Grenze International Journal of Engineering and Technology, June Issue

A Holistic Federated Learning Approach Employing Neural Networks for Climate Control within Smart Buildings

Caleb Stephen[1], Dr. Chitra R[2], Richie Suresh Koshy[3] and Joel Mathew[4]
[1-4]Karunya Institute of Technology and Sciences, Coimbatore, India
Email: calebstephen@karunya.edu.in, chitrar@karunya.edu, richiesuresh@karunya.edu.in, joelmathew20@karunya.edu.in

***Abstract*— Comfort is a universally understood concept of physical and mental well-being. Maximizing comfort is of paramount importance to everyone be it at home or in a workspace. Smart Buildings offers the technology to facilitate a greater enhancement of comfort in every occupied space. This research delves into the realm of intelligent climate control within smart buildings by leveraging new machine-learning techniques. This research proposes a holistic approach to climate control by incorporating federated learning, to learn the user's preferences across many buildings. The federated learning framework ensures that data privacy is maintained at all times. A global neural network regression model has been created that has trained on a wide variety of input data. The study contributes to the evolving field of smart building technologies, offering a sustainable and efficient solution for enhanced living and climate control.**

***Index Terms*— federated learning, climate control, regression, feature engineering, collaborative learning, data privacy, smart homes, heat index, deep learning, TensorFlow, flower, smart buildings**

I. Introduction

If something is not "smart", that means that particular entity has not been fully developed to the full potential of today's revolutionary technology. This paper seeks to introduce a new approach to climate control within smart spaces. The concept of smart is not new, it has been around since the early 1980's, but it has never been as tangible as it is today. In particular, time spent at home has increased by significant amounts when compared to before the COVID-19 pandemic, as houses have also become places of work. A study conducted in Japan [1] in 2021, showed the difference between time spent at home before and after the pandemic. On average people spent at least 2.5 hours more time at home across various age groups.

Since a paper authored in 2015 [2] in New Zealand, estimates that roughly 73.76% of one's time is spent within the house, this research work will focus on the implementation of a climate control system with a house as a placeholder since other spaces within smart buildings can also have the system integrated into.

Berlo et al. [3] defined a smart home as: "A home or working environment, which includes the technology to allow the devices and systems to be controlled automatically, may be termed a smart home". I would argue that this definition can be expanded to a building as well.

Grenze ID: 01.GIJET.10.2.233_1

Using Machine Learning to tackle the problem of predicting indoor temperature and even determining optimal temperature is not novel. However, most approaches are designed for centralized HVAC systems which are not commonly used in residential houses in India. HVAC systems are commonly used in offices and other commercial spaces like shops, etc. The integration of a smart system for climate control and more specifically for temperature control and monitoring is much easier when dealing with a HVAC system.

Similarly, it is comparatively easier to install and integrate into a smart building because at the very least it is designed with IoT in mind, making installation easier. The simplest scenario would be to have the building designed with intelligent control systems in mind. This begs the question, what about existing buildings? According to a public news report [4] only 24% of homes in Delhi, India have an air conditioning system. The beauty of the proposed workflow and infrastructure is that it can be easily integrated into any building without any major structural changes. This brings up the possibility of creating any building/space into a smart space/building. This is an intriguing possibility that is very much a reality.

Some of the other approaches like the research done in 2019 [5], only focus on the development of a model for the prediction of indoor temperature for a single house, rather than a global model that has been trained on a wide range of homes data and can be deployed anywhere. Another similar paper [6] authored in 2019 again focuses on a single two-story house.

Our proposed solution seeks to employ Machine Learning which can identify these patterns as they occur in real time and which is not confined to a single home. However, the moment when data collection is done within a home, the question of data privacy and user privacy comes into question. Conventional Machine Learning methodologies do not spend much effort to ensure data privacy as the raw data is sent to a server.

Traditional methods to tackle this problem would include using Dimensionality Reduction techniques like Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), etc, and combining them with a level of encryption. The problem with these traditional machine learning approaches is that the model's performance is subpar when compared to new techniques like our proposed federated learning solution. It no longer becomes prudent to use such methods when the performance is poor. This is why we have chosen to employ federated learning for collaborative learning.

Federated Learning is a new type of machine learning that comes under the category of unconventional hybrid machine learning models. It works to ensure the creation of a global optimal model that has aggregated weights from all of its participating clients in a secure manner to ensure privacy. The dataset used for training was obtained from NASA's Prediction of Worldwide Energy Resources (POWER) project [7], which gave timestamped temperature and humidity data from 2018-2023 (November) for six different locations.

II. Methodology

A. Dataset Acquisition and Pre-processing

The data curation and acquisition was a long drawn-out process. It was not possible to get hourly timestamped temperature and humidity data from common open-source dataset collections like Kaggle. It is probably because nobody has required it or that it is present with private companies that employ smart devices within homes.

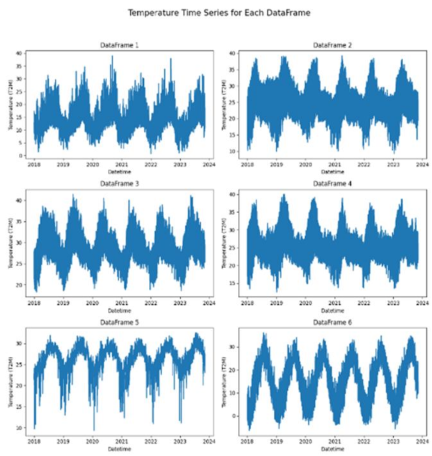

Fig. 1. Time-series distribution for each dataset

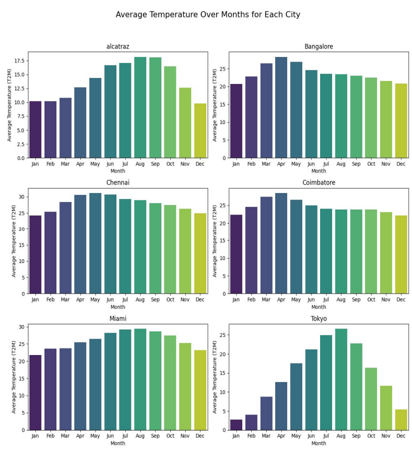

Fig. 2. Distribution of average temperature by month

Our requirements for data were specific. We needed temperature and relative humidity data from at least six locations spanning from 2018 to November of 2023. After research and searching it was determined that indoor temperature and relative humidity would be impossible to obtain. Due to this, we had to settle with outdoor hourly timestamped data of temperature and relative humidity.

Firstly to get the dataset, we looked to NASA's Prediction of Worldwide Energy Resources (POWER) project [7], which allowed us to input the latitude and longitude and then choose the features that we wanted in the dataset. We chose six locations namely, Chennai (India), Coimbatore (India), Bangalore (India), Hyderabad (India), Madurai (India), and Mumbai (India). Each of the six datasets was from Jan 1st, 2018 to November 7th, 2023. This resulted in the original dataset having a total of 51,288 data points in each of the 6 datasets.

After feature engineering which we will discuss next, the number of features increased from 4 (including timestamp) to 7.

Fig. 1, shows the time-series distribution of temperature for all six datasets from 2018 to 2024 can be seen. The distribution which clearly varies sometimes subtlely or drastically shows the difficult task that the global model has to overcome.

Fig. 2, shows the average temperature each month in three of the cities we are studying namely, Chennai, Coimbatore and Bangalore over the course of the years selected. In addition to that we have taken three other random cities, Tokyo, Miami and Alcatraz to see the distribution in other places as well.

B. Feature Engineering

A major issue present with the base features is that it does not account for the temperature difference between indoors and outdoors. Not addressing this problem would prove to be detrimental when the model is in a deployment environment. To tackle this, one calculated field called Heat Index is introduced. Using Heat Index, the other calculated fields namely Current Comfort Level and Adjusted Temperature are created.

Heat Index: This metric was developed in 1979 by Robert G. Steadman [8], to identify the feels-like temperature or skin temperature. This is the closest to any indoor temperature reading. Hence the calculated Heat Index value in Celsius is taken as the current temperature and is the target variable for the regression model. The following equation for calculating heat index is from Lans. P. Rothfusz[9][10]. The equation used to calculate the Heat Index is shown in Eq. 1.

$$Hi = c_1 + c_2 T + c_3 R + c_4 TR + c_6 R^2 + c_7 T^2 R + c_8 TR^2 + c_9 T^2 R^2 \quad (1)$$

Where:
HI = Heat Index in Fahrenheit
T = Temperature in Fahrenheit
R=Relative Humidity (percentage value)

The following categories present in Fig. 3, of danger/comfort based upon the heat index value are taken from the United States National Weather Service [11].

Celsius	Notes
27–32 °C	Caution: fatigue is possible with prolonged exposure and activity. Continuing activity could result in heat cramps.
32–41 °C	Extreme caution: heat cramps and heat exhaustion are possible. Continuing activity could result in heat stroke.
41–54 °C	Danger: heat cramps and heat exhaustion are likely; heat stroke is probable with continued activity.
over 54 °C	Extreme danger: heat stroke is imminent.

Fig. 3. Current comfort level

- *Current Comfort Level*: Based upon the Heat Index value that was calculated, the current comfort level was determined. Some of the categories based on heat index value are Comfortable, Warm/Comfortable, Some Discomfort, Great Discomfort, Dangerous, Extreme Cold, Very Cold, Cold and Cool.
- *Simulated Comfortable Temperature*: The goal of this field is to simulate what a user would set as a comfortable temperature. As this temperature would be dependant on how the user "feels", this field is dependant upon the heat index value as it is the closest to feels like temperature. In 1987, the WHO determined that indoor temperatures between 18 to 24 degrees celsius is the comfortable range [12]. A simulated

comfortable temperature is generated using the “np.random.normal” function from the numpy library, which draws random values from a normal distribution centered around the heat index value. The scale parameter of the normal distribution controls the spread of simulated values, adding an element of randomness to the comfort simulation. The “np.clip” function is applied to the simulated comfortable temperature, ensuring it falls within the defined comfortable range.

C. Theory of Federated Learning and It’s Contextual Application within the project

Federated learning can be defined as a workflow within which a central aggregator facilitates multiparty computation which in turn solves many of the issues with traditional machine learning [13]. A global model is created after training on the various client datasets. The goal is to minimize the following objective function to minimize the average training loss for all the clients:

$$minimize\ F(w) = \frac{1}{c}\sum_{i=1}^{c} F_i(w_i) \qquad (2)$$

Where:
F(w) is the global objective function (loss) to be minimized
C is the total number of clients (devices)
w represents the global model parameters
w_(i) is the local model parameters of client i
F_i (w_i) is the local loss function of client i
In our application of smart building/home climate control, we have chosen only four features namely timestamp, temperature, relative humidity and heat index so, in our case Horizontal Federated Learning is preferred since the client’s datasets have the same features.

D. Architecture having a two-step hierarchy

The federated learning architecture that is being proposed has been designed with two major components which in turn would result in the developement of an efficient global model. They are:

1. *Server:* The server holds the global model that is updated each epoch with the aggregated weights from each of the clients present.
2. *Client:* Each home that is present and is participating in the training process can be referred to as a client.

This type of architecture has been proposed in several research papers. One of those papers is the paper authored in 2021 [14], which delved into the concept of multiple clients collaborating to solve machine learning problems.

E. Deep Learning Regression Model

The TensorFlow regression model is designed to predict the simulated comfortable temperature based on input features. The model is built using the TensorFlow framework, which provides a robust platform for developing and training machine learning models.

The model architecture consists of a Feedforward Neural Network with multiple layers. The input layer incorporates features like temperature, relative humidity, and timestamp information. Subsequent hidden layers, featuring rectified linear unit (ReLU) activation functions, capture complex relationships within the data. The output layer provides the regression prediction for the simulated comfortable temperature.

The neural network architecture comprises three layers which are an input layer, the hidden layers and the output layer. The input layer accepts the standardized input features. The timestamp is extracted into Year, Hour, Day and Month individual features.

The Hidden Layers consists of two densely connected layers with 64 and 32 neurons, respectively, and Rectified Linear Unit (ReLU) activation functions. These layers facilitate the model in learning complex relationships within the data.

The output layer consists of a single neuron with a linear activation function, providing the continuous prediction for the simulated comfortable temperature in Celsius.

The model is trained for 50 epochs with a batch size of 32. These hyperparameters can be fine-tuned based on further experimentation and performance evaluation.

The model's performance is assessed on the validation set using mean squared error (MSE) as the evaluation metric. Lower MSE values indicate better predictive accuracy. This aforementioned architecture is illustrated in Fig 4.The model uses an Adam optimizer with a learning rate of 0.001. The training process was carried out for 50 epochs to ensure convergence.

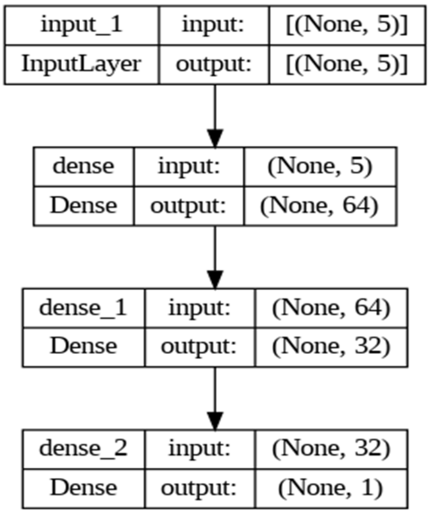

Fig. 4. Model architecture

F. Proposed Solution

The proposed solution is two-fold. One is that data aquisition part. For the purpose of this experiment the data has been taken from NASA's POWER project. For an actual deployment, the data would preferably be taken from a few homes.

The second prong is the federated learning architecture. We have the server which holds the global model and then the participants homes which we will now start referring to as clients.

For the current experiment, a single server and 6 clients have been chosen. The clients all participate in the training process which in turn help to augment and improve the performance of the global model, whose goal is to be able handle data from any home as the model has learnt enough diverse patterns.

Since federated learning is a complex mechanism, we have chosen a framework called Flower [15] was chosen to run the experiments. Flower is a user-friendly framework and helps in running such experiments.

The workflow is simple. First the global model is initialized by the server which in turn send the initial weights to all the participating homes (clients). The server waits for the training process at each epoch and retrieves the results. The clients train the global model on the datasets and return the updated weights to the central server. The weights are then aggregated by the server which in turn updates the global model. After each epoch the updated global model is shared with all the clients.

This process repeats until the global model converges or a satisfactory performance has been obtained. This workflow has been visualized in Fig. 5.

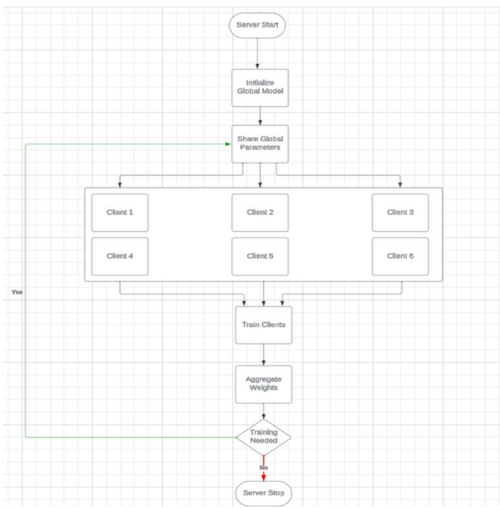

Fig. 5. Federated learning based proposed solution

G. Evalution Metrics

In order to identify whether the model is performing well or not, we employ various performance metrics. For regression tasks the most commonly used metric is Mean Squared Error (MSE).

Mean Squared Error: It refers to the unbiased estimate of error variance: the residual sum of squares divided by the number of degrees of freedom.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - Y_i')^2$$

H. Contrast between the Proposed Federated Learning and Tradional Approaches

A prudent question that should be asked is whether or not federated learning performs better than traditional machine learning methodologies. The thesis before choosing federated learning is the allure of a well performing global model that has learnt patterns from a diverse set of clients which in turn allows for it to be able to handle any kind of input data from a prospective client.

In order to compare and contrast the efficacy of federated learning global model, it is compared to traditional machine learning model benchmarks. In order to quantify and compare the performance, the TensorFlow regression deep learning model is trained on a single client and is then tested on a random subset of the entire dataset consisting of all the clients.

The reason to do this is because, the output of the model shows whether or not the model is its present state can be deployed is a different environment. The results till now suggests that training on a single client does not translate to a universal efficacy in various different global circumstances.

A by-product of this analysis is that is helps us to ascertain how well traditional machine learning regression techniques perform when compared to our proposed federated learning solution.

III. RESULTS

Firstly, we ran the TensorFlow neural network model in a conventional manner in order to identify what is the level of performance that is achieved with traditional machine learning techniques that employ a deep learning neural network for regression purposes.

As expected, the model performance was subpar when compared to federated learning. The best performance from traditional neural network regression models was a mean squared error value of 4.39. This is a very bad score which is indicative of the level of difficulty that this problem poses. As can be seen in Fig 7, it shows that for dataset number 4, which corresponds to Chennai, India, when the model is trained solely on its data but is then tested on the global model, the mean squared error score is a poor 4.74.

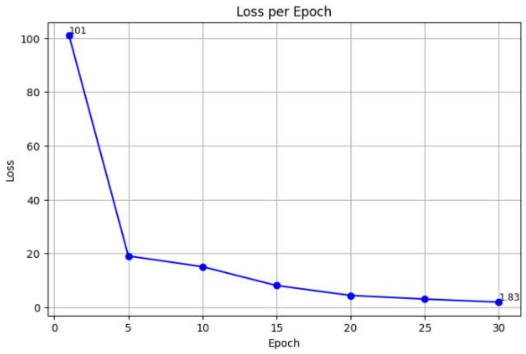

Fig. 6. Loss per epoch for the federated global model

Fig 7, also looks at the results of the federated learning process. Here the results using our TensorFlow deep learning model for regression is shown. Our federated learning solution consisted of six clients each of whose datasets had already been pre-processed and had exploratory data analysis steps done on it like Standard Scaling from the Scikit-learn library. As can be seen in Fig 6, the mean squared error value is 1.83. This value represents a 58.86% improvement in performance.

The comparison of the mean squared error of these models versus the Federated Learning model has been illustrated in Fig 7.

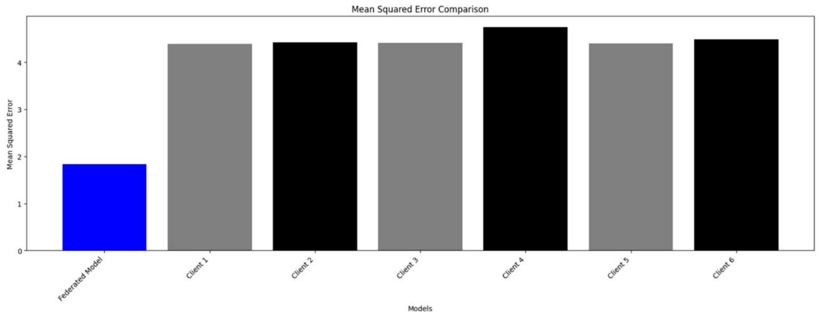

Fig 7. Mean squared error comparison

Fig 6, then explores the drop in the loss value for the federated model per epoch. The federated model was trained for 30 epochs and was tested against the combined dataset to ensure the efficacy of the model.

IV. CONCLUSIONS

The major driving force behind this experiment and project was to ascertain whether or not federated learning and its associated approaches were an improvement over the existing machine learning methodologies that are prevalent and easily available. This aspect of the project has been a resounding success.

This is demonstrated by the fact that even when taking a basic deep learning TensorFlow neural network for regression, with much hyperparameter tuning and optimization, there is a stark difference between the efficacy of the models, which has been shown in the previous section.

Many vital aspects of federated learning, particularly in the realm of data and user privacy are yet to be explored in this context. There are many options to ensure data integrity and confidentiality namely, Differential Privacy, Secure Multi-Party Computation, Secure Aggregation, etc. Even forms of data encryption can be employed.

The goal for future endevours should be to bring the mean squared error close to or below 1. This should be possible with a more keen approach with respect to the deep learning model and its hyperparameters. There are many more models that can be employed. Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) Networks, Gated Recurrent Units (GRU), Transformers, Autoencoders, Deep Belief Neural Networks (DBN), and Radial Basis Function Network (RBFN), are some of the possible deep learning architectures that can be employed in a deep learning setting within a federated learning architecture.

In addition to the possibility of trying other models, the model chosen eventually in a deployment setting must be “lightweight”. This refers to the model’s computational resources and model complexity. In this regard, some of the models that are considered to be lightweight would include, Feedforward Neural Networks (which we are currently using), Radial Basis Function Networks (RBFN), Gated Recurrent Units (GRU’s), and Autoencoders. These models are often preferred when computational resources are limited or when there's a need for faster inference times.

In conclusion, Climate control of smart buildings/spaces can be made much smarter, safer, and simpler using a federated learning approach that factors in easily available data that can be collected by smart devices while at the same time being conscious of the amount of energy being used to help optimize the indoor climate of a home.

REFERENCES

[1] Toyosada, Kanako & Nakagawa, Chika & Mitsunaga, Takehiko & Kose, Hiroyuki. (2021). Effect of the COVID-19 Pandemic on Residential Water Use Behavior in Japan. Water. 13. 3129. 10.3390/w13213129.

[2] Khajehzadeh, Iman & Vale, Brenda. (2015). How do people use large houses?.

[3] Allen, B., van Berlo, A., Ekberg, J., Fellbaum, K., Hampicke, M., & Willems, C. (2001). Design guidelines on smart homes (No. BOOK). COST 219bis guidebook.

[4] https://www.hindustantimes.com/india-news/ac-use-rising-sharply-in-india-will-surpass-all-other-home-appliances-by-2050-report-101698169969128.html (Wikipedia, 2024)

[5] R. Eini and S. Abdelwahed, "Learning-based Model Predictive Control for Smart Building Thermal Management," 2019 IEEE 16th International Conference on Smart Cities: Improving Quality of Life Using ICT & IoT and AI (HONET-ICT), Charlotte, NC, USA, 2019, pp. 038-042, doi: 10.1109/HONET.2019.8908098.

[6] Borui Cui, Cheng Fan, Jeffrey Munk, Ning Mao, Fu Xiao, Jin Dong, Teja Kuruganti, A hybrid building thermal modeling approach for predicting temperatures in typical, detached, two-story houses, Applied Energy, Volume 236,2019,Pages 101-116,ISSN 0306-2619, https://doi.org/10.1016/j.apenergy.2018.11.077.

[7] https://power.larc.nasa.gov/data-access-viewer/
[8] Steadman, R. G. (July 1979)."The Assessment of Sultriness. Part I: A Temperature-Humidity Index Based on Human Physiology and Clothing Science". Journal of Applied Meteorology.18(7): 861–873. Bibcode:1979JApMe..18..861S.doi:10.1175/1520-0450(1979)018<0861:TAOSPI>2.0.CO;2.
[9] Lans P. Rothfusz. "The Heat Index 'Equation' (or, More Than You Ever Wanted to Know About Heat Index)", Scientific Services Division (NWS Southern Region Headquarters), 1 July 1990
[10] https://en.wikipedia.org/wiki/Heat_index
[11] https://web.archive.org/web/20110629041320/http://www.crh.noaa.gov/pub/heat.php
[12] https://en.wikipedia.org/wiki/Room_temperature
[13] Bharati, Subrato & Mondal, M. Rubaiyat & Podder, Prajoy & Prasath, Surya. (2022). Federated learning: Applications, challenges and future directions. International Journal of Hybrid Intelligent Systems. 18. 19-35. 10.3233/HIS-220006.
[14] Zhang, C., Xie, Y., Bai, H., Yu, B., Li, W., Gao, Y. (2021). A survey on federated learning. Knowledge-Based Systems, Volume 216, 106775. ISSN 0950-7051. https://doi.org/10.1016/j.knosys.2021.106775.
[15] Daniel J. Beutel, Taner Topal, Akhil Mathur, Xinchi Qiu, Javier Fernandez-Marques, et al.. FLOWER: A FRIENDLY FEDERATED LEARNING FRAMEWORK. 2022. hal-0