[1]Qasim Mustafa Zainel,

[1,2] Parviz Rashidi-Khazaee,

[2,1] Leila Sharifi

# Leveraging Large Language Models and GAN-Augmented Data for Energy Load Prediction

Journal of Electrical Systems

**Abstract:**Large Language Models (LLMs) originally developed for tasks like text generation and translation have shown successful potential in capturing temporal and complex dependencies, making them suitable for different tasks. In this study, we want to propose a LLMs-based Heating Load (HL) and Cooling Load (CL) estimation model based on residential building characteristics. At first, a prompt generation module was proposed to convert in-hand tabular data to useful prompts, and then the hugging face pre-trained Bart-base model was re-trained to create a new prediction tool for residential buildings HL and CL prediction. In addition, to improve the performance of the proposed LLM-based model, a new data augmentation module was proposed based on Generative Adversarial Network (GAN) and Conditional GAN to increase the size of training data. The proposed model combines Data Augmentation and Prompt generation Modules with LLM and is named DAPM-LLM. The prediction result showed that the DAPM-LLM can predict energy usage using linguistic prompts, and the data augmentation module improved model performance by 600% and 300% in HL and CL prediction, respectively. The comparison of its results with other works shows its superiority over most of them except ensemble models. Using larger pre-trained models and sufficient data will enable these models to outperform ensemble models too. The results showed that the DAPM-LLM model can be successfully used in solving complex problems such as energy consumption prediction, and can be used by engineers and designers to select the best design/plan for building construction by using linguistic sentences.

**Keywords**: Residential building Energy usage prediction, Prompt engineering, Generative adversarial network, Tabular data augmentation tool, Bart

## INTRODUCTION

Reducing energy usage in residential building is a major concern of many engineers and designers from the entire world that consume 39% of worldwide energy production [1]. They looking for efficient tools to help them in designing energy efficient buildings. Available tools use physical models or data-driven models to help them with energy usage estimation [2]. With surprising progress in Machine Learning (ML) and artificial intelligence (AI), many powerful tool was developed to estimate and predict energy usage based on design characteristics [3]. If we can develop a tool that can predict the amount of energy consumption based on building structural characteristics, then we can help engineers and designers to choose the best design from the point of view of energy consumption to choose the best design among unique designs.

Accurately predicting amount of required heating and cooling loads in residential buildings is crucial for achieving energy efficiency, reducing environmental impact, and improving occupant comfort. With the rise of smart buildings and cities, the ability to forecast energy demands has become essential in optimizing heating, ventilation, and air conditioning (HVAC) systems. Traditional approaches to energy load prediction have relied on a combination of statistical and machine-learning methods [3], which have produced promising results [4] but are often limited by their reliance on feature engineering and domain-

[1] [1] Computer Engineering Department, Urmia University, Urmia, Iran

[2] Information Technology and Computer Engineering Department, Urmia University of Technology, Urmia, Iran

**Corresponding Author:** p.rashidi@uut.ac.ir

specific knowledge.

The state-of-the-art ensemble models in Heating load (HL) and Cooling Load (CL) prediction [4] in residential buildings, combining multiple ML models to address the nonlinearities and complexities of energy data [5, 6]. However, ensemble and other published ML models that used in-hand dataset [5], such as Gaussian processes (GP) [6], eXtreme Gradient Boosting (XGB) [7, 8], Grid Search-tuned XGB (GS-XGB) [4], Adaptive Boosting (ADA) and XGB, optimized by the Covariance Matrix Adaptation Evolution Strategy (CMAES) method (XACM) [9], meta-heuristic ensemble model [10], support vector regressor (SVR) + artificial neural network (ANN) [11], Bayesian-XGB [12], Random Forest (RF) [13, 14], Evolutionary Multivariate Adaptive Regression Splines (EMARS) [15], Multilayer Perceptron network tuned with Particle Swarm Optimization (MLP-PSO) [16], genetic programming approach (GPA) [17], Evolutionary Neural Machine Inference Model (ENMIM) [10], SVR [18], Regression Tree Ensemble (SRTE) [19], MLP tuned with PSO and Grey Wolf Optimizer (PSOGWO-MLP) [20] can become computationally expensive and may struggle to adapt to dynamic and unseen data patterns, especially when dealing with large, real-time datasets. In addition, engineers must have the proper technical knowledge to use them.

ChatGPT, AI-based chatbot created by open AI, is powered by LLM. As a result, it can be said that GPT Chat is able to understand human-like answers. Therefore, it can be said that the most important feature of GPT chat is that it can have a conversation with you just like when you are talking to a very knowledgeable person. This chatbot can talk to you about various topics, from history to philosophy and culture. In addition, it can help you in many other areas, such as passing professional exams, composing poetry, and writing code, among other abilities [21].

Recent developments in natural language processing (NLP) and deep learning have led to innovative approaches for managing sequential data, including energy load time series [22]. Promptcast, a framework that uses LLMs for energy time sreies prediction [23], LLM-based automatic building modeling platform in EnergyPlus [24], autoregressive time series predictor based on LLM for predicting a future value of time series [25], well-pre-trained LLMs, such as Claude 3, GPT-4, and Llama2 for addressing both linear and non-linear regression tasks [26] are successful application of using LLMs in solving real world complicated tasks.

Using LLM models in solving complex real world engineering problems is still in its infancy and requires a lot of efforts. The performance improvement of pre-trained LLM models depends highly on the size of data for re-training in problem context. The in-hand tabular data contains only 768 samples of residential building data [5]. To increase the size of data, and improving re-training performance, data augmentation techniques can generate new but dependent data [2]. Data augmentation techniques have been successfully applied to image data, leading to significant progress and improvements [27]. But, the in-hand data is tabular and includes numerical and categorical data. Therefore, it is necessary to consider tabular data augmentation techniques [28, 29]. The Table-GAN generates synthetic data using the vanilla GAN approach (VGAN) [30]. However, the inability of Table-GAN to regulate synthetic data creation may exacerbate imbalances in categorical features. The Conditional Tabular GAN (CTGAN) [31] and Synthetic Minority Over-sampling Technique (SMOTE) [32] have been introduced to solve these issues. The GANBLR changed vanilla GAN architectures using a Bayesian network for both the generator and discriminator [33]. The Tabular Variational Autoencoder (TVAE), a modified version of the VAE for tabular data, significantly improved classification task performance [31]. The TimeGAN addresses the data scarcity problem and enhances the accuracy of heating load prediction models [34]. Transfer learning-based data fusion is more efficient than direct data fusion and enhanced data augmentation strategies for optimal results [35]. Conditional Variational Autoencoders (CVAE) generated synthetic but potentially valuable data for constructing an energy forecast model for the next 24 hours [36, 37]. In this study, based on initial evaluation of different methods, a new data augmentation framework has been proposed based on GAN and CGAN combination to increase the size of the dataset, and use it for LLM re-training, aiming to have a more accurate prediction model.

These LLM methods bypass the need for extensive feature engineering, and can directly process the data, using linguistic descriptions, making it an attractive solution for predicting energy loads. This paper proposes a novel approach for predicting heating and cooling loads in residential buildings using LLMs for the first time, based on our knowledge. By leveraging the pre-trained knowledge and modeling capabilities of LLMs, we aim to develop and proposed a new LLM-based tool for HL and CL prediction to improve prediction accuracy and reduce the dependence on domain-specific feature engineering. This approach builds on existing research in energy load forecasting, advanced data augmentation techniques, and explores how state-of-the-art NLP techniques can be adapted for HL and CL prediction based on in-hand tabular data.

The rest of this work is organized: Section 2 discuss related works. Section 3 discusses the methodlogy, the proposed model structure, dataset, background information, and the structure of the new synthetic data generation tools. Section 4 provides model implementation results and its comparison with similar works. Section 5 discusses the results, and Section 6 presents the conclusions.

## RELATED WORK

Recent developments in natural language processing (NLP) and deep learning have led to innovative approaches for managing sequential data, including energy load time series. Large Language Models (LLMs), originally developed for tasks like text generation and translation, have shown potential in capturing temporal dependencies, making them suitable for a complex task like time-series forecasting. Xue and Salim [22] using a novel approach based on existing language models have presented a tool for predicting the energy consumption load. They enable accurate and dynamic prediction of energy consumption through configuration, fine-tuning and re-training of existing language models. Their approach by using the power of LLMs has opened a new horizon in solving complex engineering problems. Their proposed approach and its accuracy and efficiency have been investigated and confirmed using real data. They used Bart, Bigbird, and Pegasus and showed that many times the Pegasus outperforms other models [22]. Also, Xue and Salim introduced Promptcast, a framework that uses LLMs for time-series forecasting by leveraging natural language prompts. They used different LLMs to check the efficeiny of promptcast tools and showed that Bigbird and RoBERTa outperformed other models in energy prediction [23]. The application of LLMs for time-series forecasting has also been explored in other domains, such as human mobility prediction. Xue et al. [38] demonstrated how LLMs can model both spatial and temporal dependencies in mobility data, which suggests their applicability in energy forecasting, where similar dependencies exist.

Jiang et al. for providing the automatic building modeling platform in EnergyPlus software used LLM [24]. Their model changes descriptive information of buildings such as usage scenarios, equipment loads, and different geometries into linguistic descriptions and uses them to reset and train the linguistic model. Through the process of fine-tuning, the LLM, specifically T5, transforms human descriptions into EnergyPlus modeling files. Subsequently, it produces outputs that are appropriate for users by utilizing the API integrated within the Eplus-LLM platform.

Liu et al. employed multi-step generation capability of LLMs and the general-purpose token transfer, and proposed AutoTimes [25]. AutoTimes is proposed to work as autoregressive time series predictor based on LLM ability and predict a future value of time series successfully.

Vacareanu et al. examine the feasibility of addressing both linear and non-linear regression tasks within a specified context using well-pre-trained LLMs, such as Claude 3, GPT-4, and Llama2. Their research indicates that these models are capable of effectively solving regression tasks, comparable to traditional supervised techniques like RF, Bagging, and Gradient Boosting, and sometimes, they even surpass the performance of these methods [26].

## METHODOLOGY

The proposed LLM-Based HL and CL prediction models, named DAPM-LLM, contains the following steps which are graphically shown in Figure 1.

1- After loading the original in-hand data, used a 10-fold cross-validation method to split dataset into training and testing data.

a. Use 9-fold data for model training and one remaining fold for model testing.

b. Repeat this process ten times to test model accuracy and generality against each sample of data

2- Pass training data into the proposed data augmentation module

a. The proposed models' performance will be calculated with and without data augmentation.

3- Use the developed prompt generation module to convert building tabular data into sentences that could be processed by the LLM model.

4- Fine-tune the pre-trained Bart-Base model using original or newly generated prompt data.

5- Test model performance with unseen test fold and calculate performance metrics.

6- Repeat steps 1 to 5, ten times and finally analyze the model results, and compare it with other published works.

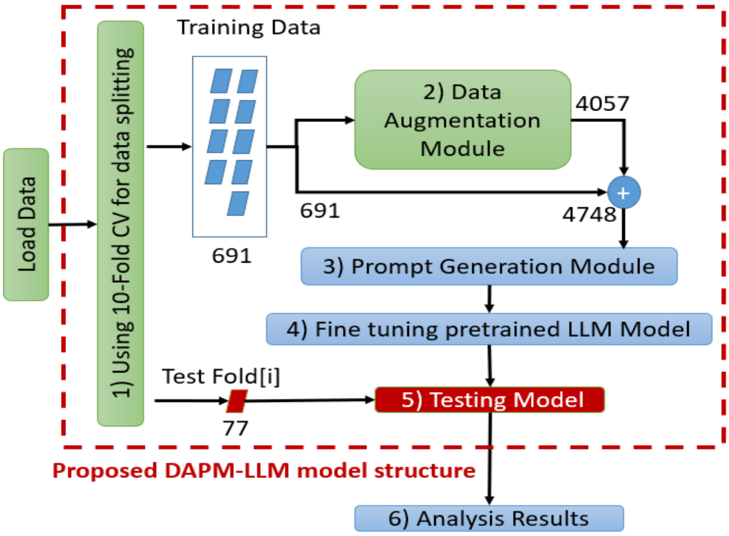
Load Data
1) Using 10-Fold CV for data splitting
Training Data
691
2) Data Augmentation Module
4057
691
4748
3) Prompt Generation Module
4) Fine tuning pretrained LLM Model
Test Fold[i]
77
5) Testing Model
Proposed DAPM-LLM model structure
6) Analysis Results

**Figure 1: The proposed DAPM-LLM model structure**

### Dataset

The selected dataset contains structural charactristics of 768 buildings [5]. Each building has eight features: relative compactness, surface area, wall area, roof area, overall height, orientation, glazing area, and glazing area distribution. It contains the HL and CL consumed in kilowatts (kW) for each building. The materials used in the buildings were identical and thus excluded from the analysis. Researchers have widely used this dataset to predict HL and CL [4, 11, 13]. The main reason of selecting this dataaset is that it containg building design structure which could be used for selecting best desing/plan for building counstruction. Although other datasets have many records, but they cannot be used for the purpose of this research, which is to select the best design based on physical characteristics, because they do not contain structural information about the buildings.

To create a sample of 720 buildings, the researchers considered 12 buildings with four distinct orientations, five glazing area distributions per building, and three glazing area variations for each building, resulting in a sample size of 12 * 4 * 5 * 3 = 720. They also included four glazing-free orientations for each building, thus generating 720 + 12 * 4 = 768 distinct simulated buildings.

### Model Training and Testing

The 10-fold cross-validation (CV) technique was employed within this work for model training and testing at each iteration. In a 10-fold CV, all data is split into ten folds (use 9 folds for training, and the remaining one fold for testing). This process is repeated 10 times, ensuring the model is tested against different unseen folds in each iteration, testing its generality, and the average performance of these ten iterations will be reported as the model performance metrics rather than relying on the best fold testing.

The proposed LLM model performance will be checked with and without data augmentation.

**Proposed Data Augmentation Module**

To improve LLM-Based HL and CL prediction performance and reduce the time and cost of model re-training, in this study a new data augmentation module was proposed to generate new synthetic data.

After an initial evaluation of different data augmentation methods, such as SMOTE [32] and VAE [31], we selected the GAN and CGAN methods to augment the data and generate new synthetic samples. Based on our knowledge, in this study, the GAN and CGAN and their combination are used for the first time to generate new data to investigate the performance of energy consumption prediction models. Many libraries and methods are available for tabular data augmentation [3]. In this work, we used the TabGan library provided by Ashrapov, which offers three different methods—GAN, Conditional GAN, and diffusion—for generating new data [39]. The main reason for selecting GAN and CGAN is their special structure that generates more valuable and relevant new data samples, which was discussed in following subsections.

**Generative Adversarial Networks (GANs)**

The most common application of Generative Adversarial Networks (GANs) is generating synthetic image data. However, they can now also be used to generate synthetic tabular data. As shown in Fig. 2, a GAN comprises two deep networks, a generator and a discriminator, which train simultaneously [40]. The generator network creates data that mimics actual data, aiming to produce outputs that the discriminator cannot distinguish from actual data. If the discriminator detects differences, both networks are updated to help generate more accurate and realistic data.

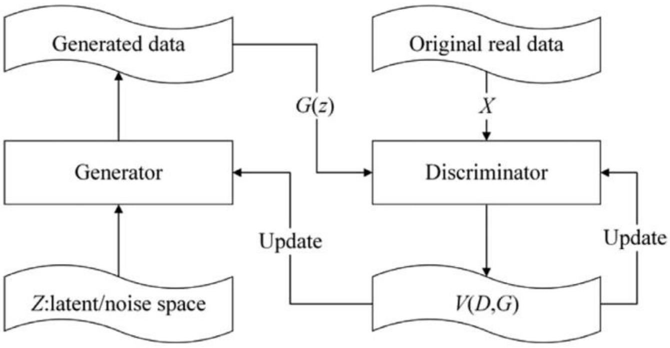

**Figure 2: Architecture of GAN[40]**

The value function V (D, G) of the GAN can be defined: Z is the noise space, G(z) represents a mapping from the noise space to the generated data space, and X is the original data space. The V (D, G) is defined as equation (1) [40]:

$$V(D,G)=E_{x\sim p_{\text{data}}(x)}[\log D(x)]+E_{z\sim p_z(z)}[\log(1-D(G(z)))] \quad (1)$$

Where z is the noise from the noise space Z, $p_z(z)$ is defined as a prior on the input noise variables, G is a differentiable function represented by a multilayer perception, x is the sample from the original space X, $p_{data}(x)$ is the distribution of the original data, and D(x) describes the possibility that x comes from the original data rather than the generator.

This objective is maximized by the discriminator and minimized by the generator through training. In other words, by resolving the following optimization problem, the generator and discriminator are trained:

$$\min_{G}\max_{D}V(D,G)=E_{X\sim p_{\text{data}}(x)}[\log D(x)]+E_{z\sim p_z(z)}[\log(1-D(G(z)))] \quad (2)$$

The discriminator and generator play a single two-step game to min-max V (D, G) to obtain a wellbehaved GAN. The discriminator (D) is optimized while the generator (G) is fixed to maximize discrimination accuracy. Subsequently, the generator G is tuned to minimize discrimination accuracy while the discriminator D is fixed. The procedure is carried out repeatedly. When the generator is known, V (D, G) in continuous space can be explained:

$$\begin{aligned} V(D) &= \int_x p_{\text{data}}(x) \log(D(x))dx + \int_z p_z(z)\log(1 - D(G(z)))dz \\ &= \int_x \left[p_{\text{data}}(x)\log(D(x)) + p_g(x)\log(1 - D(x))\right]dx \end{aligned} \tag{3}$$

Where the generative distribution, $p_g(x)$, is picked up from the initial data set, x.

The discriminator's optimal value, $D_G^*(x)$, is found in equation (4) when the generator is fixed:

$$D_G^*(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_g(x)} \tag{4}$$

By utilizing the original data as input, the discriminator estimates the conditional probability of the input data by maximizing the log-likelihood. Therefore, the min-max game in equation (2) is restructured as:

$$\begin{aligned} C(G) = \max_D V(D,G) = E_{x\sim p_{\text{data}}(x)}\left[\log \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_g(x)}\right] \\ + E_{x\sim p_g}\left[\log \frac{p_g(x)}{p_{\text{data}}(x) + p_g(x)}\right] \end{aligned} \tag{5}$$

The global optimal solution of V(D, G) and the minimum value of the virtual training criterion C(G) will be reached if and only if $p_{\text{data}} = p_g$.

### Conditional Tabular GAN (CGAN)

A Conditional Tabular GAN (CTGAN) is a GAN-based technique that uses sample rows from a tabular data distribution to model its distribution. Xu et al. developed mode-specific normalization to address the multimodal and non-Gaussian distribution challenges in CTGAN creation [31]. They introduced a conditional generator to manage unbalanced discrete columns and trained a high-quality model using multiple state-of-the-art techniques and fully connected networks.

In a Conditional GAN (CGAN), the discriminator and generator are conditioned on additional information y. This auxiliary data, such as class labels or information from different modalities, is used to condition the GAN [41]. The conditioning is implemented by incorporating y as an extra input layer to the discriminator and generator. The generator uses a joint hidden representation composed of the prior input noise $p_z(z)$ and y. The adversarial training framework gives the generator significant flexibility in creating this hidden representation.

By using the proposed data augmentation module, new 4057 synthetic sample data was generated and are combined with 691 samples in the training data, creating a large dataset containing 4748 samples. This new dataset will be utilized for tuning, training, and evaluating the proposed LLM-Based model. Within the proposed structure, the new synthetic dataset will be fed into the Prompt engineering module to generate sentences.

### Proposed Prompt Generation Module

In this study, to convert tabular data to linguistic sentences, a prompt module was developed. This module receives tabular data sample and generates a sentence that will be used for model retraining. Table 1 shows how it converted tabular data to prompt sentences. Then, this sentence will feed into tokenizer part to LLM model to be used for model re-training.

**Table 1: Sample of tabular data converted to prompt**

| Original Data | X1 | X2 | X3 | X4 | X5 | X6 | X7 | X8 | Y1 | Y2 |
|---|---|---|---|---|---|---|---|---|---|---|
| | 0.71 | 710.5 | 269.5 | 220.5 | 3.5 | 4 | 0 | 0 | 6.37 | 11.29 |
| HL Prompt Sentence | The Building Relative compactness is 0.71, Surface area is 710.5, Wall area is 269.5, Roof area is 220.5, Overall height is 3.5, Orientation is 4.0, Glazing area is 0.0, and Glazing area distribution is 0.0. What will be the heating load? | | | | | | | | | The heating load will be 6.37 |
| CL Prompt Sentence | The Building Relative compactness is 0.71, Surface area is 710.5, Wall area is 269.5, Roof area is 220.5, Overall height is 3.5, Orientation is 4.0, Glazing area is 0.0, and Glazing area distribution is 0.0. What will be the cooling load? | | | | | | | | | The cooling load will be 11.29 |

### Fine-tuning pre-trained LLM Model for HL and CL Prediction

Despite successful usage of different LLMs in time-series and energy load prediction and superiority of Pegasus [22], Bigbird and RoBERTa [23] over other models , based on in-hand hardware and GPU resources (free version of Google CoLab and GTX 1080 TI GPU with 11G memory), we selected Facebook Bart-Base model for HL and CL prediction task, and analyzed its performance with proposed data augmentation module in different cases.

BART is a sequence-to-sequence model that employs a transformer architecture, featuring a bidirectional encoder akin to BERT and an autoregressive decoder similar to GPT, developed by Hugging Face. The pre-training of BART involves two key processes: (1) introducing noise to the text through a random noising function, and (2) training a model to restore the original text. This model demonstrates notable effectiveness when fine-tuned for text generation tasks, such as summarization and translation, while also performing admirably in comprehension tasks, including text classification and question answering [42].

### Testing Model Performance Metrics

Several statistical criteria were calculated to evaluate the performance and accuracy of prediction models. These metrics assess how closely the predicted values align with the actual values. Mean Absolute Error (MAE), Mean Square Error (MSE), and Root Mean Square Error (RMSE) are widely recognized performance evaluation metrics for continuous target values [4]. Smaller values of these metrics indicate that the model predicts the target values with low error and high accuracy. These metrics will be calculated based on the equation (6) to (8):

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|o_i - y_i| \tag{6}$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(o_i - y_i)^2 \tag{7}$$

$$RMSE = \sqrt{MSE} \tag{8}$$

Where N is the number of samples in the testing dataset, $o_i$ is the predicted (estimated) value of HL or CL by the proposed model, $y_i$ is the actual value of HL or CL, and $\overline{y_i}$ is the mean value of the actual value of HL or CL.

To have a model fair comparison, the model overall performance is average of 10 different runs based on 10-fold CV.

## RESULTS

Different parameters must be defined in training and retraining LLM models. Based on in-hand resources and different tries, we selected epoch = 50 to re-train the proposed model on the original data and generated data using the proposed data augmentation module.

### HL Prediction Results

Table 2 shows the performance of the proposed HL prediction model based on LLM on the original and newly generated data based on 10-fold cross-validation.

**Table 2: HL prediction performance with and without data augmentation module**

| | MAE | | MSE | | RMSE | |
|---|---|---|---|---|---|---|
| **Fold** | **Original Data** | **New Data** | **Original Data** | **New Data** | **Original Data** | **New Data** |
| **1** | 2.13 | 0.40 | 7.35 | 0.35 | 2.71 | 0.60 |
| **2** | 2.82 | 0.39 | 14.94 | 0.33 | 3.87 | 0.58 |
| **3** | 1.30 | 0.37 | 3.40 | 0.32 | 1.85 | 0.57 |
| **4** | 2.02 | 0.40 | 7.25 | 0.34 | 2.69 | 0.58 |
| **5** | 2.81 | 0.36 | 12.85 | 0.35 | 3.58 | 0.59 |
| **6** | 1.92 | 0.35 | 6.85 | 0.29 | 2.62 | 0.54 |
| **7** | 2.72 | 0.39 | 12.90 | 0.34 | 3.59 | 0.58 |
| **8** | 1.52 | 0.33 | 4.59 | 0.25 | 2.14 | 0.50 |
| **9** | **2.70** | **0.28** | **12.55** | **0.20** | **3.54** | **0.45** |
| **10** | 1.71 | 0.32 | 5.31 | 0.30 | 2.30 | 0.54 |
| **Avg** | **2.16** | **0.36** | **8.80** | **0.31** | **2.89** | **0.55** |

The MAE varies between 1.30 and 2.82 when re-training the model on the original data, while it varies between 0.28 and 0.40 when using the data augmentation module. The presented results indicated that the proposed data augmentation module helped model performance improvement and its stability on all testing folds. Also, by using data augmentation module, the model average performance is 0.36 which improved overall performance by 600%.

Increasing epoch size or training iteration rounds will affect model performance. Table 3 shows HL prediction performance with different epochs, varying from 30 to 300. The evaluation was conducted when fold 10 was considered as the testing fold, and the other nine folds were used for model training and data generation. The presented result shows increasing epoch size helped model performance improvement.

**Table 3: HL prediction performance without data augmentation module with different epoch size**

| Epoch | MAE | MSE | RMSE |
|---|---|---|---|
| **30** | 2.23 | 10.79 | 3.28 |
| **50** | 1.71 | 5.31 | 2.30 |
| **80** | 0.51 | 0.72 | 0.85 |
| **100** | 0.43 | 0.77 | 0.88 |
| **120** | 0.37 | 0.33 | 0.58 |
| **150** | 0.36 | 0.31 | 0.56 |
| **180** | 0.32 | 0.26 | 0.51 |
| **200** | 0.27 | 0.23 | 0.48 |
| **250** | 0.28 | 0.22 | 0.47 |
| **300** | 0.30 | 0.24 | 0.49 |

To check the effect of epoch size on the DAPM-LLM prediction performance, Table 4 shows that there are no effects on its performance. In comparison with the presented result in Table 3, the new DAPM-LLM performance with epoch size equals 30 is similar to epoch size equals to 180. Therefore, the proposed data generation module helps model re-training, improves its accuracy, and reduces the time and cost of re-retraining. Figure 3 compares two cases' MAE performance with same epochs. Increasing epoch size causes to re-training model more rounds, and helped it to reach the performance level similar to the DAPM-LLM. The presented result indicates that the proposed data augmentation module helped performance improvement successfully.

**Table 4: Effect of epoch size on proposed HL prediction model performance**

| Epoch | MAE | MSE | RMSE |
|---|---|---|---|
| **30** | 0.32 | 0.29 | 0.54 |
| **50** | 0.32 | 0.30 | 0.54 |
| **80** | 0.30 | 0.25 | 0.50 |
| **100** | 0.41 | 0.70 | 0.84 |
| **120** | 0.32 | 0.28 | 0.53 |
| **150** | 0.33 | 0.27 | 0.52 |
| **180** | 0.31 | 0.26 | 0.51 |
| **200** | 0.32 | 0.29 | 0.54 |

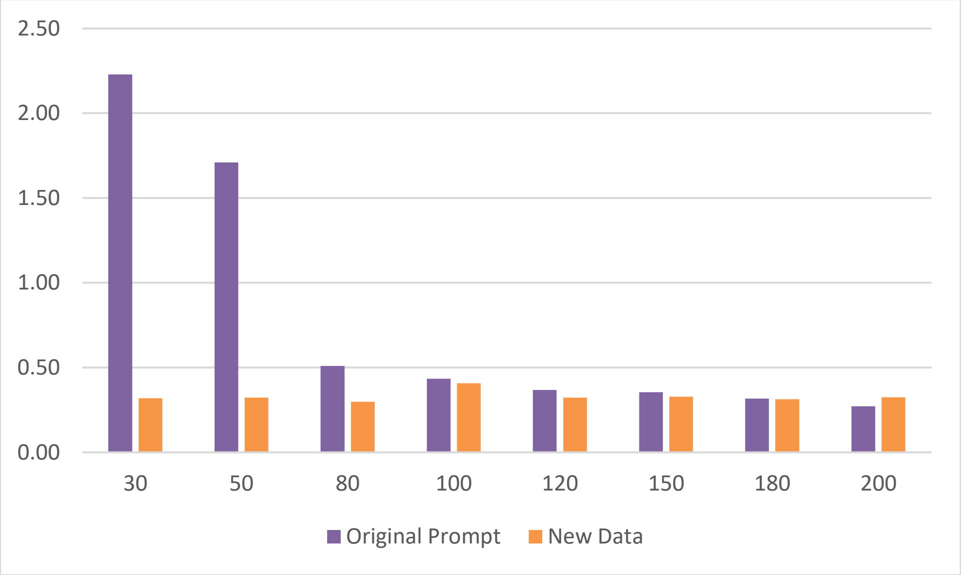

**Figure 3: Comparison of HL model performance on different epochs**

### CL Prediction Results

Table 5 shows the performance of the proposed CL prediction model based on LLM on the original and newly generated data based on 10-fold cross-validation.

**Table 5: Effect of epoch size on proposed CL prediction model performance**

| | MAE | | MSE | | RMSE | |
|---|---|---|---|---|---|---|
| **Fold** | **Original Data** | **New Data** | **Original Data** | **New Data** | **Original Data** | **New Data** |
| 1 | 2.46 | 0.63 | 9.72 | 1.02 | 3.12 | 1.01 |
| 2 | 2.49 | 0.58 | 11.10 | 1.34 | 3.33 | 1.16 |
| 3 | 2.37 | 0.90 | 9.94 | 2.95 | 3.15 | 1.72 |
| 4 | 2.38 | 0.87 | 12.00 | 3.04 | 3.46 | 1.74 |
| 5 | 2.06 | 0.74 | 7.67 | 2.83 | 2.77 | 1.68 |
| 6 | 1.93 | 0.70 | 6.93 | 2.39 | 2.63 | 1.54 |
| 7 | 2.43 | 0.86 | 11.63 | 2.25 | 3.41 | 1.50 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 8 | 2.09 | 1.14 | 8.02 | 4.43 | 2.83 | 2.10 |
| 9 | 2.46 | 0.54 | 10.12 | 0.54 | 3.18 | 0.73 |
| 10 | **2.24** | **0.50** | **9.20** | **0.85** | **3.03** | **0.92** |
| **Avg** | **2.29** | **0.75** | **9.63** | **2.16** | **3.09** | **1.41** |

As shown in table 5, the MAE varies between 1.93 and 2.49 when re-training the model on the original data, while it varies between 0.50 and 0.90 when using the data augmentation module. The presented results indicated that the proposed data augmentation module helped model performance and stability on all testing folds. Also, the DAPM-LLM models’ average performance is 0.75 which improved the models’ overall performance by 300%, when trained without data augmentation.

Increasing epoch size or training iteration rounds will affect model performance. Table 6 shows CL prediction performance with different epoch size, varying from 10 to 300. The evaluation was conducted when fold 2 was considered the testing fold, and the other nine folds were used for model training and data generation. The presented result shows increasing epoch size helped model performance improvement.

**Table 6: CL prediction performance without data augmentation module with different epoch size**

| **Epoch** | **MAE** | **MSE** | **RMSE** |
|---|---|---|---|
| 10 | 10.88 | 202.39 | 14.23 |
| 20 | 3.36 | 20.41 | 4.52 |
| 30 | 2.99 | 15.63 | 3.95 |
| 50 | 2.49 | 11.10 | 3.33 |
| 80 | 1.19 | 3.70 | 1.92 |
| 120 | 1.14 | 3.91 | 1.98 |
| 150 | 1.12 | 4.30 | 2.07 |
| 180 | 0.61 | 1.11 | 1.06 |
| 200 | 0.99 | 3.70 | 1.92 |
| 250 | 0.60 | 1.09 | 1.04 |
| 300 | 0.75 | 2.15 | 1.47 |

To check the effect of epoch size on the DAPM-LLM prediction performance, Table 7 shows that there are no effects on its performance. In comparison with the presented result in Table 6, the new model performance with epoch size equal 50 is similar to its performance when epoch size is 250. Therefore, the proposed data generation module helps model re-training, improves its accuracy, and reduces the time and cost of re-retraining. Figure 4 compares two cases' MAE performance with similar epochs. By increasing the epoch size, training the model in more rounds, helps the model re-trained with original data reached the performance of the DAPM-LLM model.

**Table 7: Effect of epoch size on proposed CL prediction model performance**

| **Epoch** | **MAE** | **MSE** | **RMSE** |
|---|---|---|---|
| 10 | 1.41 | 4.99 | 2.23 |
| 20 | 1.00 | 2.88 | 1.70 |
| 30 | 0.83 | 3.04 | 1.74 |
| 50 | 0.58 | 1.34 | 1.16 |
| 80 | 0.56 | 0.91 | 0.95 |
| 120 | 1.05 | 4.82 | 2.20 |
| 150 | 0.81 | 3.01 | 1.73 |
| 180 | 0.62 | 2.10 | 1.45 |
| 200 | 0.62 | 1.24 | 1.12 |

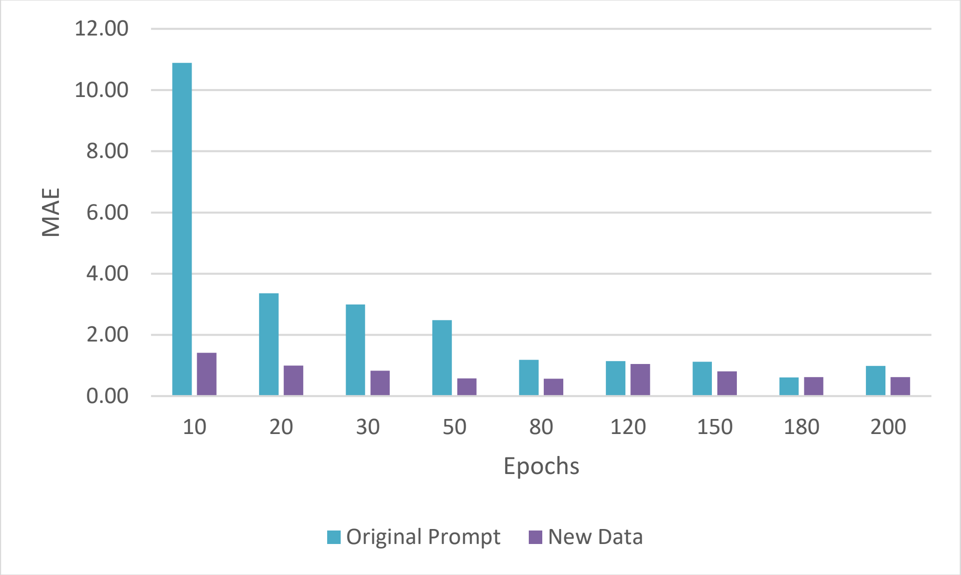

**Figure 4: Comparison of CL model performance on different epochs**

Because of the low in-hand resources in hand, it was not possible to check the performance of other large LLM models. We only check the Google Pegasus-Large LLM model [22] on the original CL data with epoch sizes of 10, 20, and 30. The Pegasus-Large is a state-of-the-art model designed for text summarization. The results presented in Table 8, indicates that the Pegasus-Large model outperforms Bart in CL prediction and shows its superiority. It was not possible to run Pegasus-Large with new data and greater epoch size, because of low available resources.

**Table 8: Comparison of Bart and Pegasus in CL prediction using original data**

| | **Pegasus-Large** | | | **Bart-Base** | | |
|---|---|---|---|---|---|---|
| **Epoch** | **MAE** | **MSE** | **RMSE** | **MAE** | **MSE** | **RMSE** |
| 10 | 3.08 | 17.22 | 4.15 | 10.88 | 202.39 | 14.23 |
| 20 | 1.64 | 5.94 | 2.44 | 3.36 | 20.41 | 4.52 |
| 30 | 1.41 | 5.22 | 2.29 | 2.99 | 15.63 | 3.95 |

**Discussion**

Table 9 compares the proposed DAPM-LLM, HL and CL performance prediction with other published work that used in-hand dataset.

**Table 9: Comparison of different HL prediction models' performance**

| | **HL Prediction** | | **CL Prediction** | |
|---|---|---|---|---|
| **Best Model** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| RF (2012)[5] | 0.51 | | 1.42 | - |
| EMARS (2014)[15] | 0.34 | 0.46 | 0.68 | 0.97 |
| SVR+ANN (2014)[43] | **0.236** | 0.35 | 0.89 | 1.57 |
| GPA (2015)[17] | 0.38 | - | 0.97 | - |
| RF (2017)[11] | 0.351 | 0.22 | **0.565** | 0.84 |
| GP (2018)[6] | **0.251** | 0.38 | **0.448** | 0.67 |
| GS-XGB (2019)[4] | **0.175** | 0.265 | **0.307** | 0.461 |
| RF (2019)[44] | 0.557 | 1.589 | | |
| MLP-PSO (2020)[16] | 1.863 | 2.569 | 2.136 | 3.122 |
| ENMIM (2020)[10] | 0.71 | 0.98 | **0.35** | 0.47 |
| MLP (2020)[18] | 0.412 | 0.483 | 1.476 | 1.739 |

| | | | | |
|---|---|---|---|---|
| SRTE (2022)[19] | 0.332 | 0.452 | **0.536** | 0.690 |
| PSOGWO-MLP (2023)[20] | 0.787 | 1.412 | 1.470 | 1.927 |
| Bayesian-XGB (2023)[12] | **0.247** | 0.380 | **0.454** | 0.757 |
| XGB (2024)[8] | 0.356 | 0.492 | **~ 0.64** | 0.922 |
| XACM (2024)[9] | ~0.65 | 0.904 | ~ 0.94 | 1.247 |
| **Proposed DAPM-LLM** | **0.32** | 0.54 | 0.64 | 0.62 |

The results showed that our proposed model outperformed most published works in HL prediction except GS-XGB (2019)[4], SVR+ANN (2014)[43], GP (2018)[6] and Bayesian-XGB (2023)[12]. Also, in CL prediction, our proposed model outperformed most published model except GS-XGB (2019)[4], GP (2018)[6], XGB (2024)[8], RF (2017)[11], ENMIM (2020)[10], Bayesian-XGB (2023)[12] and SRTE (2022)[19]. Most of outperforming models used complicated ensemble or hyper-parameter tuned methods to predict HL and CL that need more efforts and complexity.

Compared to the published works, the presented model has an acceptable performance, although it is still not as efficient as the advanced hybrid models, but it should be noted that in this study, due to the limited resources, the simple LLM, Facebook Bart-Base model was used. Using larger language models will certainly improve prediction performance. The initial comparison of the Pegasus-Large model with Bart-Base on the original CL data, Table 8, confirms this hypothesis. In addition, LLMs are still in their infancy and will achieve significant improvements in the coming years and are expected to perform better than ML models, such as hybrid ensemble models, in solving many real-world problems.

In this study, the in-hand data set contains only 768 records, that only 691 record was used for re-training and 77 record was used for model evaluation. The prediction performance indicated when epoch size is small, like 10 and 20, the model performance is feeble. To increase its performance, we increased the epoch size from 10 to 300 and check model performance. Increasing epoch size helps model performance improvement but needs more time to tune and re-train the model.

In comparison, the proposed data augmentation module has increased the number of training records by almost 7 times and has helped the model to have a suitable and stable performance even with a handful of epochs, which shows that there is no significant change in the prediction results with the increase in the number of epochs. This behavior indicates that the generated data contains most of the hidden information in the original data and, by reducing the learning time, it helps to improve the performance of the model. Also, this feature is suitable for successive re-training and updating of the model and enables the development team to create a new model and replace the previous model in a quick time in solving real-time applications.

The presented results show that the presented DAPM- LLM model makes it possible to answer the queries about amount of building energy consumption based on the design characteristics in linguistic description. For example, one engineer will interact with an AI system using a linguistic sentence:

Person: “Hi, I have a building design/plan. Can you help me estimate its heating load?”

The AI replies: “yes off course. Can you give detail information about … of your building design?”

Person: “The Building Relative compactness is 0.71, Surface area is 710.5, Wall area is 269.5, Roof area is 220.5, Overall height is 3.5, Orientation is 4.0, Glazing area is 0.0, and Glazing area distribution is 0.0. What will be its heating load?”

The AI replies: “The heating load will be 6.37 kW/h”.

This kind of interaction in linguistic sentences with the AI system is fascinating and surprising and will be possible soon easily.

The major novelty in this study is using the LLM model for the first time to predict amount of required HL and CL based on building characteristics and bring the opportunity to solve such complex problem by using linguistic sentences. In addition, to reduce the time and cost of re-training and fine-tuning the proposed model, a new data augmentation module based on GAN and CGAN was successfully proposed to increase dataset samples by generating new synthetic samples. This module helped the model performance improvement.

The major limitation of this study is low in-hand resources (free version of Google Colab and GTX TI 1080 GPU card with 11G Memory), which prevented to use and test large LLM models in CL and HL prediction.

In future works, we will try to check the performance of large LLMs in HL and CL prediction based on in-hand data and use the LLM models to directly execute regression tasks. Also, adding more layers before and after the LLM models may be helpful. Also, future studies may concentrate on investigating prompt optimization to enhance the precision and relevance of language models in high-level and commercial forecasting.

## CONCLUSION

In this study, by leveraging advancements in LMMs, a model has been proposed to predict the cooling and heating energy requirements of residential buildings in linguistic way. This model enables engineers and designers to present their requests to the model in linguistic language, and the model responds in an appropriately linguistic manner.

The result indicated that LLM-based model is able to solve such complicated tasks and surprisingly outperformed most of published ML models. Also, to improve HL and CL prediction performance, based on generative adversarial network and conditional GAN, a new hybrid data augmentation module was proposed to generate new synthetic training data, which increased data size almost 7 times. The prediction result showed that the proposed hybrid data augmentation module helped model performance improvement and indicated that having more data helped the model to perform better. Proposed data augmentation module improved HL and CL prediction performance by 600% and 300%, respectively.

The presented result indicated that it will be possible soon to communicate with AI systems in linguistic sentences and solve many complex engineering tasks. The AI systems will reply in human understanding sentences.

In the future works, we are going to use a large dataset for LLM-based models re-training and tuning and using large LLMs, which outperform base model most times, in developing more stable and accurate HL and CL prediction model.

## REFERENCES

[1] N. Somu, G. R. MR, and K. Ramamritham, "A hybrid model for building energy consumption forecasting using long short term memory networks," *Applied Energy,* vol. 261, p. 114131, 2020.

[2] H. Fang, H. Tan, R. Kosonen, X. Yuan, K. Jiang, and R. Ding, "Study of the Data Augmentation Approach for Building Energy Prediction beyond Historical Scenarios," *Buildings,* vol. 13, no. 2, p. 326, 2023.

[3] Y. Sun, F. Haghighat, and B. C. Fung, "A review of the-state-of-the-art in data-driven approaches for building energy prediction," *Energy and Buildings,* vol. 221, p. 110022, 2020.

[4] M. Al-Rakhami, A. Gumaei, A. Alsanad, A. Alamri, and M. M. Hassan, "An ensemble learning approach for accurate energy load prediction in residential buildings," *IEEE Access,* vol. 7, pp. 48328-48338, 2019.

[5] A. Tsanas and A. Xifara, "Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools," *Energy and buildings,* vol. 49, pp. 560-567, 2012.

[6] L. Goliatt, P. Capriles, and G. R. Duarte, "Modeling heating and cooling loads in buildings using Gaussian processes," in *2018 IEEE Congress on Evolutionary Computation (CEC)*, 2018: IEEE, pp. 1-6.

[7] C. Lu, S. Li, S. R. Penaka, and T. Olofsson, "Automated machine learning-based framework of heating and cooling load prediction for quick residential building design," *Energy,* vol. 274, p. 127334, 2023.

[8] O. A. Alawi, H. M. Kamar, and Z. M. Yaseen, "Optimizing building energy performance predictions: A comparative study of artificial intelligence models," *Journal of Building Engineering,* vol. 88, p. 109247, 2024.

[9] B. Sadaghat, S. Afzal, and A. J. Khiavi, "Residential building energy consumption estimation: A novel

ensemble and hybrid machine learning approach," *Expert Systems with Applications,* vol. 251, p. 123934, 2024.

[10] D.-H. Tran, D.-L. Luong, and J.-S. Chou, "Nature-inspired metaheuristic ensemble model for forecasting energy consumption in residential buildings," *Energy,* vol. 191, p. 116552, 2020.

[11] G. R. Duarte, L. G. da Fonseca, P. Goliatt, and A. C. de Castro Lemonge, "Comparison of machine learning techniques for predicting energy loads in buildings," *Ambiente Construído,* vol. 17, no. 3, pp. 103-115, 2017.

[12] B. A. Salami, S. I. Abba, A. A. Adewumi, U. A. Dodo, G. K. Otukogbe, and L. O. Oyedele, "Building energy loads prediction using bayesian-based metaheuristic optimized-explainable tree-based model," *Case Studies in Construction Materials,* vol. 19, p. e02676, 2023.

[13] F. Abdel-Jaber and K. N. Dirks, "A Review of Cooling and Heating Loads Predictions of Residential Buildings Using Data-Driven Techniques," *Buildings,* vol. 14, no. 3, p. 752, 2024.

[14] W. Gao, J. Alsarraf, H. Moayedi, A. Shahsavar, and H. Nguyen, "Comprehensive preference learning and feature validity for designing energy-efficient residential buildings using machine learning paradigms," *Applied Soft Computing,* vol. 84, p. 105748, 2019.

[15] M.-Y. Cheng and M.-T. Cao, "Accurately predicting building energy performance using evolutionary multivariate adaptive regression splines," *Applied Soft Computing,* vol. 22, pp. 178-188, 2014.

[16] G. Zhou, H. Moayedi, M. Bahiraei, and Z. Lyu, "Employing artificial bee colony and particle swarm techniques for optimizing a neural network in prediction of heating and cooling loads of residential buildings," *Journal of Cleaner Production,* vol. 254, p. 120082, 2020.

[17] M. Castelli, L. Trujillo, L. Vanneschi, and A. Popovič, "Prediction of energy performance of residential buildings: A genetic programming approach," *Energy and Buildings,* vol. 102, pp. 67-74, 2015.

[18] A. Moradzadeh, A. Mansour-Saatloo, B. Mohammadi-Ivatloo, and A. Anvari-Moghaddam, "Performance evaluation of two machine learning techniques in heating and cooling loads forecasting of residential buildings," *Applied Sciences,* vol. 10, no. 11, p. 3829, 2020.

[19] N. Pachauri and C. W. Ahn, "Regression tree ensemble learning-based prediction of the heating and cooling loads of residential buildings," in *Building Simulation,* 2022, vol. 15, no. 11: Springer, pp. 2003-2017.

[20] S. Afzal, B. M. Ziapour, A. Shokri, H. Shakibi, and B. Sobhani, "Building energy consumption prediction using multilayer perceptron neural network-assisted models; comparison of different optimization algorithms," *Energy,* vol. 282, p. 128446, 2023.

[21] S. Makridakis, F. Petropoulos, and Y. Kang, "Large language models: Their success and impact," *Forecasting,* vol. 5, no. 3, pp. 536-549, 2023.

[22] H. Xue and F. D. Salim, "Utilizing language models for energy load forecasting," in *Proceedings of the 10th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, 2023, pp. 224-227.

[23] H. Xue and F. D. Salim, "Promptcast: A new prompt-based learning paradigm for time series forecasting," *IEEE Transactions on Knowledge and Data Engineering,* 2023.

[24] G. Jiang, Z. Ma, L. Zhang, and J. Chen, "EPlus-LLM: A large language model-based computing platform for automated building energy modeling," *Applied Energy,* vol. 367, p. 123431, 2024.

[25] Y. Liu, G. Qin, X. Huang, J. Wang, and M. Long, "Autotimes: Autoregressive time series forecasters via large language models," *arXiv preprint arXiv:2402.02370,* 2024.

[26] R. Vacareanu, V.-A. Negru, V. Suciu, and M. Surdeanu, "From Words to Numbers: Your Large Language Model Is Secretly A Capable Regressor When Given In-Context Examples," *arXiv preprint arXiv:2404.07544,* 2024.

[27] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of big data,* vol. 6, no. 1, pp. 1-48, 2019.

[28] J. Fonseca and F. Bacao, "Tabular and latent space synthetic data generation: a literature review," *Journal of Big Data,* vol. 10, no. 1, p. 115, 2023.

[29] V. Borisov, T. Leemann, K. Seßler, J. Haug, M. Pawelczyk, and G. Kasneci, "Deep neural networks and tabular data: A survey. arXiv 2021," *arXiv preprint arXiv:2110.01889*.

[30] N. Park, M. Mohammadi, K. Gorde, S. Jajodia, H. Park, and Y. Kim, "Data synthesis based on generative

adversarial networks," *arXiv preprint arXiv:1806.03384,* 2018.

[31] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling tabular data using conditional gan," *Advances in neural information processing systems,* vol. 32, 2019.

[32] A. Fernández, S. Garcia, F. Herrera, and N. V. Chawla, "SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary," *Journal of artificial intelligence research,* vol. 61, pp. 863-905, 2018.

[33] Y. Zhang, N. A. Zaidi, J. Zhou, and G. Li, "GANBLR: a tabular data generation model," in *2021 IEEE International Conference on Data Mining (ICDM)*, 2021: IEEE, pp. 181-190.

[34] Y. Zhang, Z. Zhou, J. Liu, and J. Yuan, "Data augmentation for improving heating load prediction of heating substation based on TimeGAN," *Energy,* vol. 260, p. 124919, 2022.

[35] Y. Lu, Z. Tian, Q. Zhang, R. Zhou, and C. Chu, "Data augmentation strategy for short-term heating load prediction model of residential building," *Energy,* vol. 235, p. 121328, 2021.

[36] C. Fan, M. Chen, R. Tang, and J. Wang, "A novel deep generative modeling-based data augmentation strategy for improving short-term building energy predictions," in *Building Simulation*, 2022, vol. 15: Springer, pp. 197-211.

[37] C. Fan, Y. Lei, Y. Sun, M. S. Piscitelli, R. Chiosa, and A. Capozzoli, "Data-centric or algorithm-centric: Exploiting the performance of transfer learning for improving building energy predictions in data-scarce context," *Energy,* vol. 240, p. 122775, 2022.

[38] H. Xue, B. P. Voutharoja, and F. D. Salim, "Leveraging language foundation models for human mobility forecasting," in *Proceedings of the 30th International Conference on Advances in Geographic Information Systems*, 2022, pp. 1-9.

[39] I. Ashrapov, "Tabular GANs for uneven distribution," *arXiv preprint arXiv:2010.00638,* 2020.

[40] I. Goodfellow *et al.*, "Generative adversarial nets," *Advances in neural information processing systems,* vol. 27, 2014.

[41] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784,* 2014.

[42] M. Lewis, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," *arXiv preprint arXiv:1910.13461,* 2019.

[43] J.-S. Chou and D.-K. Bui, "Modeling heating and cooling loads by artificial intelligence for energy-efficient building design," *Energy and Buildings,* vol. 82, pp. 437-446, 2014.

[44] L. T. Le, H. Nguyen, J. Zhou, J. Dou, and H. Moayedi, "Estimating the heating load of buildings for smart city planning using a novel artificial intelligence technique PSO-XGBoost," *Applied Sciences,* vol. 9, no. 13, p. 2714, 2019.