Building Energy Consumption Forecasting: A Comparison of Gradient Boosting Models

Abnash Bassi
abnash.bassi@mail.utoronto.ca
University of Toronto
Toronto, Canada

Anika Shenoy
anika.shenoy@mail.utoronto.ca
University of Toronto
Toronto, Canada

Arjun Sharma
ar.sharma@mail.utoronto.ca
University of Toronto
Toronto, Canada

Hanna Sigurdson
hanna.sigurdson@mail.utoronto.ca
University of Toronto
Toronto, Canada

Connor Glossop
connorglossop1@gmail.com
University of Toronto
Toronto, Canada

Jonathan H. Chan*
jonathan@sit.kmutt.ac.th
King Mongkut's University of Technology Thonburi
Bangkok, Thailand

ABSTRACT

Abstract: Building energy consumption forecasting is essential for improving the sustainability of buildings in the context of addressing climate change. Accurate building load predictions are useful for energy efficient building design selection and demand-side management initiatives. Using historical building energy consumption data has allowed researchers to develop machine learning models to improve the accuracy of such predictions, beyond inefficient traditional approaches otherwise used by the building sector. This work examines gradient boosting machine learning models, namely LightGBM, CatBoost, and XGBoost, for the purpose of comparing their performance on a select dataset. These gradient boosting models are popular in Kaggle machine learning contest solutions but have not been compared formally for the application of building energy consumption predictions. This work applies the three gradient boosting algorithms to a synthesized dataset for a large office building in Chicago. Preliminary results from the presented comparison demonstrate that XGBoost performs better than LightGBM and CatBoost when trained on the selected dataset.

CCS CONCEPTS

• **Computing methodologies** → **Boosting**; • **General and reference** → *Surveys and overviews.*

KEYWORDS

energy consumption forecasting, gradient boosting, machine learning, LightGBM, CatBoost, XGBoost

*Corresponding author is Professor Jonathan H. Chan.

IAIT2021, June 29-July 1, 2021, Bangkok, Thailand

ACM ISBN 978-1-4503-9012-5/21/06…$15.00
https://doi.org/10.1145/3468784.3470656

ACM Reference Format:
Abnash Bassi, Anika Shenoy, Arjun Sharma, Hanna Sigurdson, Connor Glossop, and Jonathan H. Chan. 2021. Building Energy Consumption Forecasting: A Comparison of Gradient Boosting Models. In *The 12th International Conference on Advances in Information Technology (IAIT2021), June 29-July 1, 2021, Bangkok, Thailand.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3468784.3470656

1 INTRODUCTION

Climate change is an increasingly pressing concern to be addressed by all facets of modern society. The United Nations has called climate change an emergency, underscoring the urgency of reducing greenhouse gas emissions [40]. Since the late 1800s, the Earth's average surface temperature has increased by 1.18 °C, largely due to an increase in CO_2 emissions [27]. This has resulted in more frequent and extreme natural disasters, food and water insecurity, and an increase in many medical conditions [17]. Reduction of greenhouse gas emissions across sectors has been at the core of many efforts to tackle climate change and the focus of public policy changes in governments around the world.

Energy consumption in buildings plays an important role in the global environmental sustainability crisis. In 2019, buildings made up 32% of primary energy consumption within the United States [49] and approximately one third of total global energy consumption [47], a number expected to increase in the coming decades [49]. Buildings are also responsible for 30% of greenhouse gas emissions worldwide [31]. It is evident that buildings will play a substantial role in addressing climate change and overall emissions reduction.

There has been a rising push for sustainability changes in the building sector, both from the perspective of construction (reducing energy consumption through intentional building design decisions such as energy efficient materials) and from a behavioural perspective (e.g. changing human energy-use behaviour within buildings) [16]. To track and further scale the efficacy of such measures, being able to predict a building's energy consumption based on its features is vital. Moreover, changes such as future smart grid innovation and demand-side management programs from electricity utilities, which incentivize consumers to reduce electricity consumption, provide an imperative for better building energy load forecasting [34], [26], [41]. Being able to predict energy consumption of buildings is also essential for improving the control of Heating, Ventilation, and

Air-Conditioning (HVAC) systems, which account for almost half of all building energy consumption in the United States [44], [12].

Traditional approaches to predicting building energy consumption often have focused on heat transfer based models, embedded in software and fundamentally designed around physics principles. These need to be tailored to the building based on considerations like window placement, floor layouts, and material choices. These approaches are limited by the availability of highly-specific building information which often is not known [45]. System generalizability across different building types and local climates may also be limited. In comparison, machine learning methods provide a powerful set of tools for predicting building energy consumption. Increased access to building, weather, and sensor data through public datasets provides the opportunity to train robust models for this purpose [49].

1.1 Purpose

This work aims to build upon existing literature that examines the application of machine learning for building energy consumption forecasting. We present a literature review of papers that use machine learning to predict building energy consumption, precluded by an explanation of the methodology used to conduct this literature review. Some of the most cutting edge work on applying machine learning models to more robust datasets for building energy consumption prediction was developed in the American Society of Heating, Refrigerating and Air-Conditioning Engineers' Great Energy Predictor III Kaggle contest [3]. The top solutions from the ASHRAE Kaggle contest all utilised various ensembles with combinations of "Light Gradient Boosting Machine (LightGBM)", "Categorical Boosting (CatBoost)", and "Extreme Gradient Boosting (XGBoost)" [24]. To the extent of the authors' knowledge, no systematic comparison of these models has been conducted for the problem of predicting building energy consumption.

This work explores the aforementioned models conceptually and conducts a side-by-side comparison of these models on a synthesized dataset. The use of the synthesized dataset is a far-reaching extension of research by Chakraborty et al. [7]. The research questions explored are two-fold: how can we improve short-term building energy consumption predictions across a range of building types, climates, etc.; and how should individuals or organizations who are looking to harness cutting-edge solutions for building energy consumption predictions more formally approach the process of selecting between different advanced gradient boosting models.

2 LITERATURE REVIEW

Given the relevance of building energy consumption predictions to both sustainability concerns and commercial cost-savings, there are decades of literature exploring the topic, with more recent publications harnessing machine learning. For this literature review, relevant review papers and technical papers that apply machine learning to building energy consumption forecasting were selected. An emphasis was placed on papers involving LightGBM, CatBoost, or XGBoost, as these were identified through the foremost ASHRAE Great Energy Predictor III Kaggle contest as efficient and highly-accurate model candidates.

2.1 Review Methodology

The research methodology in this review is based on modifications to the methodology of previous reviews of energy consumption predictive studies [2]. The methodology consists of the following four steps:

(1) Keyword-based search: A search of relevant literature was conducted using Google Scholar, IEEE Xplore, and ScienceDirect databases. These databases were selected due to their breadth of available literature and capability of sorting by relevance. Search keywords used included "building energy consumption", "gradient boosting algorithms", "XGBoost", "CatBoost", "LightGBM", and "machine learning".
(2) Screening articles for relevance: Studies that utilised at least one gradient boosting algorithm to forecast building energy consumption were eligible for inclusion. References included in each study were investigated further for relevance.
(3) Data extraction: Data was extracted by four independent reviewers using a standardized form designed *a priori*. Implementation of specific machine learning models, data preprocessing techniques, dataset features, purpose and findings of the study, limitations, and stated next steps were recorded.
(4) Identifying areas of focus for future work: The results were examined to recognize opportunities for an improved framework using gradient boosting algorithms to predict building energy consumption.

2.2 Machine Learning Approaches

Eight machine learning algorithms are commonly used for building load prediction: Linear Regression, Support Vector Machine (SVM), Neural Networks (NN), Deep Learning, Tree-based Algorithms, Hybrid Algorithms, AutoRegressive methods, Fuzzy Time Series Models, among others [49]. Supporting techniques that are employed to increase model accuracy include feature selection, extraction, clustering, and weather forecasting. In some literature, the ensemble machine learning techniques (random forests, Gradient Boosting Machines, etc.) outperformed other machine learning techniques (Multiple Linear Regression, k-Nearest Neighbours, and Support Vector Regression) [15].

In this work, we focus on literature with machine learning using gradient boosting decision trees, Light Gradient Boost, CatBoost and XGBoost. We also examine papers that use hybrid type models which include both gradient boosting and various other models.

2.2.1 Gradient Boosting. Gradient boosting algorithms iteratively combine predictions from a set of weak learners—simple decision trees where each tree uniquely splits into different subsets of features—into an aggregated model to develop final predictions [4]. The gradient boosting algorithm is based on a gradient descent optimization of a function, $F^*(x)$. The goal is finding an approximation, $\hat{F}^*(x)$, by minimizing the value of a differentiable loss function [13]. The prediction of $F^*(x)$ is built as:

$$F^*_m(x) = F^*_{m-1}(x) + p_m h_m(x) \tag{1}$$

where p_m is the weight of the m^{th} function and h_m is a function to minimize the loss function. Each $h_m(x)$ is trained on the dataset and pseudo-residuals are calculated [4].

Gradient boosting is generally effective at regression and classification [14]; however, it can be subject to overfitting [13]. Specifically, if a model perfectly fits the pseudo-residuals ($\frac{\partial L}{\partial F(x)}$ where L is the loss function), the next iteration decrements the model by the value of the pseudo-residuals, therefore producing pseudo-residuals that equal zero. This concludes the gradient boosting process before the decision can be optimized [4].

Some key hyperparameters applied to gradient boosting decision trees include:

(1) Learning rate or shrinkage parameter, $0 < v \leq 1$. The learning rate controls the size of each step during gradient descent. Values of $v \leq 0.1$ are optimal to reduce generalization error [4].
(2) Maximum depth of decision trees—typically this value is approximately 3-5 [4].
(3) Subsample rate—typically random subsampling without replacement is used. This introduces randomness to the procedure, therefore increasing variance in the base learner estimates [13].
(4) The maximum number of features to consider, similar to random forest methods [4].
(5) Minimum number (N) of samples to split an internal node. This limits the size of trees to a maximum depth of Size of Training Data - N [4].

Gradient Boosting Regression Trees (GBRT), or Gradient Boosting Decision Trees, outperformed alternative models in 4 out of 14 papers reviewed that included GBRT [46], [42], [15]. However, [14] concluded that this suggests GBRT was only particularly successful for that specific dataset and features. Although initially successful, the GBRT method seems to have been surpassed by new models such as its derivatives CatBoost [31], and XGBoost [39].

GBRT models have been the subject of comparison to many newer models. Models such as CatBoost [31], Random Forest [43], XGBoost [43], [39], AdaBoost [34], Support Vector Machine algorithm [10], Multi Layer Perceptrons [32], and Convolutional Neural Network (CNN) and Gated Recurrent Units (GRU) framework [38], have performed better than GBRT. [29] proposed comparing seven models including GBRT and XGBoost every day during prediction and subsequently using the most successful for the next day's prediction. One proposed application of the model is using it as part of a base for a stacking model, along with RF, XGBoost, SVR, and kNN, which can improve accuracy and perform better at generalization [42]. Another area of research, beyond the scope of this work, is exploring feature selection methods such as F-regression, Mutual Information, Recursive Feature Elimination, and Elastic Net in conjunction with the GBRT model [35].

2.2.2 XGBoost. Extreme Gradient Boosting (XGBoost) is a gradient boosting algorithm designed for scalability [4]. The loss function used in XGBoost models is modified:

$$L_{xgb} = \sum_{i=1}^{N} L(y_i, F(\mathbf{x}_i)) + \sum_{m=1}^{M} \Omega(h_m) \tag{2}$$

$$\Omega(h) = \gamma T + \frac{1}{2}\lambda ||w||^2 \tag{3}$$

where T is the number of leaves in a tree, λ is the L2 regularization parameter, w is the leaf weights, and γ is the minimum loss reduction. Notably, increasing the value of γ decreases tree depth [8]. Shrinkage, also known as learning rate, controls step size, similar to general gradient boosting decision trees. Other hyperparameters that are typically used for regularization include the maximum depth of tree, and hyperparameters associated with randomization techniques such as random subsampling without replacement for trees and column subsampling at a split [4]. A combination of the gradient and a Hessian (second order gradient) passed through a hyperparameter form the loss function, allowing the user to customize it as appropriate [8].

XGBoost uses several methods to increase its efficiency. As sorting the data is typically the most time-consuming process when working with trees, XGBoost stores data into blocks which are stored in compressed sparse column (CSC) format. Each column is sorted by feature. This technique only requires that sorting occur once, at the beginning of the process [8]. A split search algorithm performs a linear scan to collect the statistics of all nodes to find the optimal split at each node. Lastly, XGBoost handles null and missing values automatically, which is beneficial when working with sparse datasets [4].

The XGBoost model has been highly successful when applied in literature. XGBoost or XGBoost hybrid models outperformed alternative models in 11 of the 15 papers reviewed containing XGBoost (for example: [39], [1], [12], [44]). XGBoost is highly efficient [12], likely because of its ability to perform parallel processing. XGBoost sorts the data prior to training, saves them as a block structure, and then uses the structure in subsequent iterations. This improves efficiency by significantly reducing the computation, since one of the most time consuming steps in decision tree learning is the sorting of the values of the features. This led to XGBoost having the highest efficiency in the comparison of Wang et al. [43]. Another feature of XGBoost is the inherent capability to select and store relevant features; thus, external feature selection algorithms are not necessarily required by the XGBoost algorithm and in some cases even cause deterioration of XGBoost performance [7].

Multiple researchers combined the use of XGBoost with other algorithms. Lu et al. proposed for CEEDMAN to be used first to de-noise the raw data by decomposing it into 12 datasets. XGBoost predicts in the 12 datasets, and the prediction results are then summed and denormalized, which outperformed SVM with particle swarm optimizer, Radial basis function NN, Least squares SVM, and CEEMDAN Random Forest [23]. Additional examples include combining EMD and ARIMA with XGBoost [48], as well as only with ARIMA [21], and combining XGBoost with k-means on Similar Days selection [51]. Within the aforementioned papers, all of these XGBoost-based models outperformed alternative models used in comparison.

However, in select papers, alternative frameworks, such as the Convolutional Neural Network (CNN) and Gated Recurrent Units (GRU) framework [38], and an Artificial Neural Network [50], were able to outperform XGBoost.

2.2.3 LightGBM. As in XGBoost, Light Gradient Boosting Machine presorts a histogram of features to decrease computational complexity. There exists a wide variety of learning hyperparameters that can be applied in the model as well as several types of randomization including column randomization and bootstrap subsampling [4]. LightGBM introduces two novel techniques to XGBoost to accelerate training speed:

(1) Gradient-based one-side sampling (GOSS): In order to filter data instances to find a split, LightGBM utilises GOSS. Since data instances with larger gradients will contribute more to decisions, LightGBM randomly drops data instances with smaller gradients when downsampling. This surpasses uniform random sampling by more accurate gain estimation [18].
(2) Exclusive Feature Bundling (EFB): LightGBM bundles features which are mutually exclusive together. Since number of bundles $\ll$ number of features, the complexity of sorting the dataset into a histogram at the beginning of the process is significantly decreased. Therefore, the complexity is decreased from $O(\#\text{data} * \#\text{feature})$ to $O(\#\text{data} * \#\text{bundle})$ [18].

LightGBM and the CatBoost method have both been used in models to predict building energy consumption, but not to the same extent as Extreme Gradient Boost (XGBoost). [9] used LightGBM on its own to predict energy consumption on the ASHRAE dataset. They also found benefits of using LightGBM over other decision tree based models to include higher efficiency, faster training speeds, lower memory usage, higher accuracy, and the ability to handle large-scale data. However, it is unclear how advantageous these benefits are relative to XGBoost and CatBoost.

2.2.4 CatBoost. A key objective of a CatBoost model is to mitigate prediction shift, which occurs as a result of target leakage when the model overestimates its utility or overfits [36], [4]. This is because gradient boosting reuses the same data instances to estimate gradients and to train the models to minimize those gradients. The model's predictions are then shifted towards the distribution of estimated gradients in the dataset and further from the true distribution of gradients [36].

CatBoost prevents shifted residuals by creating a new dataset at each step of boosting by excluding a data instance from the other models created. Specifically, $n-1$ trained datasets that exclude the original instance must be created for a model with n trees [4].

CatBoost handles any type of categorical data by vectorization. This is accomplished by a procedure including permutation of training data and encoding to convert all categorical data into numerical features. For instance, one common method is one-hot encoding where a categorical feature is represented as a set of binary features. CatBoost also creates new feature combinations by bundling original features [36]. Vectorization of features also enables element-wise comparison and this is utilised to develop symmetric decision trees [4].

Unlike LightGBM, CatBoost grows trees level-wise. CatBoost is unique in the way it handles categorical features and helps avoid overfitting to the data [4]. [37] compared LightGBM to CatBoost (among other models) and CatBoost showed superior performance as well as significantly lower complexity and implementation cost. Comparing the three gradient boost methods LightGBM, CatBoost, and XGBoost, [31] showed CatBoost was superior in terms of prediction capability and computation speed. As stated earlier, the three methods have yet to be compared on the same dataset for the prediction of building energy consumption. [31] compared CatBoost to other regression methods and identified that the relevant hyperparameters to be tuned for building energy consumption prediction are max tree depth, regularization of the L2 coefficient, and learning rate. When compared to both the random forest method and the gradient boosted decision tree method, CatBoost greatly outperformed the other two with an R^2 value of 0.897.

2.3 Dataset Limitations

There is a wide range in the extensiveness of datasets used in technical papers focused on building energy consumption predictions using machine learning.

In [4], Bentéjac et al. compares LightGBM, CatBoost, and XGBoost across 28 datasets, but none of these are related to building load prediction and offer limited problem-specific insights. Wang et al. has only two buildings in the dataset used, both at the same university, and data is gathered over a limited time frame [42]. The works of [23], [37], [34], [31], [12], [14], and [46] are a select few of the seemingly many papers that present a case study of only one building and it is unclear how applicable their respective findings are to other buildings; their conclusions are challenging to generalize beyond the limited scope of each work. The features used in [23] are also limited as more specific weather features are not included. The studies by [31] and [43] used datasets with small sample sizes. Although [26] used data from 72 buildings, all were obtained from one university campus, therefore leaving the role of weather data variation across different regional climates untested. As [44] acknowledged, a limitation observed in papers evaluated was a lack of consideration of solar radiation features in the developed machine learning models. Furthermore, [10] and [31] noted the challenge of missing data points in datasets, which in the case of [10], led to machine learning models *not* performing significantly better than linear regression models for building energy consumption prediction, contrary to typical findings. Overall, this literature review demonstrated the challenge of comparing findings between papers given that datasets varied in quality and were susceptible to limitations as indicated above.

Few papers used datasets that contain data from a multitude of buildings from varying climates. [9] and [24] discuss findings based upon the ASHRAE Great Energy Predictor III Kaggle dataset, which contains more than 20 million data points from 1,448 buildings from a variety of building use types [24]. This dataset is considered to be a notable dataset available for training and validating machine learning models for building load prediction purposes, but there remains significant work to be done on increasing the availability of additional high-quality data [49]. More access to high-quality datasets would allow for better standardized comparisons of machine learning models.

3 COMPARING LIGHTGBM, CATBOOST, AND XGBOOST ON SYNTHESIZED DATA

A prior work explored the use of XGBoost on a simulated dataset [7]. As an extension, this work will compare the performance of

alternative advanced gradient boosting models, namely LightGBM versus CatBoost versus XGBoost. For the purpose of this work, the synthesized dataset was used to reduce the impact of preprocessing decisions and approaches in their influence upon the comparison of the three aforementioned models. Using the synthesized dataset also removed the potential for skewing from missing data points. [4] compared these three models for other common datasets; our work builds upon this area of research by extending such comparisons to the significant problem of building energy consumption predictions.

3.1 Datasets

The following are publicly available datasets that can be used for building energy consumption prediction.

3.1.1 Chicago Large Office Building Dataset [7]. This simulated dataset is based on a large-size commercial office building located in Chicago, USA. It includes hourly energy readings for 4 years (2012 - 2015), which amounts to 35,040 data points for each feature (35,040 rows). The location and building size were originally randomly selected [7].

This is a synthetic database used to train the models, in order to provide a consistent basis for research, energy model development, and setting benchmarks. It can be generated much faster, in seconds rather than years for real measured datasets. This dataset is obtained from EnergyPlus simulations [11]. EnergyPlus does building energy modeling, which uses a "computer-based simulation software to perform a detailed analysis of a building's energy use and energy-using systems." [11]. The file is developed by the US Department of Energy's Building Technologies Program in collaboration with the Pacific Northwest National Laboratory, Lawrence Berkeley National Laboratory, and National Renewable Energy Laboratory [28]. The weather data is from National Renewable Energy Laboratory NSRDB [19].

The features present in the dataset are: Year, Month, Day of Month, Day of Week, Hour, Is Holiday, Daylight Savings, DHI (Watts/sq. meter), DNI (Watts/sq. meter), Dew Point (Celsius), Temperature (Celsius), Pressure (mbar), Relative Humidity (%), Wind Direction (degrees), and Wind Speed (meter/s).

Potential target variables present in the dataset (in Joules): Electricity Facility, Gas Facility, Electricity Building, Electricity HVAC, Cooling Electricity, Heating Gas, Heating Electricity, Fans Electricity, and Pumps Electricity.

3.1.2 ASHRAE - Great Energy Predictor III on Kaggle [3]. This dataset contains hourly meter readings over three years from over one thousand buildings at different sites around the world. It is not primarily used in this work; more information can be found at the source [3]. It was the topic of the Kaggle contest entitled "ASHRAE - Great Energy Predictor III".

Weather data includes: Air temperature, cloud coverage, dew temperature, precipitation depth 1 hr, sea level pressure, wind direction, and wind speed. Building data includes: Primary use, square feet, year built, and floor count. The data includes a timestamp for each feature. The target variable is the meter reading which is energy consumption in kWh. Meter readings are available for each of four targets: chilled water, electric, hot water, and steam meters.

3.1.3 Building Sites Power Consumption Dataset on Kaggle [25]. This is a public dataset to be used for building energy consumption prediction, containing information from Schneider Electric. This dataset includes features such as historical consumption, building metadata, historical weather data, and public holidays. More detailed information on the dataset can be found at the source [25].

3.2 Evaluation Metrics

3.2.1 Root Mean Squared Logarithmic Error (RMSLE). RMSLE is the evaluation metric used for the Kaggle competition and dataset "ASHRAE - Great Energy Predictor III" [3]. Smaller error means higher accuracy and better prediction. This metric measures the relative distance rather than absolute so that larger values with larger differences are measured similarly to smaller values with smaller differences between actual and predicted [33]. This is useful when dealing with large values such as electricity.

RMSLE is calculated as:

$$\epsilon = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\log(p_i + 1) - \log(a_i + 1))^2} \tag{4}$$

where ϵ is the RMSLE value, n is the total number of observations in the data set, p_i is the prediction of target, a_i is the actual target for i, and $log(x)$ is the natural logarithm of x.

3.2.2 Range Normalized Root Mean Squared Error (RN-RMSE). RN-RMSE (%) is a version of root mean squared error where the range of the dataset is used for normalization. This is useful to compare models trained on different datasets with different properties [6]. Smaller error means higher accuracy and better prediction. This model was used previously with the Chicago Large Office Building Data [7].

It is calculated as:

$$RN\text{-}RMSE(\%) = \frac{\sqrt{\frac{\sum_{k=1}^{n}(T_k - P_k)^2}{n}}}{\text{maximum}(T) - \text{minimum}(T)} * 100\% \tag{5}$$

where T is target, P is predicted, n is the total number of data points.

3.2.3 Coefficient of Determination (R^2). This model represents how well the regression estimate fits the data [6]. A value of 1 suggests a perfect model, while a value of 0 suggests no correlation. This model was used previously with the Chicago Large Office Building Data [7].

It is calculated as:

$$R^2 = 1 - \frac{\sum_{k=1}^{n}(T_k - P_k)^2}{\sum_{k=1}^{n}(T_k - \mu)^2} \tag{6}$$

where T is target, P is predicted, n is the total number of data points, and μ is the average value.

4 METHODOLOGY AND PRELIMINARY RESULTS

4.1 Preprocessing, Feature Engineering, and Feature Selection

A key step in preprocessing the data involved transforming cyclical features such as month, day of the week, hour, and wind direction into sine and cosine functions similar to the preprocessing steps developed by Chakraborty et al. [7]. As a result, the lowest and highest value of each feature are one value apart; for instance, hour 0 and hour 23 become 1 hour apart after feature engineering. The dataset for the CatBoost model was further modified by converting float values into integer values since CatBoost supports only integer features or text features [5]. Dataset features were then selected for inclusion if they were also included in the ASHRAE dataset due to the possibility of a future extension to this study in transfer learning applications. Specifically, meteorological and temporal features from the Chicago dataset were used to predict electricity consumption for building cooling for each gradient-boosting model analyzed.

4.2 Approach and Results

4.2.1 LightGBM. First the LightGBM model was trained using default hyperparameters and an arbitrary learning rate as a naive benchmark. Then, the learning rate was tuned, scaling the learning rate by factors of three, ranging from 0.0001 to 0.3. It was observed that an increase in the learning rate led to a decrease in all three error metrics. This was notably the opposite of what was expected for LightGBM hyperparameter tuning, wherein increasing the learning rate typically increases error [22]. Increasing the parameter num_leaves to 50 improved the scores of all three metrics but increasing even further to 100 showed worse results. As per LightGBM's documentation, a high num_leaves can result in overfitting if max_depth is left unconstrained and a low min_data value can lead to overfitting as there may not be enough observations to generalize well. The min_data parameter determines the minimum number of observations a tree node must have before being added. To avoid overfitting, a max_depth value of 20 and min_data was doubled from 50 (default) to 100. The model saw improvement with respect to each of the metrics when changing min_data and no change afterwards when increasing max_depth. To further reduce the likelihood of over-fitting, the train-test split was also adjusted. After a few iterations of testing different values, it was found that the best performance was with a split of train_size=0.9 (test_size=0.1) along with a higher num_leaves. Documentation also states that having a feature_fraction value < 1 is likely to reduce overfitting, so it is possible that the highest scores obtained after setting feature_fraction to 1 do involve some overfitting. Tuning other parameters such as early_stopping_rounds resulted in no significant changes. Ultimately, the random hyperparameter tuning resulted in the best scores based on the evaluation metrics.

Finally, there has been a surge in the prevalence of cutting-edge automatic machine learning tools, also known as AutoML, which can select optimal learners and conduct improved hyperparameter tuning automatically. As suggested in the LightGBM documentation, the simple LightGBM Tuner from the Optuna library was utilised to conduct further hyperparameter tuning for the LightGBM model at hand [22]. This simple LightGBM Tuner uses a stepwise algorithm for hyperparameter tuning, tuning each hyperparameter in a sequential fashion [30]. The hyperparameters tuned by the LightGBM Tuner are the feature fraction, the number of leaves, the bagging fraction, the regularization term, and the minimum number of samples in a leaf [30]. LightGBM results are summarized below in Table 1.

4.2.2 CatBoost. A CatBoost model was then trained using the same preprocessing steps as in the LightGBM model to predict electricity to cool a building. First, the model was trained using default hyperparameters (depth: 6, learning rate: 0.03, iterations: 1000). Hyperparameter tuning was then conducted to find optimal hyperparameters using CatBoostRegressor from the CatBoost library and GridSearchCV from the Scikit-learn library. This method uses 5-fold cross validation with grid-search over a parameter grid to optimize the combination from a list of user-selected parameters[20]. The hyperparameter grid included depth (1, 6, 8, and 10), learning rate (0.01, 0.03 and 0.1), and iterations (30, 50, 100, and 500). The results are summarized in Table 2.

4.2.3 XGBoost. Similarly to the LightGBM and CatBoost model, the XGBoost model was trained on the Chicago dataset using the same features and preprocessing steps. All values were converted to 32bit floats. The model was initially trained using the default XGBoostRegressor model parameters (depth: 6, learning rate: 0.3, iterations: 1000). Hyperparameter tuning was then performed using GridSearchCV from the Scikit-learn library, as described above. The hyperparameters tuned were depth (5, 7 and 10), learning rate (0.01, 0.05, 0.1) and number of iterations (50, 100, 500, 1000). The results of the default and optimized models are summarized below in Table 3.

4.3 Discussion

By using LightGBM, CatBoost, and XGBoost on a single dataset, the best results from these gradient boosting models can be compared to results obtained by [7] solely for XGBoost. Comparative results for building cooling electricity prediction are presented below in Table 4 on page 8. Both our LightGBM and XGBoost models used the same feature engineering as in [7] while also using feature selection to make sure the models would be compatible with other datasets. Our CatBoost model added additional feature engineering for CatBoost-specific model compatibility. While LightGBM's top score had its hyperparameters tuned by hand, the top scores from the CatBoost and XGBoost models were arrived at using a gridsearch algorithm for hyperparameter optimization.

Hyperparameter optimization was conducted in the previous section. For LightGBM, hyperparameter optimization using Optuna improved metrics but did not lead to optimal results; these results were presented previously in Table 1. For CatBoost, the optimized set of parameters produced a lower RMSLE and a lower RN-RMSE (%). The CatBoost R^2 scores produced by both sets of parameters are not statistically different; this is summarized in Table 2. The XGBoost optimized parameters resulted in a slightly lower RMSLE

Table 1: RMSLE, RN-RMSE (%), and R^2 score by hyperparameters set for LightGBM.

Hyperparameters				*Metrics*		
Learning Rate	Number of Leaves	Minimum Data	Maximum Depth	RMSLE	RN-RMSE (%)	R^2
Changing Learning Rate						
0.0001	10	50	10	1.2204	20.6808	0.0173
0.0003	10	50	10	1.2049	20.3195	0.0513
0.001	10	50	10	1.1545	19.1077	0.1611
0.003	10	50	10	1.0285	16.0605	0.4074
0.01	10	50	10	0.6984	9.0326	0.8125
0.03	10	50	10	0.2427	3.5253	0.9714
0.1	10	50	10	0.1301	2.4485	0.9862
0.3	10	50	10	0.1179	2.1862	0.9890
0.5	10	50	10	0.1213	2.1287	0.9896
Changing Minimum Data						
0.5	10	100	10	0.1204	2.1333	0.9895
Changing Maximum Depth						
0.5	10	100	20	0.1204	2.1333	0.9895
Changing Number of Leaves						
0.5	50	100	20	0.1087	2.1619	0.9893
0.5	100	100	20	0.1142	2.2263	0.9886
Changing Feature Fraction to 1.0 *(Previous tests done with* 0.5*)*						
0.5	50	100	20	0.0940	1.9995	0.9908
Changing train_size to 0.5, 0.7, 0.9, *&* 0.9 *with higher num_leaves (Previous tests done with* 0.75*)*						
0.5	50	100	20	0.0996	2.1523	0.9893
0.5	100	100	20	0.0863	2.0234	0.9906
0.5	50	100	20	0.1014	2.0820	0.9910
0.5	100	100	20	0.0910	1.9915	0.9917
Using Optuna with bagging fraction of 0.965, lambda_l1 of $1.94252402914098 * 10^{-8}$*, and lambda_l2 of* $1.5691107062976204 * 10^{-5}$						
0.003	623	50	10	0.1585	2.3616	0.9872

Table 2: RMSLE, RN-RMSE (%), and R^2 score by hyperparameters set for CatBoost.

	Hyperparameters			*Metrics*		
	Learning Rate	Iterations	Maximum Depth	RMSLE	RN-RMSE (%)	R^2
Default	0.03	1000	6	0.1244	3.6270	0.9698
Optimal	0.1	500	8	0.1214	3.6180	0.9699

Table 3: RMSLE, RN-RMSE (%), and R^2 score by hyperparameters set for XGBoost.

	Hyperparameters			*Metrics*		
	Learning Rate	Iterations	Maximum Depth	RMSLE	RN-RMSE (%)	R^2
Default	0.3	1000	6	0.0894	1.8724	0.9920
Optimized	0.1	1000	5	0.0879	1.9112	0.9917

value while causing a slight increase in the RN-RMSE (%). The R^2 values are nearly identical. These results were presented in Table 3. This is quite similar to what was seen previously when optimizing the CatBoost model.

The possibility of overfitting was investigated by comparing each model's performance (in terms of RMSLE, RN-RMSE (%), and R^2) after each iteration of 5-fold cross validation. The results are summarized in Table 5 on page 8. There was no indication of overfitting based on an examination of the mean and sample standard deviation performance of the five iterations.

Table 4: Metric comparison table with best performance for predicting cooling electricity consumption from CatBoost, LightGBM, and our XGBoost versus the Chicago dataset paper on XGBoost.

Metric	*[7]'s XGBoost*	*Our LightGBM*	*Our CatBoost*	*Our XGBoost*
RN-RMSE (%)	2.43	1.9915	3.6180	1.8724
R^2	0.99	0.9917	0.9699	0.9920

Table 5: Average and Standard Deviation in RMSLE, RN-RMSE (%), and R^2 score for 5-fold cross validation using LightGBM, CatBoost, XGBoost model.

Sample Average ± Standard Deviation	RMSLE	RN-RMSE (%)	R^2
LightGBM	0.1564 ± 0.0230	2.4031 ± 0.6054	0.9864 ± 0.0040
CatBoost	0.1217 ± 0.0018	3.6212 ± 0.0050	0.9699 ± 0.0001
XGBoost	0.0161 ± 0.0025	2.7386 ± 0.4583	0.9819 ± 0.0039

5 CONCLUSION

Predicting energy consumption in all buildings has a significant impact on addressing climate change, reducing energy consumption, innovating smart grids, controlling HVAC systems, and selecting energy-efficient building design candidates. Therefore, developing reliable and effective machine learning models to predict building energy consumption is a valuable endeavor. In this work, a literature review was conducted that summarized the previous uses of gradient boosting techniques: gradient boosting regression trees, XGBoost, LightGBM, and CatBoost, applied to building energy consumption prediction. Each model was explained and reviewed. This existing literature was extended by conducting an experimental comparison of XGBoost, LightGBM, and CatBoost on a synthetic dataset. Based on our three key metrics (RMSLE, RN-RMSE (%), R^2), the XGBoost model performed best on this dataset.

Limitations of this work include the synthesized nature of the dataset. Future applications of the models to real measured data would provide strengthened conclusions on the models. Furthermore, although this work compares three gradient boosting algorithms, there is room for further investigation into the comparison of the models. For example, a comparison of runtime performance was not conducted, namely because machine specific runtime is not broadly applicable to various datasets or problems.

In future work, transfer learning and domain adaptation may be applied to use the models on diverse datasets. Transfer component analysis would also strengthen the application of transfer learning and domain adaptation. As discussed in the limitations, a future step is comparing the model on a real measured dataset to the results obtained on the synthetic dataset. For example, this may include using the ASHRAE Great Energy Predictor III dataset on Kaggle [3]. Another future work may include further examination of the preprocessing steps to improve the models, potentially developing some basic preprocessing software to execute preprocessing on various datasets.

REFERENCES

[1] Mabrook Al-Rakhami, Abdu Gumaei, Ahmed Alsanad, Atif Alamri, and Mohammad Mehedi Hassan. 2019. An Ensemble Learning Approach for Accurate Energy Load Prediction in Residential Buildings. *IEEE Access* 7 (2019), 48328–48338. https://doi.org/10.1109/ACCESS.2019.2909470

[2] Kadir Amasyali and Nora M. El-Gohary. 2018. A review of data-driven building energy consumption prediction studies. *Renewable and Sustainable Energy Reviews* 81, 1 (Jan. 2018), 1192–1205. https://doi.org/10.1016/j.rser.2017.04.095

[3] ASHRAE. 2019. ASHRAE - Great Energy Predictor III. https://www.kaggle.com/c/ashrae-energy-prediction

[4] Candice Bentéjac, Anna Csörgő, and Gonzalo Martínez-Muñoz. 2021. A comparative analysis of gradient boosting algorithms. *Artificial Intelligence Review* 54 (03 2021). https://doi.org/10.1007/s10462-020-09896-5

[5] CatBoost. 2021. *Transforming categorical features to numerical features.* Yandex. https://catboost.ai/docs/concepts/algorithm-main-stages_text-to-numeric.html?lang=en

[6] Debaditya Chakraborty and Hazem Elzarka. 2018. Performance testing of energy models: are we using the right statistical metrics? *Journal of Building Performance Simulation* 11, 4 (2018), 433–448. https://doi.org/10.1080/19401493.2017.1387607

[7] Debaditya Chakraborty and Hazem Elzarka. 2019. Advanced machine learning techniques for building performance simulation: a comparative analysis. *Journal of Building Performance Simulation* 12, 2 (2019), 193–207. https://doi.org/10.1080/19401493.2018.1498538

[8] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (San Francisco, California, USA) *(KDD '16)*. Association for Computing Machinery, New York, NY, USA, 785—-794. https://doi.org/10.1145/2939672.2939785

[9] Shamit Roy Chowdhury, Sushruta Mishra, Annie Olivia Miranda, and Pradeep Kumar Mallick. 2021. Energy Consumption Prediction Using Light Gradient Boosting Machine Model. *Advances in Power Systems and Energy Management Lecture Notes in Electrical Engineering* 690 (Jan. 2021), 413–422. https://doi.org/10.1007/978-981-15-7504-4_39

[10] Hengfang Deng, David Fannon, and Matthew J. Eckelman. 2018. Predictive modeling for US commercial building energy use: A comparison of existing statistical and machine learning algorithms using CBECS microdata. *Energy and Buildings* 163 (March 2018), 34–43. https://doi.org/10.1016/j.enbuild.2017.12.031

[11] EnergyPlus. 2021. EnergyPlus | EnergyPlus. https://energyplus.net/

[12] Cheng Fan, Fu Xiao, and Yang Zhao. 2017. A short-term building cooling load prediction method using deep learning algorithms. *Applied Energy* 195 (2017), 222–233. https://doi.org/10.1016/j.apenergy.2017.03.064

[13] Jerome H. Friedman. 2002. Stochastic gradient boosting. *Computational Statistics & Data Analysis* 38, 4 (2002), 367–378. https://doi.org/10.1016/S0167-9473(01)00065-2 Nonlinear Methods and Data Mining.

[14] Alfonso González-Briones, Guillermo Hernández, Tiago Pinto, Zita Vale, and Juan M. Corchado. 2019. A Review of the Main Machine Learning Methods for Predicting Residential Energy Consumption.. In *2019 16th International Conference on the European Energy Market (EEM)*. IEEE, Ljubljana, Slovenia, 1–6. https://doi.org/10.1109/EEM.2019.8916406

[15] Monika Goyal, Mrinal Pandey, and Rahul Thakur. 2020. Exploratory Analysis of Machine Learning Techniques to predict Energy Efficiency in Buildings. In *2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*. IEEE, Noida, India, 1033–1037. https://doi.org/10.1109/ICRITO48877.2020.9197976

[16] Jessica Grove-Smith, Vera Aydin, Wolfgang Feist, Jürgen Schnieders, and Stefan Thomas. 2018. Standards and policies for very high energy efficiency in the urban building sector towards reaching the 1.5°C target. *Current Opinion in Environmental Sustainability* 30 (2018), 103–114. https://doi.org/10.1016/j.cosust.2018.04.006 1.5°C Climate change and urban areas.

[17] Intergovernmental Panel on Climate Change. 2021. *Global Warming of 1.5°C. An IPCC Special Report.* Technical Report. United Nations. https://www.ipcc.ch/sr15/chapter/spm/

[18] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems* 30 (2017), 3146–3154.

[19] National Renewable Energy Laboratory. 2021. NSRDB Data Viewer. https://maps.nrel.gov/nsrdb-viewer/?aL=x8CI3i%255Bv%255D%3Dt%26Jea8x6%255Bv%255D%3Dt%26Jea8x6%255Bd%255D%3D1%26VRLt_G%255Bv%255D%3Dt%26VRLt_G%255Bd%255D%3D2%26mcQtmw%255Bv%255D%3Dt%26mcQtmw%255Bd%255D%3D3&bL=clight&cE=0&lR=0&mC=4.740675384778373%2C22.8515625&zL=2

[20] Scikit Learn. 2021. *sklearn.model_selection.GridSearchCV.* SciKit Learn. https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

[21] Pin Li and Jin-Suo Zhang. 2018. A New Hybrid Method for China's Energy Supply Security Forecasting Based on ARIMA and XGBoost. *Energies* 11, 7 (2018), 1–28. https://doi.org/10.3390/en11071687

[22] LightGBM. 2021. *Parameters Tuning.* Microsoft Corporation. https://lightgbm.readthedocs.io/en/latest/Parameters-Tuning.html

[23] Hongfang Lu, Feifei Cheng, Xin Ma, and Gang Hu. 2020. Short-term prediction of building energy consumption employing an improved extreme gradient boosting model: A case study of an intake tower. *Energy* 203 (2020), 117756. https://doi.org/10.1016/j.energy.2020.117756

[24] Clayton Miller, Pandarasamy Arjunan, Anjukan Kathirgamanathan, Chun Fu, Jonathan Roth, June Young Park, Chris Balbach, Krishnan Gowri, Zoltan Nagy, Anthony D. Fontanini, and Jeff Haberl. 2020. The ASHRAE Great Energy Predictor III competition: Overview and results. *Science and Technology for the Built Environment* 26, 10 (2020), 1427–1447. https://doi.org/10.1080/23744731.2020.1795514

[25] Mobius. 2021. Building Sites Power Consumption | Kaggle. https://www.kaggle.com/arashnic/building-sites-power-consumption-dataset

[26] Jihoon Moon, Jinwoong Park, Eenjun Hwang, and Sanghoon Jun. 2018. Forecasting power consumption for higher educational institutions based on machine learning. *The Journal of Supercomputing* 74, 8 (01 Aug. 2018), 3778–3800. https://doi.org/10.1007/s11227-017-2022-x

[27] National Aeronautics and Space Administration. 2021. *Climate Change: How Do We Know?* NASA. https://climate.nasa.gov/evidence/

[28] United States Department of Energy. 2020. Commercial Prototype Building Models | Building Energy Codes Program. https://www.energycodes.gov/development/commercial/prototype_models

[29] Simona-Vasilica Oprea and Adela Bâra. 2019. Machine Learning Algorithms for Short-Term Load Forecast in Residential Buildings Using Smart Meters, Sensors and Big Data Solutions. *IEEE Access* 7 (2019), 177874–177889. https://doi.org/10.1109/ACCESS.2019.2958383

[30] Kohei Ozaki. 2020. *LightGBM Tuner: New Optuna Integration for Hyperparameter Optimization.* Optuna. https://medium.com/optuna/lightgbm-tuner-new-optuna-integration-for-hyperparameter-optimization-8b7095e99258

[31] Yue Pan and Limao Zhang. 2020. Data-driven estimation of building energy consumption with multi-source heterogeneous data. *Applied Energy* 268 (2020), 114965. https://doi.org/10.1016/j.apenergy.2020.114965

[32] Nikolaos G. Paterakis, Elena Mocanu, Madeleine Gibescu, Bart Stappers, and Walter van Alst. 2017. Deep learning versus traditional machine learning methods for aggregated energy demand prediction. In *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe).* IEEE, Turin, Italy, 1–6. https://doi.org/10.1109/ISGTEurope.2017.8260289

[33] Peltarion. 2021. Mean squared logarithmic error (MSLE) | Peltarion Platform. https://peltarion.com/knowledge-center/documentation/modeling-view/build-an-ai-model/loss-functions/mean-squared-logarithmic-error-(msle)

[34] Tiago Pinto, Isabel Praça, Zita Vale, and Jose Silva. 2021. Ensemble learning for electricity consumption forecasting in office buildings. *Neurocomputing* 423 (2021), 747–755. https://doi.org/10.1016/j.neucom.2020.02.124

[35] Aida Mehdipour Pirbazari, Antorweep Chakravorty, and Chunming Rong. 2019. Evaluating Feature Selection Methods for Short-Term Load Forecasting. In *2019 IEEE International Conference on Big Data and Smart Computing (BigComp).* IEEE, Kyoto, Japan, 1–8. https://doi.org/10.1109/BIGCOMP.2019.8679188

[36] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. 2017. *CatBoost: unbiased boosting with categorical features.* Technical Report. arXiv. arXiv:arXiv:1706.09516

[37] Y D Qin, J Ke, and B Wang. 2021. Exploring New Building Energy Saving Control Strategy Application under the Energy Internet of Things. *IOP Conference Series: Earth and Environmental Science* 701, 1 (March 2021), 012060. https://doi.org/10.1088/1755-1315/701/1/012060

[38] Muhammad Sajjad, Zulfiqar Ahmad Khan, Amin Ullah, Tanveer Hussain, Waseem Ullah, Mi Young Lee, and Sung Wook Baik. 2020. A Novel CNN-GRU-Based Hybrid Approach for Short-Term Residential Load Forecasting. *IEEE Access* 8 (2020), 143759–143768. https://doi.org/10.1109/ACCESS.2020.3009537

[39] Saleh Seyedzadeh, Farzad Pour Rahimian, Parag Rastogi, and Ivan Glesk. 2019. Tuning machine learning models for prediction of building energy loads. *Sustainable Cities and Society* 47 (2019), 101484. https://doi.org/10.1016/j.scs.2019.101484

[40] United Nations. 2021. *The Climate Crisis – A Race We Can Win.* United Nations. https://www.un.org/en/un75/climate-crisis-race-we-can-win

[41] United States Energy Information Administration. 2001. *Electric Utility Demand Side Management - Archive.* Technical Report. United States Energy Information Administration. https://www.eia.gov/electricity/data/eia861/dsm/

[42] Ran Wang, Shilei Lu, and Wei Feng. 2020. A novel improved model for building energy consumption prediction based on model integration. *Applied Energy* 262, 1 (March 2020), 012060. https://doi.org/10.1016/j.apenergy.2020.114561

[43] Ran Wang, Shilei Lu, and Qiaoping Li. 2019. Multi-criteria comprehensive study on predictive algorithm of hourly heating energy consumption for residential buildings. *Sustainable Cities and Society* 49 (2019), 101623. https://doi.org/10.1016/j.scs.2019.101623

[44] Zhe Wang, Tianzhen Hong, and Mary Ann Piette. 2020. Building thermal load prediction through shallow machine learning and deep learning. *Applied Energy* 263 (2020), 114683. https://doi.org/10.1016/j.apenergy.2020.114683

[45] Hai xiang Zhao and Frédéric Magoulès. 2012. A review on the prediction of building energy consumption. *Renewable and Sustainable Energy Reviews* 16, 6 (2012), 3586–3592. https://doi.org/10.1016/j.rser.2012.02.049

[46] Xiaotong Yu, Zhezhuang Xu, Xin Zhou, Jiejun Zheng, Yuxiong Xia, Larry Lin, and Shih-Hau Fang. 2019. Load Forecasting Based on Smart Meter Data and Gradient Boosting Decision Tree. In *2019 Chinese Automation Congress (CAC).* IEEE, Hangzhou, China, 4438–4442. https://doi.org/10.1109/CAC48633.2019.8996810

[47] Wu Yucong and Wang Bo. 2020. Research on EA-Xgboost Hybrid Model for Building Energy Prediction. *Journal of Physics: Conference Series* 1518 (04 2020), 012082. https://doi.org/10.1088/1742-6596/1518/1/012082

[48] Wu Yucong and Wang Bo. 2020. Research on EA-Xgboost Hybrid Model for Building Energy Prediction. *Journal of Physics: Conference Series* 1518 (04 2020), 012082. https://doi.org/10.1088/1742-6596/1518/1/012082

[49] Liang Zhang, Jin Wen, Yanfei Li, Jianli Chen, Yunyang Ye, Yangyang Fu, and William Livingood. 2021. A review of machine learning in building load prediction. *Applied Energy* 285 (2021), 116452. https://doi.org/10.1016/j.apenergy.2021.116452

[50] Yong Zhang, Xiaomin Zhou, Yinglei Teng, Jinghui Fang, and Weijun Zheng. 2018. Resource Allocation for Multi-User MEC System: Machine Learning Approaches. In *2018 International Conference on Computational Science and Computational Intelligence (CSCI).* IEEE, Las Vegas, NV, USA, 794–799. https://doi.org/10.1109/CSCI46756.2018.00159

[51] Huiting Zheng, Jiabin Yuan, and Long Chen. 2017. Short-Term Load Forecasting Using EMD-LSTM Neural Networks with a Xgboost Algorithm for Feature Importance Evaluation. *Energies* 10, 8 (2017), 1–20. https://doi.org/10.3390/en10081168