurons in the input layer, whereas the number of neurons in the outer layer depends on the classes to be learned. Through experimentation (or neural architecture search, or hyperparameter tuning, or meta-learning, or other approaches), the choice of the number of hidden layers is determined along with the number of neurons in each hidden layer. Determining the number of neurons is crucial because it can compromise the model's efficacy as too many neurons can result in overfitting, and too few neurons can lead to underfitting. While selecting the training dataset, it needs to be noted that a training set typically requires a rich and balanced representation of all classes for a proper generalization to occur. During training, input vectors are in a random order to ensure the model achieves global learning and does not lead to biased outputs due to class-selective learning. Multilayer Perceptrons (MLPs) can be trained to implement any given nonlinear input-output mapping and in the resulting testing phase, MLPs prove their ability in interpolation by generalization even in data sparse regions [34].

The Perceptron is recognized as an algorithm, and essentially gets its name from imitating the human-like function of perception. Rosenblatt's perceptron machine (1958) depended on a fundamental unit of computation, the neuron, and inputs in the input layer are connected in a weighted sum. When the weighted sum surpasses a pre-determined threshold, the neuron generates an output. Learning in MLPs requires adjusting the weights of constituent perceptrons to yield low error on the training data which is normally carried out with the backpropagation algorithm as it tries to minimize the loss function (e.g. Mean Squared Error (MSE)) though other algorithms can be used for this purpose as well [35]. Various techniques can be implemented to handle overfitting of the training data which often happens when models are too complex and exhibit low predictive performance due to modeling random error or noise in the training data. Generally, to avoid this overfitting issue, a higher error on the training data can be allowed and a large variety of regularization methods can be adopted [33].

2.2. Research studies utilizing Multilayer Perceptrons in facility management

Some of the common practical use cases of MLPs are pattern classification, recognition, prediction, and similarity estimation [17]. Deep learning has been used in various instances for construction project management, and also in the Facility Management of the Building Lifecycle Management phase in improving occupancy satisfaction, occupancy comport, building efficiency, and maintenance costs.

MLPs were also utilized for predicting construction project duration with a time-cost predictive model and obtained better accuracy compared to linear regression with data that involved information of contracted and real time of construction and contracted and real price of construction [36].

Reliable prediction methods are necessary to be implemented for optimal energy management to deal with uncertainties in generation and demand, especially for electric energy systems [37]. present a load forecasting model for electrical consumption of a medical clinic with an MLP where the inputs consisted of loads, data concerning the type of day (e.g. weekday/holiday), time of the day and weather data as an innovative technique that can be easily implemented into Building Automation System (BAS).

As HVAC contributes to high energy consumption within corporate buildings, constant monitoring is required so energy efficiency or occupancy satisfaction is not compromised. A simple optimized HVAC system with Internet-of-Things (IoT) is suggested for automating HVAC with demand response [38]. The thermal parameters from the sensors and occupancy feedback are collected for real-time processing in the distributed cloud environment and MLP was used for the predictive model of time-series forecasting and

Mixed Integer Linear Programming (MILP) problem was used for optimizing the HVAC control [38]. The results from the research by Ref. [38] showed 20%–40% reduction in energy consumption while maintaining thermal comfort of occupants, consistently proving MLP to be an optimal choice for prediction.

Occupancy satisfaction is a crucial goal for facility managers and one way to reduce the dissatisfaction of occupants is with an efficient facility management capable of addressing maintenance issues and planning for them in advance. Thermal discomfort is a major part of occupancy dissatisfaction in indoor environments and has been one of the most common complaints that require attention and addressing in buildings [39]. To address this [39] proposed a predictive maintenance strategy with MLP to analyse and then predict the number of thermal complaints for the upcoming week to allow facility managers in allocating resources to handle such complaints thereby improving occupancy satisfaction and building performance.

Energy consumption is not only related to building management costs but also to different types of environmental problems and emphasises on implementing deep learning for building energy consumption prediction to improve decision-making in decreasing energy usage [40]. Not much research has been dedicated to exploring building energy consumption prediction at the construction phase, though predicting energy usage by utilizing deep learning with input key features of a building design before construction can decrease the construction of non-environment-friendly buildings [40].

For classification problems, MLP is being widely used and the research conducted by Zhao et al. [41] achieved an accuracy of 82% in classifying rooms into nine different types for Two-Dimensional (2D) residential building plans. Generally, the prediction accuracy in automatic room classification is low when a 2D plan has insufficient details with missing attributes such as furniture information [41]. This shows that MLPs can still give satisfactory results in contexts with insufficient information. The literature review section demonstrates that MLPs are a viable choice for this research study, whilst covering broader ground which includes various other MLP applications in facility management. With appropriate design of the hidden layers, an efficient MLP model can be developed, and hence, this research utilizes MLPs for charging load prediction with external factors only.

3. Materials and methods

3.1. Research design overview

This research includes many typical steps that are required to solve complex machine learning problems. The objective is to predict the water charge load required to be chilled at night during off-peak hours so it can be discharged to the building during the day. The application is quite crucial, and predictions need to provide an adequate amount of accuracy, or the model will not be useful. The TES-AC used in this study uses a water-cooled chiller which is a part of an HVAC system for a commercial building. It includes twenty heat storage tanks and dissipates heat from the coolant using a plate heat exchanger. The heat gets out of the building using cooling water placed on the rooftop. The rest of the TES-AC plant is placed in the basement for bigger space and reduce heat loss. During building operation, chilled water travels from the heat storage tanks through pipes around the buildings and at each air vent there are fans that push air and release cool air into the environment. Warm water then travels back to the storage tanks where it is stored for cooling during the charging period.

Currently the facility managers in the TES-AC plant use their expertise to determine the water volume to be charged for the next day. They have sensors in the water tanks that read the water volume every 15 min and using a fixed schedule they check on the remaining water volume and therefore decide if they need extra water to be chilled using the chiller during operating hours.

Every TES-AC system might be different to some extent and each one might have different sensor types and data, so in order to generalize the solution more, none of the sensor data from the TES-AC were used. The model developed only relied on external data that can be retrieved by anyone. The data used to make the predictions consisted of the weather data based on the location of the building the TES-AC is used in, the data informing which day of the week it is and whether that day is a weekend or a public holiday, and finally the occupancy data.

Occupancy data were available for the historical data used to train the model, however, the occupancy in the future is not known. Therefore, this approach does not predict the water charge directly, as it needs to predict the occupancy before it can feed that prediction as an input to predict the water charge. While using a predicted value for the input of the main prediction might be making the problem more complicated, after thorough experimentation with the model and different parameters, it was noticed that the model was making better predictions if the occupancy data was included and therefore a second model to predict occupancy for the next day was also developed.

3.1.1. Data extraction

However, before discussing the details of the developed models and their designs and their results, the data has to be discussed. The TES-AC data provided for this research was from the company that owns the TES-AC, and the data was given in its raw format straight from the system in the form of a Microsoft Access database file. The data was then converted to comma separated files (CSV) for ease of usage and to allow manipulation with Python.

The data was analyzed carefully to understand the different types of data available and to understand which data was being collected. The main target was on the water data to understand how much water the TES-AC was chilling every day as that is the main focus. The TES-AC consisted of twenty water tanks, each of the same size and material and completely identical in every aspect. However, not all the tanks are used at the same time and the technical staff managing the TES-AC change the usage according to their needs. The data included sensor data indicating the water volume in each tank for a given time. The data was recorded every 15 min and stored accordingly.

Before working on extracting the data however, it needed to be preprocessed and cleared of issues. It is very unlikely for a dataset collected from sensors to not have problematic instances in it. Using the wide array of data manipulation libraries in Python like NumPy and Pandas the bad data were removed from the dataset. This included removing the data that did not have values or had wrong data types instead of actual numbers. Despite this step removing some of the data from the dataset, it was necessary and crucial.

The next step was to calculate how much water the TES-AC was using every day and to do so it was important to understand how the staff was using the chiller on a daily basis. As per information acquired directly from the company, the chiller runs for 2 h every day during the off-peak hours early in the morning before the occupants arrive and when the electricity is cheaper than usual. During those 2 h the chiller will cool a certain volume of water and store it in some of the tanks. During the rest of the day the chilled water will be dispersed throughout the building and the heated water coming back from the building enters the tanks once more. If there is not enough chilled water in the tanks to last the day, the staff are forced to use the chiller again during standard working hours when the electricity demand is higher, raising their costs.

Given this context, it can be deduced that the water volume will increase in the water tanks if new chilled water is entering into it or returned heater water. On the other hand, when water volume does decrease it is because water is being sent to the chiller to either cool or disperse to the building. It is to be noted that this research ignored the fact that a small amount of water might be decreasing due to natural evaporation and assumed all water lost is due to either of the above cases. This TES-AC is well insulated and natural evaporation is very minimal and the water mostly gets recycled.

To calculate the water that is being chilled every day and used, code was written to calculate the amount of water that decreases the water volume reading for the sensor. This was assumed as water being sent to the building as chilled water; therefore, it was water that needed to be chilled. In the situations where the staff did not add enough chilled water in the tanks and the chiller gets used, the new chilled water does not return to the tanks but is sent directly from the chiller to the building.

Therefore, in those situations the chiller would decrease the water in the tanks to chill the returned heated water and send it again to the building, making it safe to assume that whenever the water volume in the tanks decreases, it is because this water is being chilled for usage and that makes it count towards the water volume used every day. This calculation only ran on the 22 h where the main charging load is not happening, to ignore the initial charged load volume as it will interfere with the values. For example, if the initial water volume is also calculated, it will be calculated twice as it is discharged during the day. Another issue arises if that water volume discharged was not fully used. By only calculating the amount of water that leaves the tanks during the other 22 h it can be calculated in a more accurate way and a more precise value can be used. This was the best way to know the amount of water used as the chiller did not have any specific sensors to measure how much water it chills. This calculation ran for each day and each tank, and the final data extracted included the date and the total amount of water used for that day.

3.1.2. Weather data extraction

The weather data was retrieved for the specific location of the TES-AC is using open-source weather data available online. Python was again used for this task and the BeautifulSoup library was used to scrape a weather website for the historical weather data with the same date range of the data from the TES-AC. After the data was extracted both the water level data and the weather data were merged into one dataset. Out of the different weather metrics extracted, temperature, humidity, and atmospheric pressure were used.

The method to retrieve the weather for the next day is slightly different, however. Using an open-source weather website an API key is generated which is then used to make a GET request to the website to retrieve the weather information for the next day. The request returns a JSON output which is then parsed, and the needed information is extracted accordingly. This occurs every time a new prediction for the next day is required.

3.1.3. Days & holidays data extraction

Extracting the days of the week and public holidays was easier than the weather data. Python contains a datetime library and a calendar library that makes it easy and possible to determine what a specific date is. That data was added to the dataset and then the holidays library was used to retrieve the information if a specific date was a public holiday or not.

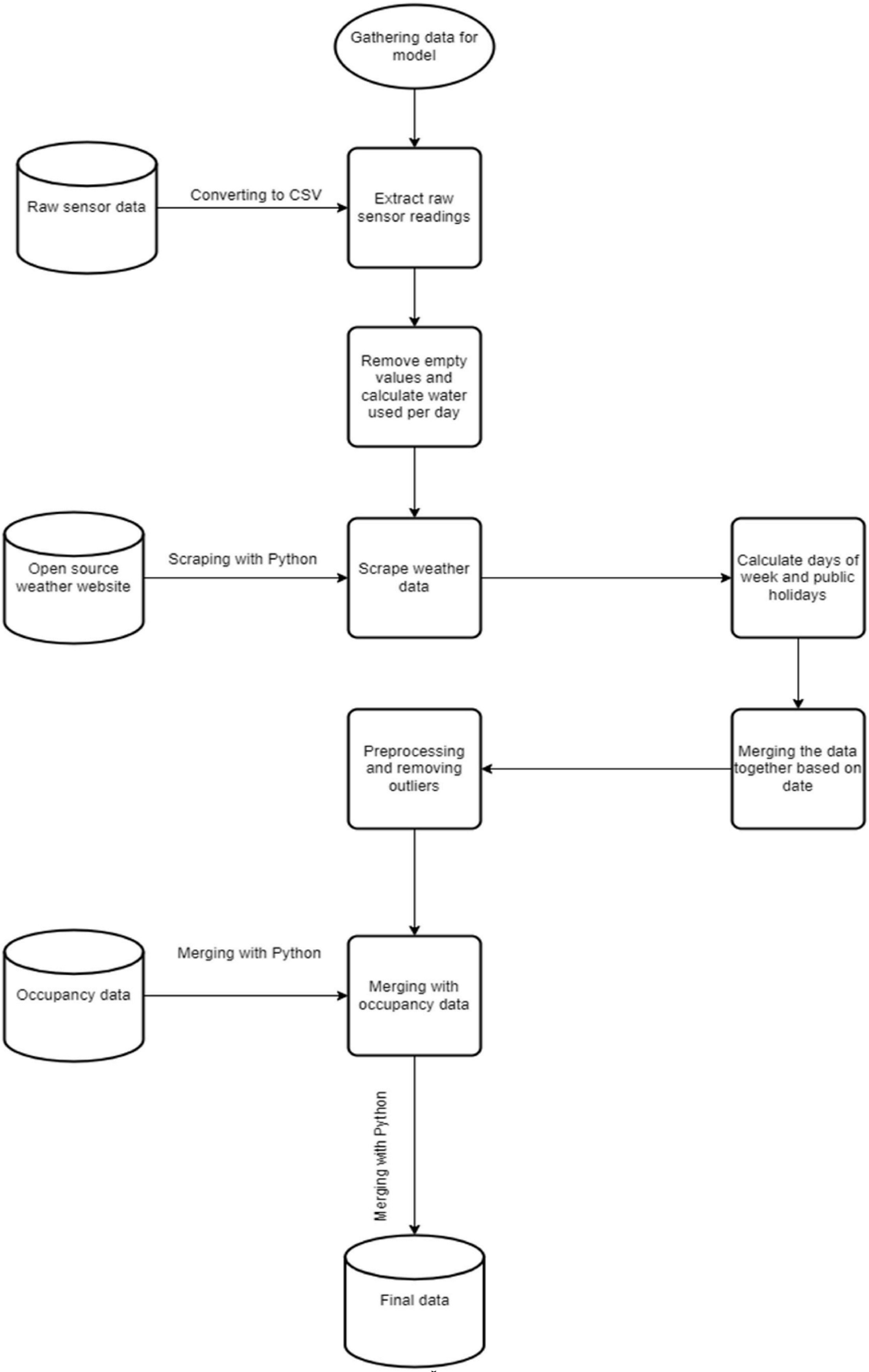
3.1.4. Merging data with occupancy

All of the data including water used per day, weather temperature, humidity, atmospheric pressure, name of the day, and whether that day was a public holiday or not were all merged together based on the date as that was the common factor. In doing so, any records that did not have the same date were removed from the dataset. The occupancy data was provided separately and simply included the date and the approximate number of people that were in the building for that day. The final dataset was then saved separately and ready for further preprocessing.

3.1.5. Preprocessing

The first step was to remove any empty values in the dataset which was done earlier. However, the step was repeated again after the rest of the data was added. The next step was to remove outlier values as they can negatively impact the training process. By checking some of the maximum and minimum values of the data it was apparent that there were a few outlier values that did not make much sense in relationship with the rest of the data. These values might be due to sensor reading errors or machine faults. They are significantly fewer in number but will impact on the overall training process. A general rule of thumb was used to remove those outlier values by keeping only the values that were less than or equal to the mean value added to the standard deviation multiplied by three.

To further improve the dataset a one-hot encoding representation was used on the days of the week data, replacing strings corresponding to week names by a binary vector of zeros with a single element set to 1, which indicates the day of the week based on its position. This step can help in the training process and make it faster to train a more complex model. Furthermore, label encoding was used on the public holiday data to represent a day with a one if it is a public holiday and a zero if it is not. The weather data that was



(caption on next page)

Fig. 1. Flow Chart outlining the Process of Data Gathering.

extracted had unit strings attached to it and therefore those strings were split to keep only the numbers and store them as appropriate floating numbers.

The following preprocessing step only took place after experimentation with the water volume data and trying to solve it as a regression problem by predicting a specific water volume to be charged. However, this method was not very effective and did not make much sense as predicting an exact water volume is not very realistic. Instead, providing a water volume range is more realistic and provides the staff with a close range so that using their expertise they can determine the exact figure. This process would also make prediction easier as the model needs to predict a range rather than an exact number. To do this however, the problem needed to be converted to a classification problem rather than a regression problem and that required labels.

The water volume was labelled according to their values and split into seven unique labels, each indicating the water volume range. The labels started from Low_1 having the least water volume range and increased to Low_2 and Low_3. After that, Mid_1 was used and so on. After three variations of Low, Mid, and High there was one label for any water volume above those ranges and that was labelled ExtraHigh. By checking the water volume number in each row, the proper label was added, and the data was finalized. The final data was scaled using the min-max scaler to keep it uniform and easier to train.

3.1.6. Research process diagram

Fig. 1 shows a simple flowchart indicating the process through which the final data was gathered. Starting from raw sensor values that get converted to CSV files, to calculating the water used every day from those sensor readings, Python is then used to scrape the required weather data and Python libraries are used to calculate the day of the week based on date and whether a date is a public holiday or not. This is all merged and then preprocessing takes effect where outliers also get removed. Finally, the preprocessed data is merged with the sensor data to create the final dataset.

3.2. Materials

This research used certain development tools to design neural networks and to test and evaluate them. These are widely used tools used by industry experts and researchers alike. The research used Python for all the coding work required and that included the use of certain Python libraries namely.

- Pandas
- NumPy
- BeautifulSoup
- Datetime
- Holidays
- Matplotlib
- Scikit-learn

As for the development environment a Jupyter notebook environment provided by Kaggle was used to train and evaluate the models. The reason behind this is because Kaggle can run multiple environments at the same time which allows for concurrent runs to test different hyperparameters. The range of hyperparameters to test was large and it would have taken more than 12 h to conduct the hyperparameter search if tested sequentially. Therefore, the ranges were split into different environments and run concurrently, since Kaggle does not allow a single instance to run for more than 12 h. Kaggle made this process much faster and efficient than running the code on a local environment.

3.3. Research methodology

3.3.1. General approach

Every Machine Learning problem is unique and has different requirements and therefore the solutions might vary and by extension the approach. However, for this research the general approach was to rely on external data only that are generally available anywhere and not rely on the specific sensors coming from the TES-AC. The main reason behind this is to allow this research to have a higher reproducibility chance and to make the solution more general and more applicable elsewhere. Relying on specific sensor data coming from the TES-AC would only allow similar TES-AC models to apply this solution. Regardless, sensor data from the TES-AC are applicable to predict maintenance related values that might be useful to predict machine component failure, or anomaly detection in sensor readings. Regarding the water volume that will be required to chill and get used by the building for the next day, external factors are important. The external weather affecting the surrounding environment of the building might have an impact on how hard the TES-AC has to work that day to maintain optimal indoor temperatures. This research focuses on a TES-AC that is used to cool an office building with a small shopping complex occupying the lower grounds. The occupancy changes based on which day of the week it is and whether it is public holiday or not. Therefore, knowing which day of the week it is, and whether that day is a public holiday or not and finally the occupancy number are all crucial for this research.

Finding a suitable and reliable model based on these factors is not an easy task and the model would need to discover a complex pattern to produce reliable predictions. Neural networks have been proven to find complex patterns that other linear machine learning models do not discover. Multi-layer perceptron models are a good choice for this type of problem. In preliminary stages of this research, when the problem was to predict an exact water volume, a Multilayer perceptron regressor was used as the model of choice, however, that was eventually replaced with a Multilayer perceptron classifier after the problem was re-defined as a classification problem.

Linear machine learning models and kernel methods have also been tested briefly on the data, but their results were not satisfactory despite their training time being very short and fast to test. The pattern between the input data and the output is not very straightforward and therefore a deep learning neural network model is more appropriate in this case.

3.3.2. Dataset imbalance and oversampling

The dataset contained 10 labels, of which 3 of them occupied 79% of the dataset while the remaining 7 labels occupied only 21% of the dataset. This indicated an imbalanced dataset that would result in the model mostly favoring predictions to the more dominant labels. To overcome this issue an oversampling approach was taken. Oversampling is a common technique used when there are labels or classes that are a minority but predicting them is important and they cannot be overlooked. The approach essentially duplicates examples of the minority labels in the dataset to make them occur more frequently without compromising the integrity of the data.

Prior to oversampling the data, 15% of the dataset was randomly removed to represent the test set on which the data would be finally evaluated on. The remaining dataset was resampled to increase the instances of the minority labels before the training process.

3.3.3. Model configuration

A very important part of any machine learning model is the configuration of the hyperparameters. Hyperparameters differ based on the chosen algorithm and in the case of this research the MLP classifier implemented in the widely used machine learning library scikit-learn was used. The details of how the algorithm works and an in-depth discussion of each hyperparameter both fall outside the scope of this work, and therefore will not be elaborated upon further. This research develops a deep learning MLP classifier that provides decent predictions to the water charging load in a TES-AC and evaluates how these predictions could be useful for facility managers using TES-ACs.

There were 5 main hyperparameters that were fine-tuned until acceptable predictions were accomplished. Those were the maximum number of iterations that the solver would run before it would stop. However, if convergence occurred before that, the model would stop training. The hidden layer size is another hyperparameter that was focused on, and it is one of the most important hyperparameters to be fine-tuned in any neural network. However, finding the optimal hidden layer size is extremely tricky and time consuming.

The three other hyperparameters were the activation function for the hidden layer, the solver type and finally the alpha value. To avoid manually changing the value of the hyperparameters and training the models separately, which would be very time-consuming, a grid search was used instead. A grid search takes in a range of values for each hyperparameter and trains the model several times with all the different combinations from the list provided and returns the best hyperparameters found using a scoring metric specified.

In an effort to come up with the best parameters a wide range of values were given for each hyperparameter. The maximum iteration number was set to 60,000 to give enough time for complex networks to converge. The hidden layer sizes were given a range starting from (2,1) till (300,1) indicating 2 units in just one layer to 300 units in one layer. Moreover, two hidden layers with 2 units–300 units were also tested, denoted by hidden layer sizes (2,2) to (300,2). All three different solvers were tried including the adam solver, sgd and the lbfgs solver. The activation function had 4 different variations, which were all tested including the Identity, Logistic, Tanh and Relu functions. Finally, the given range was used for the alpha value or learning rate (0.1,0.32,1.0,1e-3,1e-4,1e-5,1e-6,1e-7,1e-8,1e-9,1e-10).

The scoring metric used for the search evaluation was the weighted precision and 4 cross validations were used for each combination of parameters. The training process was extremely long, and the best parameters were noted down for further usage. Finally, after the grid search found the best parameters, these were used as the final hyperparameters for each model. The best parameters were a hidden layer size consisting of a single layer of 250 units, the tanh activation function, a learning rate of 0.01 and the LBFGS solver.

3.3.4. Occupancy Detection model

To train the water charge load models, occupancy was used as one of the features during the training process. However, unlike the other features it is not possible to get the value for the next day until the day has already passed. All the weather features can be obtained from a public API and the day features are easily obtainable since the names of the days or the public holidays will not change. Occupancy on the other hand needs to be predicted first for every new day. Similarly, a machine model needs to be developed to predict the occupancy.

The features used for this model were the same as the water charging load model with the exclusion of the water volume data. Therefore, to train the model the features included the weather data, the days data and the previous occupancy data to compare against. This was modelled as a regression problem to predict the number of occupants for the next day. The same preprocessing of data was applied to the data of this model and the same min-max scaling technique was used.

Following the same model tuning method used for the other two models, the grid search algorithm was used to determine the best parameters from a list of values given. The same range of values was used except for the hidden layer size where the maximum increased to (500,1) and (500,2) respectively. The maximum iterations were also set to 20,000 runs. After the grid search was completed, the best parameters consisted of a hidden layer size consisting of a single layer of 310 units, the LBFGS solver, and a learning rate of 1e-5.

4. Results

4.1. Water charging load models

We first evaluate the model predicting the water charging load for the next day. When evaluating any classification model there are several metrics that could be considered, and it usually depends on the problem. Precision is calculated by dividing the number of true positives (TP) with the total number of TP and false positives (FP). Recall is calculated by dividing the TP with the total of TP and false negatives (FN). Each one of these metrics might be more useful than the other depending on the scenario, however, the model was evaluated using a combination of both precision and recall known as the f1-score. The f1-score considers both precision and recall and is a good metric to evaluate a classification model on.

There are 2 typical versions of the f1-score in this multi-class context, depending on the averaging technique used. The simple average simply sums all of the f1-scores (one for each label) and divides this by the number of labels, without considering how many occurrences of each label there are. This method of averaging is called the macro-average value. A better way to calculate the average f1-score would be to consider the weight of each label which is called the support value. The more a label occurs in the data the higher support value it gets, and the weighted average considers the weight of each label to provide more reliable evaluations.

The evaluation metrics of the model can be seen in Table I. The model scored 93% macro average precision, and 96% weighted average precision. It scored a 93% macro average recall value and weighted average recall value. Finally, it scored 92% macro average f1-score and 94% weighted average f1-score. It is to be noted that these metrics are based on the predictions done on the 15% of the dataset that was taken out initially for testing purposes. This makes these results the most unbiased and reliable ones to use, and all the other metrics obtained while training were not included in this paper as they were used only to refine the model further.

It has to be noted however that the prediction provides a label for the facility managers that gives them a small range of values to choose from. This method will not provide them with an exact volume to charge but should get them close enough to it.

Fig. 2 shows a stacked bar chart indicating the predictions made by the model. Each bar indicates one of the labels and the stacks inside are divided based on the labels that got predicted as the label on the x-axis. As seen in the chart, the model predicted nearly perfectly for most of the labels, except for the label Low_2. Overall, the prediction accuracy of the model is considerably good, and the results are promising.

4.2. Occupancy model evaluation

The occupancy model was evaluated in a different way since it was a regression prediction and not a classification. Regression is tricky to evaluate because in many cases the evaluation metrics might not be giving a good result, but the prediction could actually still be useful and therefore the model is suitable for usage. Therefore, for regression problems the margin of error is measured. There are two main metrics to measure the error which are the mean squared error (MSE) and the root mean squared error (RMSE).

RMSE is used to evaluate this model as it returns the average deviation between the prediction and the target in the same units as the values in the dataset. Given that the occupancy number is in the thousands, the margin of error is acceptable as long as it is in the hundreds. After training and evaluating the model, the RMSE was found to be 160, which is acceptable, considering the fact that the occupancy number is usually between 2000 and 3000. Fig. 3 shows a scatter plot depicting the actual points in green and the predicted points in purple. It can be seen that the model still manages to get close predictions even when the actual values deviate from the normal average and it does manage to get close predictions to the outlier values as well.

5. Discussion

This research presented a machine learning solution that aims to predict the water charge load that will be required to cool a commercial building the next day. The research focused on predicting the load using external data that is found everywhere and not tied down to a specific sensor type or depending on the model of TES-AC used. The research utilized the power of MLP neural networks in finding complex patterns in data and developed two models that followed a similar design to predict the final output.

The problem was divided into smaller problems to make solving it less complex and to provide a more reliable result. The output of this research could be very beneficial for facility managers handling TES-AC systems as it can greatly reduce their energy consumption. One of the main issues that TES-ACs face is not knowing how much water will be needed for the next day, and as such more or less water might be charged and stored in the water tanks. Storing less water means they have to use the chiller during on-peak hours and increase their energy consumption and storing more means they would have wasted energy cooling more water than they needed. However, a similar solution to the one proposed in this research can provide a more accurate way of knowing what range of water volume the facility managers need to charge for that day which would reduce the instances where the water charged is too low or too high.

5.1. Limitations

The proposed solution provided satisfactory predictions however it is not a perfect solution but a step in the right direction. One of the main limitations of the proposed model is that it depended on a large amount of data to make predictions. When trained with anything less than a year's data the results were unsatisfactory and therefore more data was needed. This might not always be viable as some TES-AC facilities have not been running for many years, therefore making this solution infeasible for them. The model would need further fine-tuning and improvements to be able to make reliable predictions using less data to train.

Another limitation is that the data was evaluated using the grid search cross-validation method only. Other network parameterizations and hyperparameter search procedures, more extensive search ranges, and finer granularities, could be considered in future work, in order to obtain even better parameter configurations.

5.2. Future directions

In the future this research could be further improved by allowing the designed model to train using smaller datasets, so that it can get trained using a few months only instead of years. A more thorough hyperparameter search can also be conducted using different types of algorithms like global stochastic optimization methods and with much bigger range values. In addition to that, other external features could be added to the data to improve the training process and provide better predictions., for example, more weather variables like wind speed, whether a day is cloudy/sunny/etc. Furthermore, if the model could be improved to remove the reliance on occupancy data it would be a better solution, as the model would not need to rely on a predicted value to make a prediction.

6. Conclusion

Thermal energy storage air conditioning (TES-AC) systems are considered to be a promising solution for reducing energy consumption and greenhouse gas emissions in commercial buildings. However, concerns have been raised regarding their operational efficiency, as the way in which they are used could either save a lot of money or increase costs substantially. Many stakeholders are hesitant to make the shift to TES-ACs because they cannot guarantee their operational efficiency and are looking for short-term solutions.

Despite these concerns, TES-ACs can be very beneficial to the environment in the long run. An efficient operation of TES-ACs will reduce costs in many commercial buildings and also contribute positively to the environment. In a time when the global climate crisis is at its peak, the need for technological innovations to ease the burden on our planet should be clear.

Deep learning has been used to solve many complex real-life problems and optimizing TES-ACs should be one of them. Deep learning neural networks can adapt to many TES-ACs and provide accurate predictions that eventually lead to the usage of machine learning in TES-ACs, a standard that would result in more people shifting to TES-ACs to reduce their costs and resulting in a green choice of ACs.

The novelty of this research lies in the development of a machine learning model that can predict the water volume required to be chilled for the TES-AC plant to satisfy the cooling needs of the building for the next day. Unlike other studies that aim to create or invent a new machine learning algorithm, the present study focuses on applying existing machine learning algorithms in a novel way to solve a specific problem faced by facility managers operating TES-ACs. The proposed machine learning model utilizes input variables, such as weather data and information about the day, to make accurate predictions about the water volume required to satisfy the building's cooling needs. The research aimed to address the challenge faced by facility managers in determining the appropriate water volume that needs to be chilled to minimize electricity bills and carbon emissions, while also taking into account the variable electricity charges during peak and off-peak hours.

One of the key aspects that makes this research unique is its focus on creating a solution that can be easily adopted and repurposed for other TES-AC plants, without requiring a very specific sensor input that might not be available in a different TES-AC plant. The model was designed to avoid reliance on sensor data specific to the TES-AC plant, which makes the proposed solution widely applicable and adaptable to other TES-AC plants across various locations.

The machine learning model developed was a Multilayer Perceptron neural network, which provided the best prediction result. Instead of predicting an exact water volume, the model predicts a specific water range as a target output, providing facility managers with a small set of ranges to choose from. This approach ensures that the facility managers can use their expertise to determine the best water volume within a range, thereby minimizing errors. The final model achieved an accuracy of 93.4%, which is a significant result that shows that there is potential in this research field.

Author statement

Conceptualization: Mirza Rayana Sanzana, Mostafa Osama Mostafa Abdulrazic, Jing Ying Wong; Data curation: Mirza Rayana Sanzana, Mostafa Osama Mostafa Abdulrazic, Jing Ying Wong; Formal analysis: Mirza Rayana Sanzana, Mostafa Osama Mostafa Abdulrazic, Tomas Maul; Funding acquisition: Mirza Rayana Sanzana, Jing Ying Wong; Investigation: Jing Ying Wong, Tomas Maul, Chun-Chieh Yip; Methodology: Mirza Rayana Sanzana, Mostafa Osama Mostafa Abdulrazic; Project administration: Jing Ying Wong, Tomas Maul; Resources: Jing Ying Wong; Software: Mirza Rayana Sanzana, Mostafa Osama Mostafa Abdulrazic; Supervision: Jing Ying Wong, Tomas Maul, Chun-Chieh Yip; Validation: Mirza Rayana Sanzana, Mostafa Osama Mostafa Abdulrazic, Jing Ying Wong, Tomas Maul; Visualization: Mirza Rayana Sanzana, Mostafa Osama Mostafa Abdulrazic; Roles/Writing - original draft: Mirza Rayana Sanzana; Writing - review & editing: Mostafa Osama Mostafa Abdulrazic, Tomas Maul, Jing Ying Wong.

Table 1
Evaluation metrics for the developed model for charging water load prediction.

Model	Precision		Recall		F1-Score	
Main Data	0.93 **Macro Avg.**	0.96 **Weighted Avg.**	0.93 **Macro Avg.**	0.93 **Weighted Avg.**	0.92 **Macro Avg.**	0.94 **Weighted Avg.**

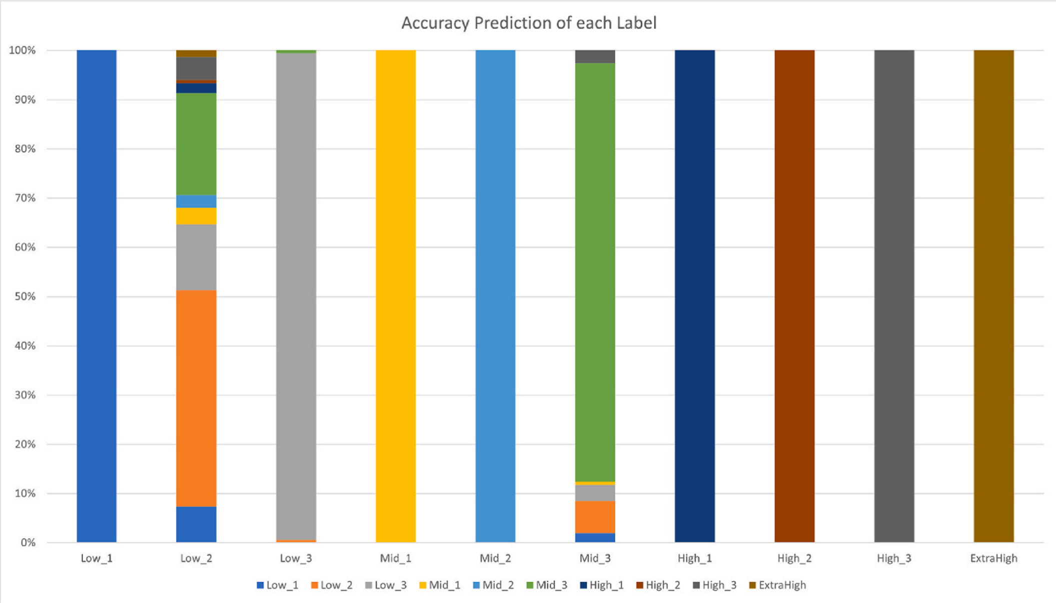


Fig. 2. Stacked Bar Chart indicating the Accuracy Prediction.

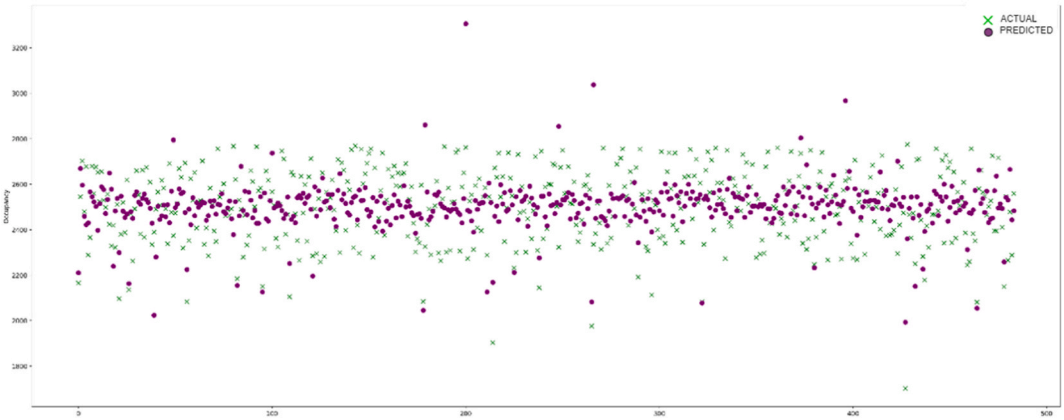


Fig. 3. Scatter plot of the actual and predicted values for the occupancy detection.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Mirza Rayana Sanzana reports administrative support and article publishing charges were provided by University of Nottingham.

Data availability

The data that has been used is confidential.

References

[1] A. MacMillan, Global warming 101. (NRDC), 10 2019, https://www.nrdc.org/stories/global-warming-101#weather, 2016, March 11.
[2] Y. Dong, M. Coleman, S.A. Miller, Greenhouse gas emissions from air conditioning and refrigeration service expansion in developing countries, Annu. Rev. Environ. Resour. 46 (2021) 59–83, https://doi.org/10.1146/annurev-environ-012220-034103.
[3] A. Al-Abidi, A. Mat, S.B. Sopian, K. Sulaiman, M.Y. Lim, C. H, A. Th, Review of thermal energy storage for air conditioning systems, Renew. Sustain. Energy Rev. 16 (8) (2012) 5802–5819, https://doi.org/10.1016/j.rser.2012.05.030.
[4] I. Sarbu, C. Sebarchievici, A comprehensive review of thermal energy storage, Sustainability 10 (1) (2018) 191, https://doi.org/10.3390/su10010191.
[5] I. Bajaj, X. Peng, C.T. Maravelias, Material screening for thermochemical energy storage in solar power systems, Computer Aided Chemical Engineering 50 (2021) 179–184, https://doi.org/10.1016/B978-0-323-88506-5.50029-2.

[6] Y. Sun, S. Wang, F. Xiao, D. Gao, Peak load shifting control using different cold thermal energy storage facilities in commercial buildings: a review, Energy Convers. Manag. 71 (2013) 101–114, https://doi.org/10.1016/j.enconman.2013.03.026.

[7] I. Dincer, On thermal energy storage systems and applications in buildings, Energy Build. 34 (4) (2002) 377–388, https://doi.org/10.1016/S0378-7788(01)00126-8.

[8] A. Mehari, Z.Y. Xu, R.Z. Wang, Thermal energy storage using absorption cycle and system: a comprehensive review, Energy Convers. Manag. 206 (2020), https://doi.org/10.1016/j.enconman.2020.112482.

[9] E. Guelpa, V. Verda, Thermal energy storage in district heating and cooling systems: a review, Appl. Energy 252 (2019), https://doi.org/10.1016/j.apenergy.2019.113474.

[10] K. Amarasinghe, D. Wijayasekara, H. Carey, M. Manic, D. He, W.-P. Chen, Artificial neural networks based thermal energy storage control for buildings, in: 41st Annual Conference of the IEEE Industrial Electronics Society, IEEE, Yokohama, Japan, 2015, https://doi.org/10.1109/IECON.2015.7392953.

[11] C.M. Eastman, C. Eastman, P. Teicholz, R. Sacks, K. Liston, BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors, John Wiley & Sons, 2011.

[12] P. Fowler, Building information modeling: B.I.M. And building lifecycle management (BLM). https://www.petefowler.com/blog/2014/06/09/bim-and-blm, 2014.

[13] C.K. Lee, H.Y. Chan, C.Y. Poon, K.M. Yip, P.H. Yuen, Smart BIM-AM journey to green buildings, in: IOP Conference Series: Earth and Environmental Science, vol. 290, IOP Publishing Ltd, Prague, 2019, https://doi.org/10.1088/1755-1315/290/1/012050.

[14] C. Ventures, Discover How BIM Is Implemented in Each Step of the Construction Value Chain, 2021, 09 2, 202209, https://www.cemexventures.com/discover-how-bim-is-implemented-in-each-phase-of-the-construction-industry/.

[15] Y. Jeong, A study on the BIM evaluation, analytics, and prediction (EAP) framework and platform in linked building ontologies and reasoners with clouds, Advanced BIM Applications in the Construction Industry 2018 (2018), https://doi.org/10.1155/2018/5478381.

[16] B. Haines, The benefits of lifecycle BIM for facility management. Retrieved 11 2019, from fm: systems. https://fmsystems.com/blog/the-benefits-of-lifecycle-bim-for-facility-management, 2016.

[17] M.R. Sanzana, M.O. Abdulrazic, J.Y. Wong, T. Maul, C.-C. Yip, Analysis of machine learning techniques for predictive maintenance in cooler condition, in: The International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS) 2022. Penang, IEEE, Malaysia, 2022, https://doi.org/10.1109/ISPACS57703.2022.10082814.

[18] M.R. Sanzana, M.O. Abdulrazic, J.-Y. Wong, T.H. Maul, C.-C. Yip, Effects of external weather on the water consumption of thermal-energy-storage air-conditioning system, Energy Nexus 10 (2023), https://doi.org/10.1016/j.nexus.2023.100187.

[19] H. Nguyen, Y. Makino, Y. Lim, Y. Tan, Short-term prediction of energy consumption of air conditioners based on weather forecast, in: 2017 4th NAFOSTED Conference on Information and Computer Science. Hanoi, IEEE, Vietnam, 2017, https://doi.org/10.1109/NAFOSTED.2017.8108063.

[20] G.M.P. Revati, U. Suryawanshi, A. Sheikh, S. Bhil, Load profile prediction in smart building using data driven approaches, in: 2021 31st Australasian Universities Power Engineering Conference (AUPEC), IEEE, 2021, https://doi.org/10.1109/AUPEC52110.2021.9597702.

[21] C. Li, Z. Ding, D. Zhao, J. Yi, G. Zhang, Building energy consumption prediction: an extreme deep learning approach, Energies 10 (10) (2017) 1525, https://doi.org/10.3390/en10101525.

[22] M.R. Sanzana, T. Maul, J.Y. Wong, M.O. Abdulrazic, C.-C. Yip, Application of deep learning in facility management and maintenance for heating, ventilation, and air conditioning, Autom. ConStruct. 141 (2022), https://doi.org/10.1016/j.autcon.2022.104445.

[23] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444, https://doi.org/10.1038/nature14539.

[24] I.C. Education, Deep learning. (IBM). https://www.ibm.com/cloud/learn/deep-learning, 2020, 05 01.

[25] Mathworks. (n.d.). The Mathworks: Deep Learning Toolbox. Retrieved 01 29, 2022, from https://www.mathworks.com/products/deep-learning.html.

[26] J. Pingel, Introduction to deep learning: machine learning vs, deep learning (2017), 01 25, 2022, https://www.mathworks.com/videos/introduction-to-deep-learning-machine-learning-vs-deep-learning-1489503513018_b.html.

[27] F. Rosenblatt, The perceptron: a probabilistic model for information storage and organization in the brain, Psychol. Rev. 65 (6) (1958) 386–408, 10, 1037/h0042519.

[28] S. Abirami, P. Chitra, Chapter Fourteen - energy-efficient edge based real-time healthcare support system, in: The Digital Twin Paradigm for Smarter Systems and Environments: the Industry Use Cases, vol. 117, 2020, pp. 339–368, https://doi.org/10.1016/bs.adcom.2019.09.007.

[29] H. Taud, J. Mas, Multilayer perceptron (MLP), in: Geomatic Approaches for Modeling Land Change Scenarios, 2017, pp. 451–455, https://doi.org/10.1007/978-3-319-60801-3_27.

[30] F. Murtagh, Multilayer perceptrons for classification and regression, Neurocomputing 2 (5–6) (1991), https://doi.org/10.1016/0925-2312(91)90023-5.

[31] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake Vanderplas, Arnaud Joly, Brian Holt, G. Varoquaux, API design for machine learning software: experiences from the scikit-learn projectLars Buitinck, in: ECML PKDD Workshop: Languages for Data Mining and Machine Learning, Cornell University, 2013, pp. 108–122, https://doi.org/10.48550/arXiv.1309.0238.

[32] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Perrot, Scikit-learn: machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[33] C. Bento, Multilayer perceptron explained with a real-life example and Python code: sentiment analysis, 07 11, 2022, https://towardsdatascience.com/multilayer-perceptron-explained-with-a-real-life-example-and-python-code-sentiment-analysis-cb408ee93141, 2021.

[34] A. Meyer-Baese, V. Schmid, Chapter 4-The Wavelet Transform in Medical Imaging. In Pattern Recognition and Signal Analysis in Medical Imaging (Second Edition), 2014, pp. 113–134, https://doi.org/10.1016/B978-0-12-409545-8.00004-2.

[35] T. Menzies, E. Kocagüneli, L. Minku, F. Peters, B. Turhan, Sharing data and models in software engineering, in: Chapter 24-Using Goals in Model-Based Reasoning, 2015, pp. 321–353, https://doi.org/10.1016/B978-0-12-417295-1.00024-2.

[36] S. Petruseva, V. Žujo, V. Zileska Pancovska, Neural network prediction model for construction project duration, Int. J. Eng. Res. Technol. 2 (11) (2013). https://www.ijert.org/research/neural-network-prediction-model-for-construction-project-duration-IJERTV2IS110333.pdf.

[37] A. Bagnasco, F. Fresi, M. Saviozzi, F. Silvestro, A. Vinci, Electrical consumption forecasting in hospital facilities: an application case, Energy Build. 103 (2015) 261–270, https://doi.org/10.1016/j.enbuild.2015.05.056.

[38] A. Rajith, S. Soki, M. Hiroshi, Real-time optimized HVAC control system on top of an IoT framework, in: 2018 Third International Conference on Fog and Mobile Edge Computing (FMEC). Barcelona, IEEE, Spain, 2018, https://doi.org/10.1109/FMEC.2018.8364062.

[39] S. Assaf, I. Srour, Using a data driven neural network approach to forecast building occupant complaints, Build. Environ. 200 (2021), https://doi.org/10.1016/j.buildenv.2021.107972.

[40] R. Olu-Ajayi, H.A. Alaka, Building energy consumption prediction using deep learning, in: Environmental Design and Management International Conference, 2021.

[41] X. Zhao, T.-H. Wang, C. Peng, Automatic room type classification using machine learning for two-dimensional residential building plans, Proceedings of the 40th Conference on Education and Research in Computer Aided Architectural Design in Europe (eCAADe 2022) 2 (2022) 593–600. Retrieved from, http://papers.cumincad.org/cgi-bin/works/paper/ecaade2022_158.