ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

FORECASTING ELECTRICITY CONSUMPTION THROUGH A FUSION OF HYBRID RANDOM FOREST REGRESSION AND LINEAR REGRESSION MODELS UTILIZING SMART METER DATA

DR. B. RAVI PRASAD[1], MR. DUDEKULA SIDDAIAH[2], PROF. TS. DR. YOUSEF A.BAKER EL-EBIARY[3], DR. S. NAVEEN KUMAR4, DR. K SELVAKUMAR[5]

[1]Professor, Department of CSE Marri Laxman Reddy Institute of Technology and Management, Hyderabad- 500043

[2]Research Scholar, Department of Computer Science, YBN University, Ranchi- 834010

[3]Faculty of Informatics and Computing, UniSZA University, Malaysia

[4]Associate Professor, Department of Computer Science and Engineering, Mother Teresa Institute of Engineering and Technology, Palamaner

[5]Professor, Department of Information Technology, Annamalai university, Chidambaram

[1]rprasad.boddu@gmail.com, [2]siddu.associate@gmail.com, [3]yousefelebiary@unisza.edu.my, [4]navinkumar781@gmail.com, [5]kskaucse@gmail.com.

ABSTRACT

This project seeks to forecast energy usage by utilizing data from smart meters and the Random Forest Regressor algorithm. A logical flow of processes includes data preparation, model training, and model performance assessment. The use of the Label Encoder method to the dataset initiates the conversion of category variables into numerical representations. This crucial phase guarantees that the model can skillfully handle the data processing. After that, missing data are handled using the Simple Imputer technique, which judiciously fills in the blanks with suitable measurements like mean or median values. The train_test_split function divides the dataset into training and testing subsets, preparing the way for model training. The Hybrid Random Forest Regressor approach is used in combination with the LR methodology to train the predictive model. Numeric characteristics are standardized using the Min Max scaling approach, which aligns them into a common range, to ensure the best model performance. A wide range of evaluation measures, such as mean_absolute_error, mean_squared_error, and median_absolute_error, are used to evaluate the model's effectiveness after training. These metrics provide a lot of insightful information by measuring the precision and accuracy of the model's forecasting abilities. The Random Forest Regressor algorithm, together with various preprocessing techniques, allows this research to forecast energy use from smart meter data with a high degree of accuracy. A spectacular Mean Absolute Error of proposed method is 70.79, outperforms over existing methods, SARIMA and SVR+LR and an equally excellent Median Absolute Error of 30.46 are achieved by the resulting model. The proposed model is implemented using Python software. These error rates provide quantifiable benchmarks that reveal the model's performance features and are an indication of its extraordinary predictive precision. The study's findings have enormous potential for improving cost effectiveness, eco-aware practices, and energy management effectiveness.

Keywords: *Smart Meter Data; Energy Consumption; Random Forest Regressor; Simple Imputer; Min Max Scale*r

1 INTRODUCTION

Smart meters are cutting-edge tools that monitor and log power use in both residential and commercial facilities on a regular basis. They offer thorough data on energy use trends, enabling more precise invoicing, effective energy management, and the opportunity for demand response programmes. Smart meters have spread widely in recent years, generating enormous volumes of data on energy usage. Accurately estimating future energy demand is one of the chief problems facing the energy business. For a variety of stakeholders, including power utilities, grid operators, and consumers, accurate projections are essential because they allow for improved resource allocation, load balancing, and cost optimization [1]. Data from smart meters is a useful tool for forecasting energy use since it gives in-depth details on individual energy usage trends. Using previous data from smart meters and other pertinent variables, models are created for smart meters that may predict future energy use. This prediction assignment frequently includes both short- and long-term forecasting (for example, projecting energy usage for the upcoming day or week or for the upcoming month or year). Several machine learning and statistical modelling approaches are used to address this prediction job. Regression models, time series analysis, ensemble techniques, and deep learning strategies are some of these. The model used will rely on the particulars of the prediction job, the qualities of the data, and the computational resources that are available [2].

In addition to the smart meter data itself, other factors can influence energy consumption and need to be considered in the prediction models [3]. These variables might include the season, the period of day, the time of the week, holidays, and special occasions. Including these outside variables in the prediction models can increase their precision and allow for more accurate forecasts. There are various advantages to making accurate projections of energy usage using data from smart meters. They give electric utilities the ability to maximize the production and distribution of energy, which results in more effective resource management and cost reductions. Consumers might possibly lower their energy costs by having a greater understanding of and control over their energy consumption. Additionally, precise forecasts support demand response programmes, where users can reduce their energy consumption during peak hours to ease system stress [4]. In conclusion, smart meter energy consumption prediction is essential for maximizing energy management, lowering expenses, and fostering effective resource allocation [5]. Predictive models may offer precise projections by utilizing past smart meter data and other pertinent variables, which will help power utilities and customers in the transition to a more sustainable and energy-efficient future. A popular machine learning approach for regression problems, such as estimating energy usage in the context of data from smart meters, is the Random Forest Regressor. It is a system of ensemble learning which employs a number of decision trees to offer precise forecasts. The smart meter dataset may be used to train a prediction model using the Random Forest Regressor. The Random Forest Regressor algorithm excels at handling intricate correlations and interactions between features, which makes it excellent for identifying patterns of energy usage in data from smart meters. It makes use of the group of decision trees to produce reliable forecasts that are accurate [6]. Once the model has been trained and predictions have been made, additional assessment and analysis may be done using the right metrics that are root mean squared error or mean absolute error, to judge the model's performance and make any required adjustments. In order to anticipate patterns of energy use, this study investigates a unique method that combines the abilities of Hybrid Random Forest Regression and Linear Regression models. This study intends to increase the accuracy of consumption projections using the comprehensive dataset offered by smart meters, assisting utilities, decision-makers, and users in maximizing energy management and resource allocation. The hybrid RF and LR model combines the predictions of both models to achieve a more accurate and reliable prediction. The RF model captures the complex non-linear patterns in the data, while the LR model provides a linear approximation to capture the overall trend and linear relationships. The combination of these models leverages the strengths of both approaches and yields improved prediction performance. To implement the hybrid model, the predictions from both the RF and LR models are combined using weighted averages [7]. The weights assigned to each model can be determined based on the performance of each model on the training data or through expert knowledge. The weights reflect the relative importance or confidence in the predictions of each model. By utilizing the hybrid RF and LR model for energy consumption prediction, we can benefit from the flexibility and accuracy of RF in capturing complex patterns, as well as the interpretability and simplicity of LR. This approach provides a robust and reliable prediction model that can be applied in energy management systems, cost optimization, and decision-making processes. Utilizing data from smart meters, this design combines the advantages of both the Hybrid Random Forest Regression and Linear Regression models. By utilizing the power of sophisticated data-driven approaches, it successfully addresses the study objectives and provides a robust method for forecasting electricity use, leading to a more precise and insightful knowledge of consumption trends [8].

The key contributions of the project "Smart Meter Prediction for Energy Consumption" using the Random Forest Regressor algorithm can include:

- The project introduces a hybrid model merging Random Forest and Linear Regression, harnessing their distinct strengths to enhance prediction accuracy while maintaining interpretability.
- Through Label Encoder and Simple Imputer techniques, the project ensures seamless

conversion of categorical variables and effective handling of missing data, improving model robustness.

- Utilizing Min-Max Scaler, the project standardizes numerical features, preventing dominance issues, and promoting efficient convergence during model training.
- By assessing mean absolute error, mean squared error, and median absolute error, the project offers a comprehensive understanding of prediction performance, aiding nuanced decision-making.
- Accurate electricity consumption predictions empower practical energy management decisions, enabling cost-effective energy utilization and aligning with sustainability objectives.

This essay is organized as follows: While Section 3 elaborates on the issue description; Section 2 provides relevant work that is intended to grasp the proposed paper using the existing approaches. The proposed Smart meter technique for forecasting energy use is shown in the fourth segment. Section 5 tabulates and visually displays the outcomes and performance measures. Chapter 6 concludes with a presentation of the conclusion and future work.

2 LITERATURE REVIEW

Jaiswal, Chakravorty, and Rong [9] presents a paper titled "Distributed Fog Computing Architecture for Real-Time Anomaly Detection in Smart Meter Data" proposes a hierarchical Fog Computing approach to address the difficulties in processing real-time sensor data for detecting abnormalities in power consumption. This approach is suggested as an alternative to Cloud Computing Offers a pertinent and timely contribution. While the potential advantages of the proposed architecture are emphasized, clarity might be improved by a more in-depth description of the particular benefits of fog computation and a comparison with alternatives based on the cloud. The paper's applicability and influence in the realm of IoT and Big Data analysis might also be enhanced by offering more implementation insights, resolving security and privacy issues, and verifying the method with actual data. The robustness of the article might be improved by considering the security and privacy elements of the suggested architecture, which includes data encryption and access restrictions; given smart meter data contains sensitive information about homes. The deployment of a systematically decentralized Fog Computing architecture advances the field of real-time detection of anomalies in smart meter data. It effectively emphasizes the benefits of computational fogging over standard cloud computing in dealing with the difficulties of processing large amounts of data. The study might increase its effect and offer helpful insights for both researchers and professionals working in the fields of the Internet of Things (IoT), big data, and collaborative computing by considering the areas for advancement stated above.

Sajjad et al. [10] elaborates that the crucial area of electric energy forecasting is addressed in the research article "A Novel CNN-GRU-Based Hybrid Approach for Short-Term Residential Load Forecasting." with a hybrid model combining Convolutional Neural Network (CNN) and Gated Recurrent Units (GRU). The authors persuasively make the case that current methods including GBR, ANNs, ELM, and SVM have difficulties capturing non-linear correlations and the flexibility of real-world situations, necessitating the development of more reliable and precise forecasting methods. The research offers a structured framework using data improvement and training stages, utilizing CNN for the extraction of features and GRU for improved sequence learning, through the suggested hybrid model. The model outperforms the competition in terms of computational effectiveness, prediction accuracy, and feature extraction, which may be explained by CNN's feature extraction capabilities and GRU's efficient gated structure. AEP (Appliances Energy Prediction) and IHEPC (Individual Household Electric Power Consumption) datasets more specifically show minimized error rates compared to baseline models, which supports the method's effectiveness in the experimental validation across developed energy forecasting datasets. The paper's new hybrid technique, which provides a viable route for increasing the accuracy and effectiveness of short-term domestic load forecasting, is its main contribution.

Buzau et al. [11] Presents the paper which contributes significantly to the field of electricity utilities by tackling the substantial issue of non-technical losses, which have a substantial impact on revenue. The proposed solution introduces an innovative and comprehensive approach to address this challenge. By employing a hybrid deep neural network, the paper emphasizes the potential of cutting-edge technology to revolutionize anomaly and fraud detection in smart meters. Notably, the paper's key strength lies in its ability to autonomously learn relevant features from raw data, thus circumventing the labor-intensive process of handcrafted feature engineering. The architectural design of the hybrid model, integrating a long short-term memory (LSTM) network and a multi-layer perceptrons (MLP) network, is strategically devised to handle both sequential and non-sequential data. The LSTM network adeptly captures patterns and trends within daily energy consumption history, while the MLP network accommodates additional contextual information such as contracted power and geographical details. This holistic approach enables the model to make comprehensive decisions, effectively addressing the multifaceted nature of non-technical losses. The empirical evaluation is a testament to the efficacy of the proposed hybrid neural network. Through thorough experimentation on real smart meter data from a prominent electricity utility, the model demonstrates remarkable performance improvements when compared to both state-of-the-art classifiers and previously employed deep learning models for non-technical losses detection. This substantiates the robustness and practical applicability of the proposed approach, providing a viable solution for an industry-wide concern. Utilizing real smart meter data from a prominent electricity utility, the study provides a robust approach that not only addresses a pressing industry challenge but also showcases practical applicability and advancement over the state-of-the-art techniques.

Electric utility companies throughout the world have suffered significant financial losses as a result of the urgent issue of electricity theft. Every year, power worth $6 billion is stolen in the United States alone. Physical assaults like line tapping or meter manipulation are frequently used to steal electricity from consumers. New kinds of power theft efforts are made feasible by the smart grid concept. First, cybercrime may be used to perpetrate electricity theft. Smart meters are put at purchasers' locations and routinely report the customers' usage for monitoring and invoicing reasons via the advanced metering infrastructure (AMI). In this situation, unscrupulous consumers may hack into smart meters to tamper with the readings in order to lower their power bills. In order to generate power and resell it to the grid operator at a profit, customers may set up distributed generation (DG) units founded on renewable energy bases at their sites according to the smart grid concept. In this project [12], Convolutional-neural-networks (CNN), deep-feed-forward-neural-networks (DNN), and recurrent-neural-networks with gated-recurrent-units (RNN-GRU) were examined for their performance in detecting electrical cyberattacks. However, the disappearing or expanding gradient problem might affect RNNs when they are being trained. It might be challenging to learn new things or for the model to become unstable when the gradient values grow or shrink dramatically over time. The capacity of RNNs to recognize long-term relationships in sequences is constrained by this issue. One area where machine learning is heavily used is in the predicting of electric usage using data from smart meters. Classification and clustering techniques must be used to thoroughly analyze the smart meter data in order to forecast peak demand and electric appliance use. A critical phase in the project and expansion of the electric power organization is the prediction of household appliances and high-demand periods. However, due to variations in customer demand and device consumption level demand, an in-depth and comprehensive examination of purchasers' smart meter data is compulsory to recognize key traits and the causes of deviation in both.

This paper [13] focuses on utilizing information from the Irish and Umass repositories to estimate high demand and amounts of electrical appliance use correlate with private consumers' activities. The findings of the customer peak demand forecast are then used to analyses the customer's lifestyle. To anticipate the appliance consumption level and peak demand, the supervised and unsupervised machine learning algorithms CLARA clustering, support vector machine (SVM), and artificial neural network are used. In order to anticipate the average household's usage of electric appliances over the course of a year using smart data collected at 1-minute intervals, mean electric appliance consumption values are produced. Only the average weekly consumption of the combined homes is computed for the customers' peak demand consumption. With 99.6% accuracy, the SVM-based forecasting of consumer energy usage outperforms other studies in the same field of research. The outcome demonstrates that the approaches and algorithms used are being used to their fullest potential. To attain best performance, SVMs require careful tuning of a number of hyper parameters. The act of the model can be strongly impacted by the choice of parameters, comprising the regularization parameter (C), kernel type, and kernel parameters. Choosing the right parameter values frequently calls for knowledge or considerable testing. In the project, feature scaling with the Min Max Scaler is essential for getting the dataset ready for model training and enhancing the prediction model's performance. To avoid any data leaking, the Min Max Scaler is employed independently to the dataset's training and testing sets. The scaler is fitted using the training set, and the minimum and maximum values of each feature are determined. The training and testing sets are then both transformed using these calculated values, maintaining consistency between the two. The project guarantees that the random forest regressor model receives standardized input data by scaling the numeric features using the Min Max Scaler, increasing the model's capability to precisely estimate energy usage. This preprocessing step improves the model's performance by enabling it to recognize the complex patterns and correlations found in the data from smart meters.

3 PROBLEM STATEMENT

The above literature review provides insights into various approaches and techniques used in the domain of electricity consumption forecasting and anomaly detection in the context of smart meter data. However, it also highlights several persistent challenges and areas for improvement in this field. Efficient and accurate forecasting of electricity consumption, as well as the detection of anomalies and non-technical losses in smart meter data, remains a critical concern for electric utility companies, particularly in the era of smart grids. Existing methods have shown limitations in capturing non-linear correlations, ensuring data security and privacy, and handling the multifaceted nature of the problem. Researchers and professionals in the domains of Internet of Things (IoT), big data, and collaborative computing seek more reliable, precise, and secure forecasting and anomaly detection solutions that can not only provide robust results but also consider the unique challenges posed by smart meters. To address these issues, there is a growing need for innovative approaches that leverage advanced technologies such as hybrid machine learning models, deep neural networks, and fog computing architectures [14].

4 PROPOSED HYBRID RANDOM FOREST AND LINEAR REGRESSION METHOD

The flow diagram for the smart meter prediction process starts with the data preprocessing step and it is shown in in Figure 1. In order to enable model training and assessment on unobserved data, the preprocessed dataset is then splitting for training and testing sets. The Random Forest Regressor technique is then used for model training. To properly estimate energy usage, this system builds an ensemble of decision trees and integrates their estimates. Using the training dataset, the research approach is trained to identify patterns and connections between the input characteristics and the target variable. Feature scaling is the following step after the model has been trained. The Min-Max Scaler is used to scale the dataset's numerical features, ensuring that each feature contributes equally to model training. Any bias that may arise from variances in their magnitudes is eliminated by modifying the features to a comparable range. After scaling, a number of measures are used to gauge the model's effectiveness [15]. To assess the precision and accuracy of the prototype's predictions, mean absolute error, mean squared error, and median absolute error are computed. These metrics offer information on the model's efficiency in calculating energy use. The model can then be utilized for prediction after being assessed and found to be satisfactory. The trained model may be given new or undiscovered data points, and it will produce estimates for energy usage based on the patterns it discovered during training. In conclusion, the flow diagram shows how data preparation, model training, feature scaling, performance evaluation, and prediction happen in order. Using the Random Forest Regressor algorithm and data from smart meters, this methodical technique offers precise calculation of energy use.

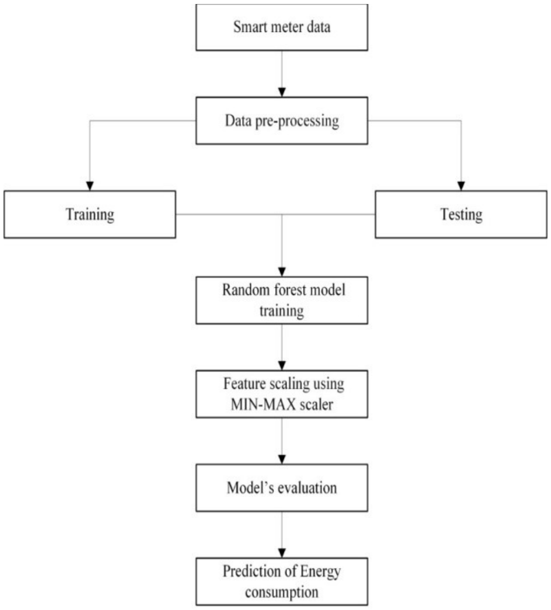
Smart meter data
Data pre-processing
Training
Testing
Random forest model training
Feature scaling using MIN-MAX scaler
Model's evaluation
Prediction of Energy consumption

Figure 1. Workflow of Smart Meter Prediction for Energy Consumption

4.1 Dataset

This dataset has been sourced from Kaggle and encompasses information derived from smart meters installed across a range of buildings, with the primary objective of enabling the prediction of electricity consumption patterns. Each entry in the collection represents a different meter reading instance and has accompanying attributes that offer further context. While the "timestamp" provides the date and time of the reading, the "building_id" specifically identifies the building where the meter is installed. The "meter_reading" gives a specific meter's kWh of power usage at the moment it was recorded. Additional attributes include "primary_use," "square_feet," "year_built," and various weather-related parameters like "air_temperature," "cloud_coverage," and "wind_speed" that describe the building's current environmental conditions at the time of reading. Examples of these attributes include "education," "office," and "primary_use." With the use of these variables, predictive models might be created using this information to project changes in power usage, allowing for well-informed energy management and efficiency strategy decisions across a range of building types and environments. To manage missing values, normalize data, and create pertinent characteristics for efficient model training and precise predictions, preprocessing and analytical procedures may be required [16].

4.2 Data Pre-processing

The scikit-learn package has a preprocessing method called the Label Encoder. Categorical variables are converted into numerical representations using this technique. Due to the requirement of numerical input in many machine learning methods, this is important [17] Each category in a categorical feature is given a distinct integer value by the Label Encoder. In this case, if a feature has the categories "red," "green," and "blue," the Label Encoder will assign the numbers 0, 1, and 2, correspondingly, for each of those groups. The fit and transform techniques are offered by the Label Encoder class in scikit-learn. Based on the training data, the fit technique learns how to map categories to numerical values. This mapping is used by the transform technique to convert the category feature into numerical values [18]. In order to manage missing values and encode categorical variables into numerical representations, the smart meter dataset is imported and preprocessed. This guarantees that the data is in an appropriate format for additional investigation [19]. (1) helps to derive label encoder is as follows, Where the ith row resembles to the alter of class vector,

$$y_i = \left[y_{i1} y_{i2}, \dots, y_{iq}\right]^T \in \mathcal{Y} \tag{1}$$

According to OvO decomposition rule, Y can be transformed into a ternary encoded label matrix which is depicted in (2) and (3)

$$L = [L^1 L^2, \dots, L^q] \in \{-1,0,+1\}^{m\times l} \tag{2}$$

$$L_j \in \{-1,0,+1\}^{m\times l_j} \tag{3}$$

(4) corresponds to the encoded label matrix of the j th class space. Let l_{ia}^j be the factor in ith row and ath column of L_j , its value is resolute as follows:

$$l_{ia}^j = \begin{cases} +1, & if\ y_{ij} = p_a^j \\ -1, & if\ y_{ij} = n_a^j \\ \ \ 0, & otherwise \end{cases} \tag{4}$$

The encoded feature will contain the transformed numerical representation of the categorical feature. These transformed values can then be used as input for further analysis or machine learning models.

4.3 Model Selection

To regulate the amount of regularization used, the Lasso model is instantiated with predefined regularization strength (alpha). The training set of data, which consists of the independent characteristics and the matching target variable for energy consumption, is used to fit the model. The coef_ attribute is used to get the coefficients of the features after fitting the Lasso regression model. To assess the significance of each factor in forecasting energy use, the absolute values of the coefficients are determined. The scikit-learn library's train_test_split function is an essential tool for separating a dataset into training and testing sets. In machine learning projects, it is frequently used to assess model performance on unobserved data and prevent overfitting. In the context of the research the train_test_split function may be used to split the smart meter dataset into two distinct sets: one for training the predictive model and another for assessing its effectiveness. It enables the project to develop a trustworthy measure of the model's accuracy and generalizability by assessing it on unknown data by separating the dataset using train_test_split. The test_size argument, which specifies the percentage of the input data to be utilised for testing, is also sent to this method together with the input data and target variable. In the project, the testing set is set to 20% of the total data, and the remaining 80% is used for training the models [20].

In machine learning projects, it is frequently used to assess model performance on unobserved data and prevent overfitting. The train_test_split function may be employed to split the smart meter dataset into two distinct sets: one for training the predictive model and another for assessing its effectiveness. It enables the project to develop a trustworthy measure of the model's accuracy and generalizability by assessing it on unknown data by separating the dataset using train_test_split. The test_size argument, which specifies the percentage of the input data to be utilized for testing, is also sent to this method together with the input data and target variable.

4.4 Feature Scaling using Min-Max Scaler

By ensuring that all numerical characteristics are scaled similarly, feature scaling keeps the learning process from being dominated by any particular feature. The frequently used Min Max Scaler converts the numerical properties by scaling them to a predetermined range, often between 0 and 1. By putting the characteristics into a similar range by scaling, the linkages between them are maintained. Each numerical characteristic in the smart meter dataset is normalized using the Min Max Scaler, ensuring that the values fall within the appropriate range. When working with data on energy usage, this normalization step is very crucial since it enables the model to consider all characteristics equally and capture the relative value of each information during training. Before training the model, the input data are scaled using the Min-Max Scaler data preparation approach. All of the input characteristics are scaled to the same range using this normalization approach, which is typically between 0 and 1 [21]. All features are scaled using the Min-Max Scaler to the same range, which makes it simpler for the model to assess the relative weights of various features and produce precise predictions. Additionally, by scaling the data, the optimization technique used to train the model can achieve better convergence rates more quickly. The supplied information demonstrates how the min-max approach was used to scale the movement rate observing data to a range of 0 to 1 is represented in (5).

$$X_i^m = \frac{x_i^m - X_{Min}^m}{X_{Max}^m - X_{Min}^n} \tag{5}$$

Where x_i^m is any value of a variable m; X_{Max}^m and X_{Min}^n are the maximum and the minimum values of that variable; $x_{i,scaled}^m$ is the value after scaling. The problem of one feature overwhelming the others due to its wider range of values may be avoided by normalizing the input data using the Min-Max Scaler. If features are not normalized, the model may give the features with greater values an excessive amount of weight, which might lead to subpar model performance. The Min-Max Scaler makes sure that each feature has an equal influence on the model predictions by scaling all features to the same range.

4.5 HRF-LR Technique Employing for Prediction of Energy Consumption

The experiment has shown how well the Random Forest Regressor algorithm predicts energy usage using information from smart meters. It offers knowledge on cost reduction, cost optimization, and sustainable practices. The accuracy and usability of the prediction model in real-world energy consumption situations can be improved by further developments and the research of cutting-edge approaches [22]. The Random Forest Regressor model was trained and evaluated using metrics such as median absolute error, mean squared error, and mean absolute error. The Random Forest Regressor model was trained and evaluated using the following metrics The train score represents the performance of the model on the training data. A score of 0.9803149820302954 indicates that the model has achieved a high level of accuracy in predicting energy consumption based on the features in the training dataset. It may be concluded from this that the idea has successfully mastered the fundamental patterns and connections in the training data. A score of 0.8921697245702427 suggests that the model is generalizing well to new data and is able to make precise calculations on unseen instances. However, the test score is slightly lower than the train score, indicating that there may be some overfitting of the model to the training data. It is important to monitor and optimize the model to achieve a balance between training and test performance. The Random Forest Regressor model was trained using a subset of 5 features from the available dataset. The selection of characteristics is an important stage in creating a predictive model because it identifies the aspects that are most pertinent to predicting energy usage. By utilizing a smaller set of features, the model can focus on the most informative attributes, potentially improving its performance and reducing computational complexity. The random forest regressor architecture is given in Figure 2.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

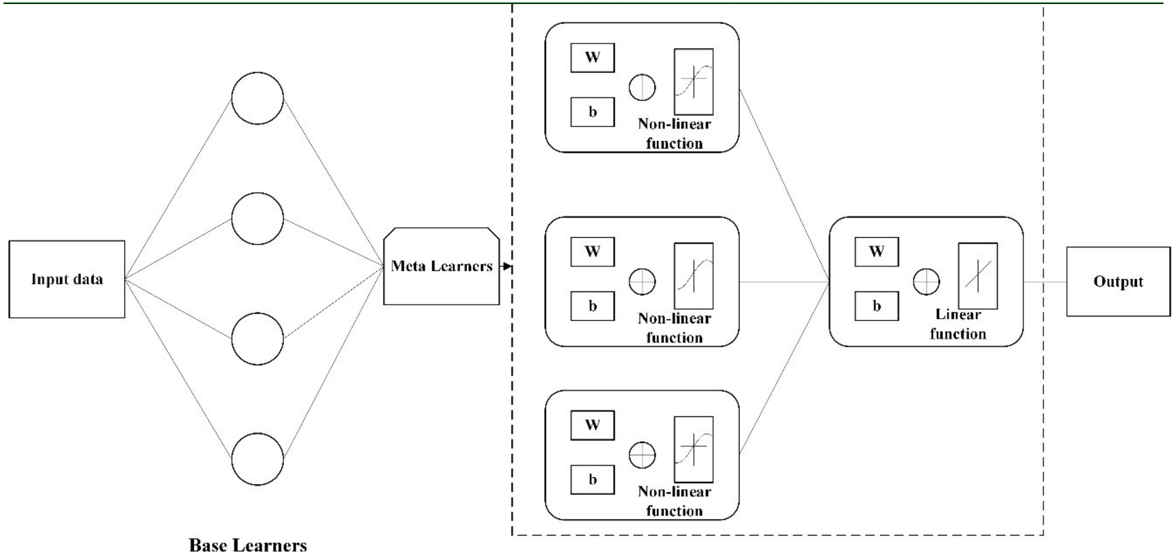

Figure 2. Random Forest Architecture

Random forests, also known as random decision forests, are a method of collective learning that relies on building multiple decision trees during the training process while minimizing a given mistake metric and delivering an average of their forecasts as an output. It is used in the current study as the prediction algorithm. As the foundation of bagging, a sample of size 'n' is chosen at random from the training set, and this sample is subsequently fitted to a regression tree. The same data may reappear in this sample, which is a bootstrap sample and was selected through replacements. Each measurement has a 1/n probability of being selected when n data points with replacements are randomly picked to construct a bootstrap sample. Unpredictable variables that are independent and have the same distributions serve as a representation of this random decision. The bagging approach selects a set of bootstrap examples, applies the CART algorithm to those samples to create a set of prediction trees, and then aggregates the output from each classification. Alongside bagging, the RF algorithm seeks to identify the best chopping path using just the characteristics selected during node separating. By minimizing a cost function, the goal is to discover a suitable combination to chop, and the process is repeated until all the trees are completely grown. RF depends on projections from multiple trees that are combined to provide a consequence that is better than any one tree in the algorithm. Due to the fact that bootstrap combination creates independent trees using many training sets, its key benefit is tolerance to noise. While the mean of multiple independent trees is not susceptible to noise, a poor predictor (regression tree) could. The same goal of preventing overfitting is achieved by choosing a random subset of characteristics during each split. The RF simulations that are within the subject matter of this study were created and evaluated using the Ensembles component from the Python Scikit toolkit. The branches were extended until every leaf was clean or had less than two examples, with the square root of the error being used as a metric to gauge the quality of a split. In the entire forest, 100 trees were automatically subject to consideration.

The HRF-LR model is a powerful approach for energy consumption prediction from smart meter data. This model combines the strengths of both RF and LR to improve prediction accuracy and capture complex relationships within the data. Random Forest Regression is an ensemble learning technique that utilizes multiple decision trees to make predictions. It excels in handling non-linear relationships, capturing interactions between features, and handling outliers and missing values. RF constructs a multitude of decision trees and aggregates their predictions to obtain a robust and accurate prediction (6)[23].

$$y_{pre} = \sum_{i=1}^{t} W_i \times T_i\,(x) \qquad (6)$$

Where, x is the input variable, y_{pre} denotes the predicted value corresponding to the input x, t is the amount of the constructed decision trees. Linear Regression, on the other hand, is a traditional statistical modeling technique that assumes a linear relationship between the input features and the target variable. It is widely used for its interpretability and simplicity. LR estimates the coefficients of the linear equation that best fits the data, allowing for straightforward interpretation of feature importance and the direction of their impact on the target variable [24]. The general form of a LR model is as shown in (7)

$$\hat{y} = x_0 + \sum_{i=1}^{n} x_i C_i \qquad (7)$$

Where $\hat{y}$ is the model output, $C_i{}'s$ are the independent input variables to the model, and $x_0, x_1, x_2, \dots, x_m$ are partial regression coefficients. The hybrid equation for the Random Forest Regression and Linear Regression model can be represented as follows (8):

$$y\_pred = w1 * RF(x) + w2 * LR(x) \qquad (8)$$

Where, y_pred is the predicted energy consumption value. $RF(x)$ indicates the prediction from the Random Forest Regression model using input features x. $LR(x)$ indicates the prediction from the Linear Regression model using input features x. $w1$ and $w2$ are the weights assigned to the predictions of the Random Forest Regression and Linear Regression models, respectively. These weights determine the contribution of each model to the final prediction. The weights $w1$ and $w2$ can be determined based on the performance of each model on the training data or through expert knowledge. The weights can be adjusted to give more importance to one model over the other, depending on their respective strengths and weaknesses.

5 RESULTS AND DISCUSSION

The result and discussion part for the Random Forest Regressor-based energy consumption prediction model provides a thorough evaluation of the model's performance as well as insightful conclusions. Metrics like mean absolute error, mean squared error, and median absolute error are used to estimate the model's presentation and provide quantifiable measurements of how well it predicts energy usage. The talk also looks at the Random Forest Regressor algorithm's feature significance rating in order to recognize significant characteristics and comprehend how they affect energy usage. The findings are explained in terms of energy consumption patterns, allowing for a fuller comprehension of the variables affecting energy use. Additionally, the comparison with benchmark models or earlier research confirms the viability of the selected strategy and accentuates the project's contributions. Limitations and future directions are also discussed, offering information on prospective advancements and potential routes for study. The project's practical implications for energy management, including resource allocation optimization and the promotion of sustainable practices, are highlighted in the outcome and discussion section. The MAE value was found to be 70.79465688782157. This metric measures the average absolute difference between the predicted and actual energy consumption values. A lower MAE indicates that the model's predictions are, on average, closer to the true values. In this project, the MAE suggests that the model has a moderate level of accuracy in predicting energy consumption. The MSE value was calculated as 19648.03629805024. MSE represents the proportional variation among the projected and actual values for power usage. It bounces advanced weightage to larger errors. In this project, the MSE value provides an understanding of the magnitude of the errors, with higher values indicating larger discrepancies between predictions and actual values. The MedAE value was determined to be 30.456483333333836. MedAE is a robust measure of error that represents the median absolute difference between the predicted and actual energy consumption values. It is fewer complex to outliers related to the MAE. A lower MedAE indicates that the model's predictions are more accurate and less affected by extreme values. These evaluation metrics provide insights into the performance and accuracy of the Random Forest Regressor model in predicting energy consumption. While the MAE and MedAE values indicate a moderate level of accuracy, the MSE value suggests that there are significant errors in some predictions. Further analysis and optimization of the model can be done to improve its predictive capabilities and reduce the errors.

A statistical breakdown of many building-related topics, including energy use, is shown in Table I. Building_id, cloud_coverage, meter_reading, year_built, air_temperature, dew_temperature, square_feet, precip_depth_1_hr, sea_level_pressure, wind_direction, and wind_speed is just a few of the columns that are present. Here, the descriptive statistics are provided for each column, including the count (number of data points), mean (average value), standard deviation (measure of data dispersion), 25th percentile, median (50th percentile), 75th percentile, minimum value and maximum value.

Table 1. Statistical Information about Various Aspects Related to Buildings and their Energy Consumption

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

	count	Mean	Std	Min	25%	50%	75%	Max
building_id	698675.0	50.061555	29.345241	0.0	24.0	50.0000	76.000	100.0
meter_reading	698675.0	231.255069	382.396065	0.0	0.0	71.2593	302.715	4521.0
square_feet	698675.0	88803.210646	109440.853130	283.0	24456.0	53130.0000	103286.000	487433.0
year_built	698675.0	1995.637087	14.383392	1968.0	1985.0	2001.0000	2007.000	2016.0
air_temperature	698452.0	22.841813	6.030032	1.7	18.9	23.9000	26.700	36.1
cloud_coverage	394159.0	3.0433406	2.119795	0.0	2.0	2.0000	4.000	9.0
dew_temperature	698452.0	16.824991	6.512371	-9.4	13.3	18.3000	22.200	25.6
precip_depth_1_hr	698591.0	1.370126	12.870762	-1.0	0.0	0.0000	0.000	343.0
sea_level_pressure	691953.0	1017.985766	4.035453	992.0	1015.5	1018.0000	1020.600	1030.2
wind_direction	678753.0	156.437364	118.367931	0.0	60.0	140.0000	260.000	360.0
wind_speed	698675.0	3.376827	2.156156	0.0	2.1	3.1000	4.600	15.4

5.1 Correlation Heat Map

The correlation coefficients, which can vary from -1 to 1, are shown in the annotations inside each cell of the heatmap. A correlation coefficient of -1 indicates a perfectly negative correlation, a correlation coefficient of 0 specifies no association at all, and a correlation value of 1 shows the complete positive relationship. Finding variables that are highly associated with one another is made simpler by visualizing the correlation matrix in a heatmap. By offering insights into probable patterns and interactions within the dataset, this aids in comprehending the connections and dependencies between variables.

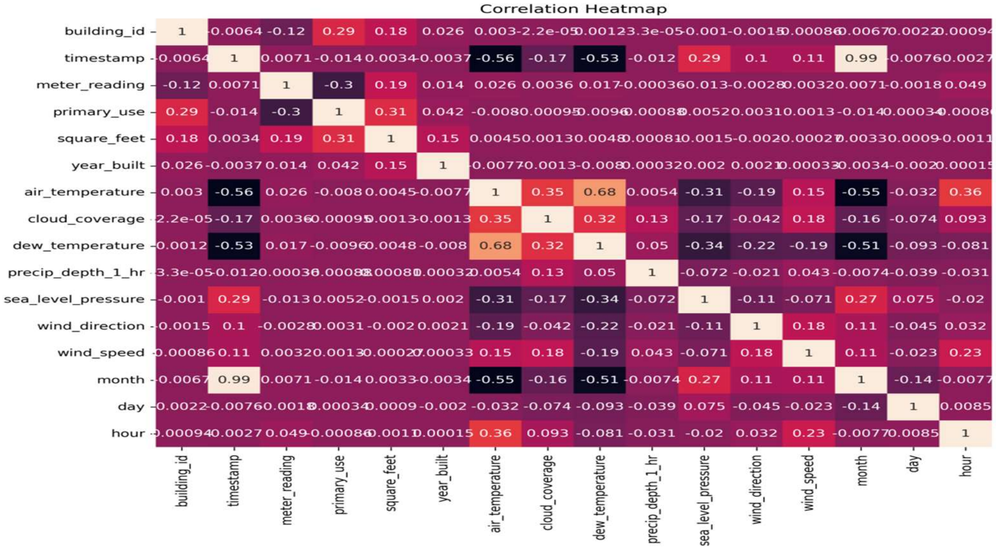

Figure 3. Correlation between Variables

A visual depiction of the correlation between variables in the dataset is provided by the produced heatmap in Figure 3. Warmer colours are used to represent positive correlations whereas colder colours are used to display negative correlations. The correlation values are indicated as annotations on the heatmap cells, providing information about the connections between various variables. Strong connections between variables are highlighted by the heatmap, which is helpful for understanding relationships and seeing potential patterns or trends in the data.

5.2 Kernal Density Estimate Plot

A Kernal density estimation plot for a particular variable in the sample is represented by each subplot in Figure 4. A smoothed approximation of the probability density function of a continuous variable is provided by these figures. By calculating the underlying probability density, this figure illustrates the distribution of each variable. It displays the distribution's form and draws attention to the data's peaks, troughs, and modes [25].

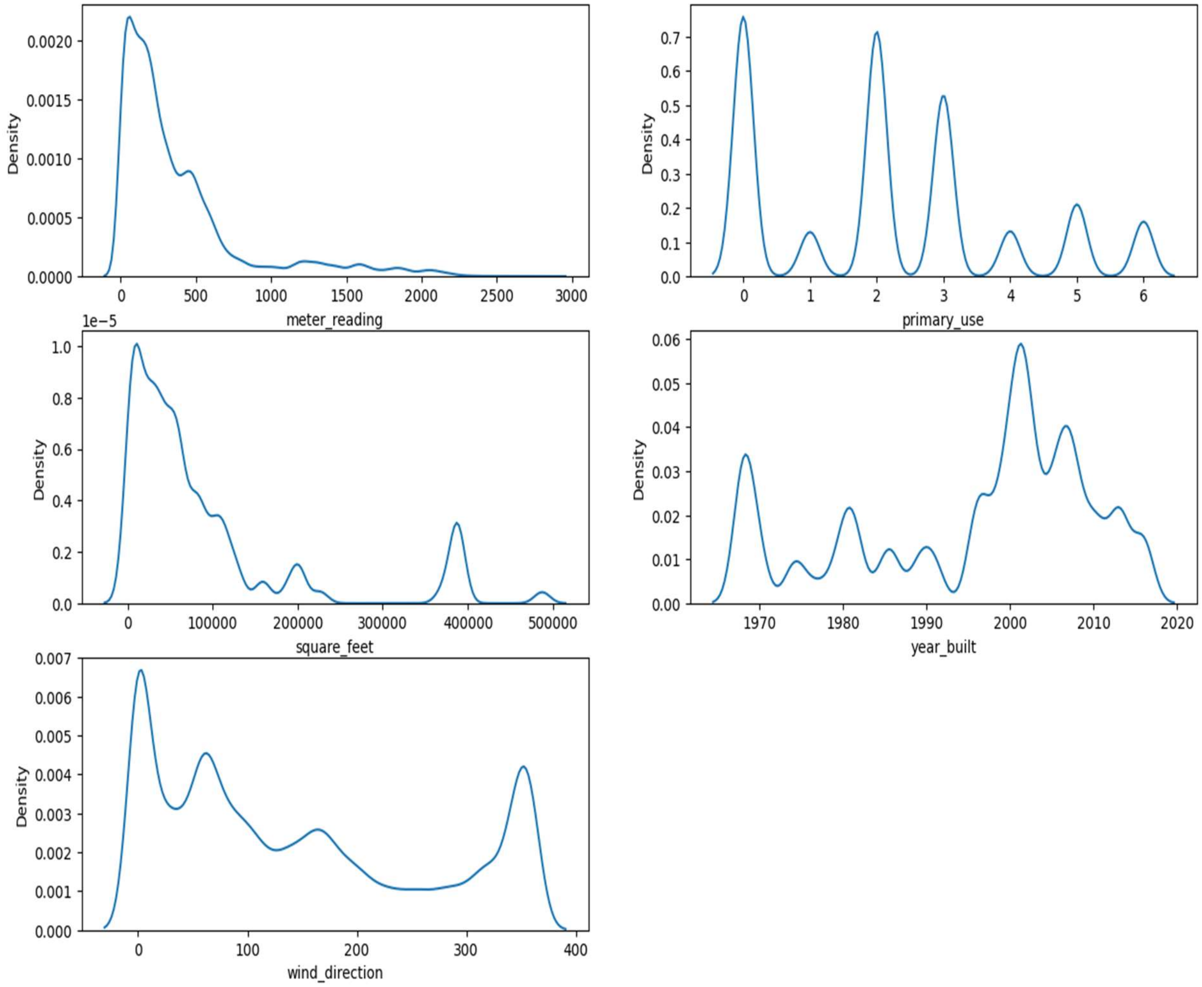

Figure 4. Kernal Density Estimation Plot for a Specific Variable

The connected variable's numbers are shown on the x-axis, while the predicted values for density are shown on the y-axis. The picture enables a visual evaluation of the distribution patterns and changes within the dataset by showing the density for each variable. It aids in comprehending the distribution, core patterns, and potential outliers of the data. One may learn more about the distribution trends, concentrations, and fluctuations in the data for any particular variable by looking at the KDE plots. A visual grasp of the distributional properties of the data is made possible by the KDE plots, which offer a smoothed picture of the probability density function. In comparison to a histogram, KDE may produce a fewer packed plot that is simpler to understand, particularly when presenting several variations. However, distortions may occur if the basic distribution is restricted or irregular. The usage of appropriate smoothing parameters determines the depiction's quality, just as a histogram [26].

5.3 Count Plot

Figure 5 demonstrates that the 'year_built' variable's x-axis indicates its distinct values. An individual year is represented by each tick on the x-axis. The count or frequency of occurrences for each year is shown on the y-axis. It shows how many occurrences of the same 'year_built' value there are in the dataset. Every distinct "year_built" value is represented by a bar in the plot, and the height of each bar reflects the number of occurrences for that particular year. The bar is raised if more structures or properties were constructed in a given year. The count plot may be used to determine how properties are distributed across time. It aids in comprehending the frequency of various building eras and might offer useful details on the vintage and age of the structures in the dataset.

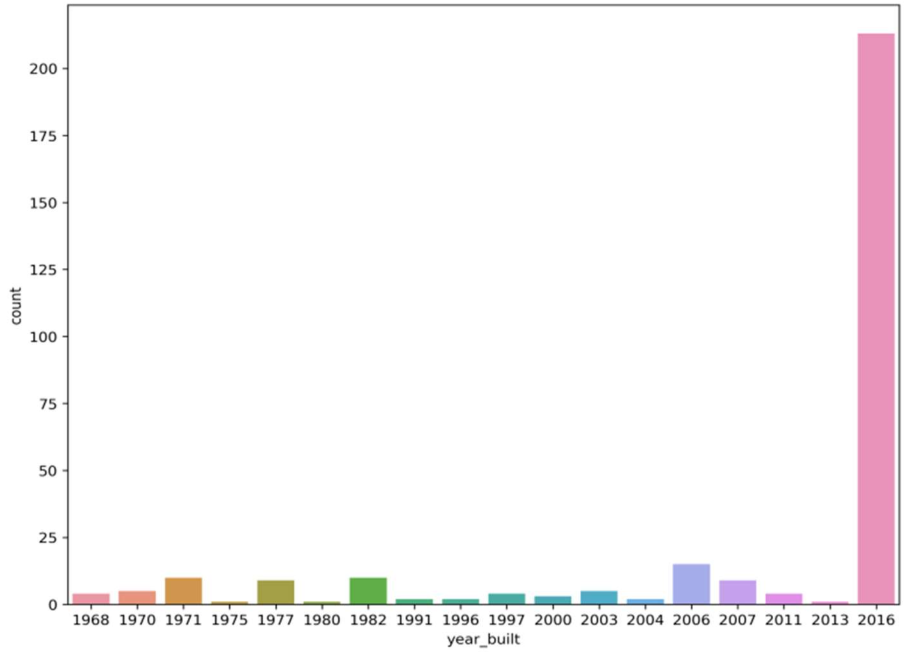

Figure 5. Count of Occurrences for Specific Year

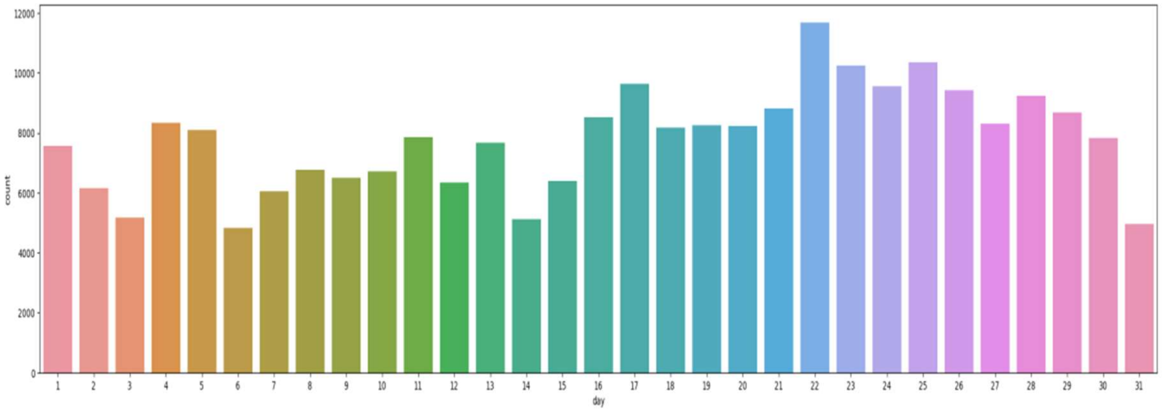

Figure 6. Frequency of Events Corresponds to Day

The width and height of the resulting plot in inches are determined by the figure size, which in Figure 6 is set to (30, 6). Each distinct value of the 'day' variable's frequency or count of occurrences is shown on the countplot. The unique values of the 'day' variable, which normally corresponds to the days of the month, are displayed on the x-axis. The count or regularity of occurrences for each day is shown on the y-axis. The height of each bar in the plot, which signifies a single day, corresponds to how frequently that day appears in the dataset. The countplot aids in analyzing the frequency of activities or observations associated with every day, offering important insights into temporal trends or patterns in the dataset.

5.4 Scatter Plot

The scatter plot shows how the indices (on the x-axis) and associated values (on the y-axis) relate to one another. The data point indices are shown on the x-axis. An individual index of the data is associated with each point on the x-axis. The values being plotted, whether they be actual values or expected values are represented by the y-axis. Blue scatter spots on a graph represents the actual numbers. The actual value of a data point from the test set is represented by each blue point. Red scatter dots represent the expected values on the graph. The predicted value of a data point produced from the prediction model is represented by each red point.

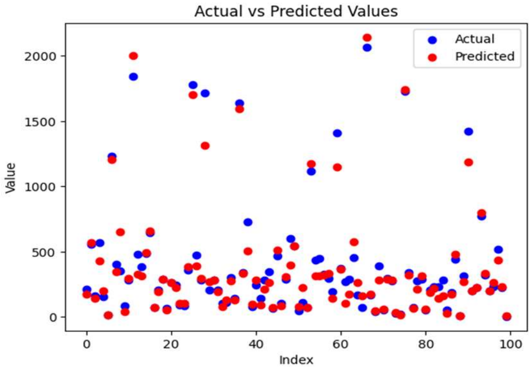

Figure 7. Scatter Plot to Identify the Relationship between the Indices

The 'Index' x-axis in Figure 7 shows the data points' indices. The values being plotted are indicated by the word "Value" on the y-axis. The plot's title, "Actual vs Predicted Values," sums well the goal of the visualization. To distinguish between the actual values (blue) and the expected values (red), a legend has been provided. The presentation and accuracy of the prediction perfect may be evaluated graphically by comparing the actual and forecast morals in the scatter plot. It aids in determining whether there are any variations or conflicts between the model forecasts and the actual data points.

5.5 Line Plot

The discrepancy among the actual values and the expected values is shown by the line plot. The data point indices are shown on the x-axis. An individual index of the data is associated with each point on the x-axis. The difference between each data point's actual value and its forecasted value is shown on the y-axis. Calculated and shown as a line is the discrepancy between the measured values and the projected values. The difference's size and direction are shown by the line. When the anticipated values are greater than the actual values, as shown by positive values, the predicted values are lower than the actual values, as indicated by negative values.

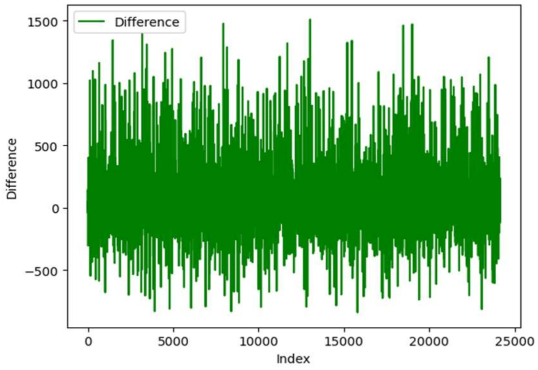

Figure 8. Determined Differences between Actual Values and Expected Values are Represented as Lines

The indices of the data points are represented by the 'Index' x-axis in Figure 8. The 'Difference' label on the y-axis denotes the variance between the actual and expected values. The line is marked as "Difference" by the addition of a legend. The line plot makes it easier to see any flaws or differences between the projected and actual numbers. The size and direction of the prediction errors are shown by positive or negative departures from the zero line. This graph aids in comprehending the prediction model's overall performance and locating any predictable biases or discrepancies in the forecasts.

The line plot in Figure 9 shows the correlation between the indices' (x-axis) values and corresponding values. The data point indices are shown on the x-axis. An individual index of the data is associated with each point on the x-axis. The values being plotted, whether they be actual values or expected values are represented by the y-axis. A blue line represents the actual numbers on the graph. The first 10 data points from the test set are represented by the blue line, which shows their actual values. A red line represents the expected values on the graph. For the first 10 data points acquired from the prediction model, the red line indicates the predicted values.

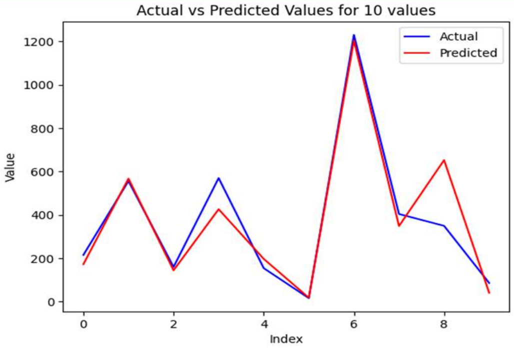

Figure 9. Line Plot Visualizes the Relationship Between the Indices

The indices of the data points are displayed on the x-axis, which is designated as "Index." The values being plotted are indicated by the word "Value" on the y-axis. Actual versus Predicted Values for 10 Values is how the plot's title is chosen to describe the visualization's goal. Figure 9 shows a legend that separates the actual values (blue) from the anticipated values (red).

The actual values are shown as a blue line in Figure 10. The first 100 data points from the test set are shown by the blue line, which depicts their actual values. A red line represents the expected values on the graph. The first 100 data points acquired from the prediction model are represented by the red line, which shows the predicted values. The indices of the data points are displayed on the x-axis, which is designated as "Index." The values being plotted are indicated by the word "Value" on the y-axis. The plot's title, "Actual vs Predicted Values for 100 values," explains what the visualization is meant to show. To distinguish between the actual values (blue) and the expected values (red), a legend has been provided.

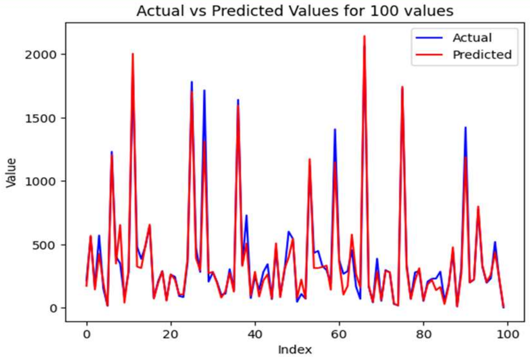

Figure 10. Actual vs Predicted Values for 100 values

5.6 Heatmap

The link between the indices (y-axis), the actual values (x-axis), and the accompanying residuals are shown visually via the heatmap. Actual values from the dataset are shown on the x-axis. In Figure 11, the indices of the data points are shown on the y-axis. The average residual value for a particular combination of an actual value and an index is shown in each heatmap cell. The discrepancy between the actual and projected values is used to determine the residuals. The residuals quantify the discrepancy between expected and actual values. Positive residuals indicate that the anticipated values are higher than the actual values, while negative residuals indicate that the values that are anticipated are less than the actual values.

The heatmap in Figure 11 shows the residuals' magnitude in a range of colors from cool (blue) to warm (red). The average residual value for each combination of an actual value and an index is shown in the annotated heatmap cells. For easier reading, the format of the annotations is adjusted to two decimal places (".2f"). The 'Actual' x-axis represents the actual values from the dataset and is marked as such. The data points' indices are shown on the y-axis, which is designated as "Index." The goal of the visualization is described by the plot's title, "Actual vs Predicted (Residuals)". The heatmap makes it easier to see the distribution and trends of the residuals. By displaying the differences between the anticipated and actual values across various real values and indices, it aids in analysing the prediction model's accuracy.

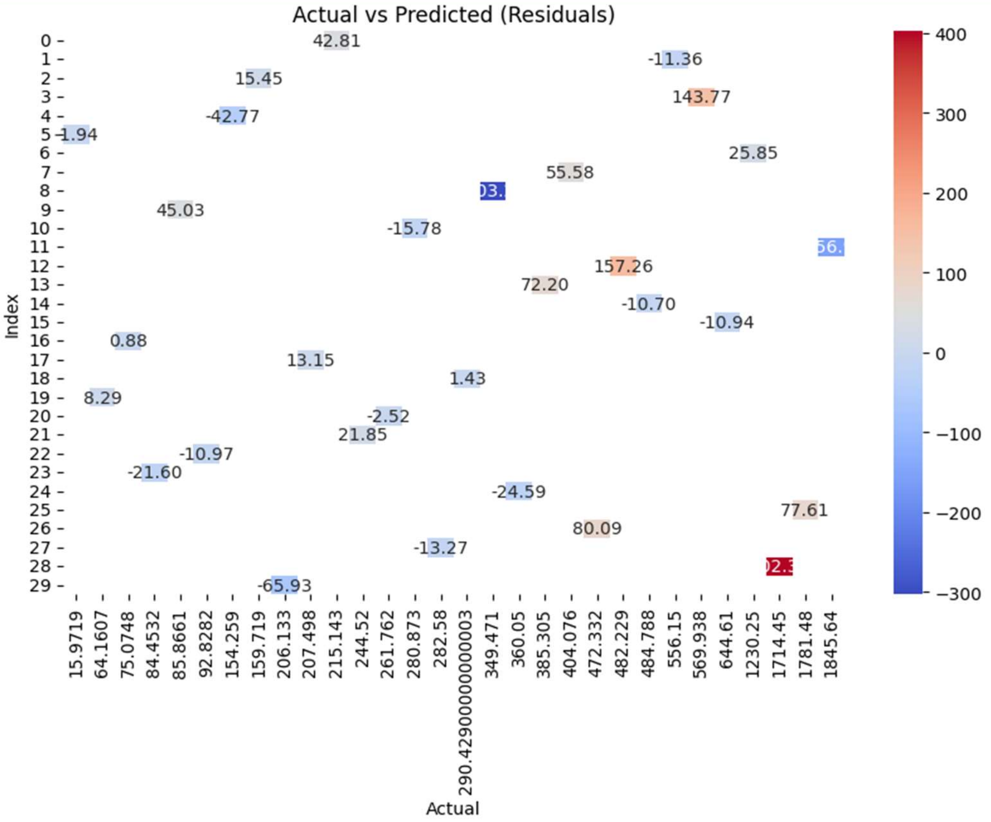

Figure 11. Heatmap

Table 2. Error Rate Comparison with Existing Methods

Models	MAE (%)
SARIMA [27]	72.15
SVR + LR [27]	75.63
Proposed HRF-LR	70.79465688782157

The Table 2 provides an evaluation of different models based on two performance metrics. Metrics are commonly used to assess the accuracy of predictive models.

In Figure 12 error rate comparison with existing methods is depicted, and the proposed method achieves lower error when compared to existing methods.

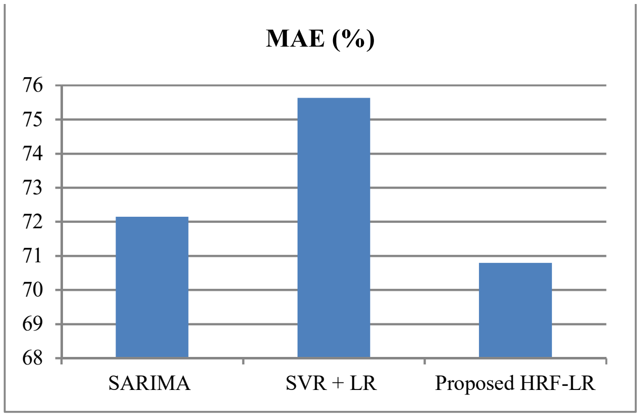

Figure 12. Error Rate Comparison with Existing Methods

6 CONCLUSION AND FUTURE WORK

The project's goal was to forecast energy usage using data from smart meters and the Random Forest Regressor algorithm. To get the data ready for modeling, preprocessing methods like Label Encoding and managing missing values using Simple Imputer were used. The train_test_split technique was used to separate the dataset into training and testing sets for the assessment of the model. To guarantee that each feature contributes equally during model training, feature scaling was done using the Min Max Scaler. The Random Forest Regressor model was trained and evaluated using metrics such as mean absolute error, mean squared error, and median absolute error. Heatmaps, scatter plots, line plots, count plots, and other visualization techniques were used to analyses the data, assess model performance, and comprehend the link between variables. The findings demonstrated that the Random Forest Regressor model was able to estimate energy usage accurately based on the provided characteristics. Throughout the investigation, strict control methods were used. The use of cross-validation to evaluate model performance as well as data pretreatment methods to address outliers and missing data are some of these controls. These tests ensured that the study findings appropriately represent the model's predicting abilities and those they remain reliable and consistent. The main features impacting the forecast of energy usage were identified by a study of feature significance. Comparing the performance of other machine learning algorithms against that of the Random Forest Regressor might be useful. Support vector machines, gradient boosting, and neural networks are a few examples of algorithms that might provide an alternate modeling strategy and perhaps increase prediction accuracy. The model's forecasting skills can be improved by including outside data sources, such as weather data. A more complete picture of the link between energy use and environmental elements may be obtained by including the information that weather conditions have on energy use. Incorporating more advanced machine learning methods, including deep learning and neural networks, might help increase the forecasting of power usage utilizing hybrid models in the future. This would improve the system's ability to recognize complex connections and patterns in data from smart metres, perhaps resulting in even more precise forecasts.

REFERENCES:

[1] F. Gong *et al.*, “Integrated scheduling of hot rolling production planning and power demand response considering order constraints and TOU price,” *IET Gener. Transm. Distrib.*, vol. 16, no. 14, pp. 2840–2851, Jul. 2022, doi: 10.1049/gtd2.12412.

[2] E. U. Haq, X. Lyu, Y. Jia, M. Hua, and F. Ahmad, “Forecasting household electric appliances consumption and peak demand based on hybrid machine learning approach,” *Energy Rep.*, vol. 6, pp. 1099–1105, Dec. 2020, doi: 10.1016/j.egyr.2020.11.071.

[3] S. Jung, J. Moon, S. Park, S. Rho, S. W. Baik, and E. Hwang, “Bagging Ensemble of Multilayer Perceptrons for Missing Electricity Consumption Data Imputation,” *Sensors*, vol. 20, no. 6, p. 1772, Mar. 2020, doi: 10.3390/s20061772.

[4] B. W. Billings and K. M. Powell, “Grid-responsive smart manufacturing: A perspective for an interconnected energy future in the industrial sector,” *AIChE J.*, vol. 68, no. 12, Dec. 2022, doi: 10.1002/aic.17920.

[5] S. S. Kholerdi and A. Ghasemi-Marzbali, “Interactive Time-of-use demand response for industrial electricity customers: A case study,” *Util. Policy*, vol. 70, p. 101192, Jun. 2021, doi: 10.1016/j.jup.2021.101192.

[6] S. Hussain, M. W. Mustafa, T. A. Jumani, S. K. Baloch, and M. S. Saeed, “A novel unsupervised feature-based approach for electricity theft detection using robust PCA and outlier removal clustering algorithm,” *Int. Trans. Electr. Energy Syst.*, vol. 30, no. 11, Nov. 2020, doi: 10.1002/2050-7038.12572.

[7] A. Alsirhani, M. Mujib Alshahrani, A. Abukwaik, A. I. Taloba, R. M. Abd El-Aziz, and M. Salem, “A novel approach to predicting the stability of the smart grid utilizing MLP-ELM technique,” *Alex. Eng. J.*, vol. 74, pp. 495–508, Jul. 2023, doi: 10.1016/j.aej.2023.05.063.

[8] S. Hussain, M. W. Mustafa, K. H. A. Al-Shqeerat, F. Saeed, and B. A. S. Al-rimy, “A Novel Feature-Engineered–NGBoost Machine-Learning Framework for Fraud Detection in Electric Power Consumption Data,” *Sensors*, vol. 21, no. 24, p. 8423, Dec. 2021, doi: 10.3390/s21248423.

[9] R. Jaiswal, A. Chakravorty, and C. Rong, “Distributed Fog Computing Architecture for Real-Time Anomaly Detection in Smart Meter Data,” in *2020 IEEE Sixth International Conference on Big Data Computing Service and Applications (BigDataService)*, Oxford, United Kingdom: IEEE, Aug. 2020, pp. 1–8. doi: 10.1109/BigDataService49289.2020.00009.

[10] M. Sajjad *et al.*, “A Novel CNN-GRU-Based Hybrid Approach for Short-Term Residential Load Forecasting,” *IEEE Access*, vol. 8, pp. 143759–143768, 2020, doi: 10.1109/ACCESS.2020.3009537.

[11] M.-M. Buzau, J. Tejedor-Aguilera, P. Cruz-Romero, and A. Gomez-Exposito, "Hybrid Deep Neural Networks for Detection of Non-Technical Losses in Electricity Smart Meters," *IEEE Trans. Power Syst.*, vol. 35, no. 2, pp. 1254–1263, Mar. 2020, doi: 10.1109/TPWRS.2019.2943115.

[12] D. B. S. Reddy, S. Shresta, S. Sathhvika, and P. L. M. Shreya, "Detection of Electricity Theft Cyber-Attacks In Renewable Distributed Generation for Future IoT-based Smart Electric Meters," *Turk. J. Comput. Math. Educ. TURCOMAT*, vol. 14, no. 2, Art. no. 2, Mar. 2023.

[13] F. Z. Abera and V. Khedkar, "Machine Learning Approach Electric Appliance Consumption and Peak Demand Forecasting of Residential Customers Using Smart Meter Data," *Wirel. Pers. Commun.*, vol. 111, no. 1, pp. 65–82, Mar. 2020, doi: 10.1007/s11277-019-06845-6.

[14] J. D. Fernández, S. P. Menci, C. M. Lee, A. Rieger, and G. Fridgen, "Privacy-preserving federated learning for residential short-term load forecasting," *Appl. Energy*, vol. 326, p. 119915, Nov. 2022, doi: 10.1016/j.apenergy.2022.119915.

[15] A.-L. Klingler and F. Schuhmacher, "Residential photovoltaic self-consumption: Identifying representative household groups based on a cluster analysis of hourly smart-meter data," *Energy Effic.*, vol. 11, no. 7, pp. 1689–1701, Oct. 2018, doi: 10.1007/s12053-017-9554-z.

[16] "Predict daily consumption: ARIMA and mixed model." Accessed: Aug. 17, 2023. [Online]. Available: https://kaggle.com/code/marikali/predict-daily-consumption-arima-and-mixed-model

[17] D. Shah, Z. Y. Xue, and T. M. Aamodt, "Label Encoding for Regression Networks," 2022, doi: 10.48550/ARXIV.2212.01927.

[18] E. Bisong, "Introduction to Scikit-learn," in *Building Machine Learning and Deep Learning Models on Google Cloud Platform*, Berkeley, CA: Apress, 2019, pp. 215–229. doi: 10.1007/978-1-4842-4470-8_18.

[19] B.-B. Jia and M.-L. Zhang, "Multi-Dimensional Classification via Decomposed Label Encoding," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 2, pp. 1844–1856, Feb. 2023, doi: 10.1109/TKDE.2021.3100436.

[20] Z. Qu, H. Liu, Z. Wang, J. Xu, P. Zhang, and H. Zeng, "A combined genetic optimization with AdaBoost ensemble model for anomaly detection in buildings electricity consumption," *Energy Build.*, vol. 248, p. 111193, Oct. 2021, doi: 10.1016/j.enbuild.2021.111193.

[21] C. Zhang, L. Ma, and W. Liu, "A Machine Learning Approach for Prediction of the Quantity of Mine Waste Rock Drainage in Areas with Spring Freshet," *Minerals*, vol. 13, no. 3, p. 376, Mar. 2023, doi: 10.3390/min13030376.

[22] C. A. C. Montanez and W. Hurst, "A Machine Learning Approach for Detecting Unemployment Using the Smart Metering Infrastructure," *IEEE Access*, vol. 8, pp. 22525–22536, 2020, doi: 10.1109/ACCESS.2020.2969468.

[23] B. He, S. H. Lai, A. S. Mohammed, M. M. S. Sabri, and D. V. Ulrikh, "Estimation of Blast-Induced Peak Particle Velocity through the Improved Weighted Random Forest Technique," *Appl. Sci.*, vol. 12, no. 10, p. 5019, May 2022, doi: 10.3390/app12105019.

[24] V. Sharma, S. Ghosh, S. Dey, and S. Singh, "Modelling PM2.5 for Data-Scarce Zone of Northwestern India using Multi Linear Regression and Random Forest Approaches," *Ann. GIS*, pp. 1–13, Feb. 2023, doi: 10.1080/19475683.2023.2183523.

[25] P. Vermeesch, "On the visualisation of detrital age distributions," *Chem. Geol.*, vol. 312–313, pp. 190–194, Jun. 2012, doi: 10.1016/j.chemgeo.2012.04.021.

[26] N. Jiwani, K. Gupta, and P. Whig, "Novel HealthCare Framework for Cardiac Arrest With the Application of AI Using ANN," in *2021 5th International Conference on Information Systems and Computer Networks (ISCON)*, Mathura, India: IEEE, Oct. 2021, pp. 1–5. doi: 10.1109/ISCON52037.2021.9702493.

[27] D. M. F. Izidio, P. S. G. De Mattos Neto, L. Barbosa, J. F. L. De Oliveira, M. H. D. N. Marinho, and G. F. Rissi, "Evolutionary Hybrid System for Energy Consumption Forecasting for Smart Meters," *Energies*, vol. 14, no. 7, p. 1794, Mar. 2021, doi: 10.3390/en14071794.