energies

MDPI

Article

A Comparative Analysis of Hyperparameter Tuned Stochastic Short Term Load Forecasting for Power System Operator

B. V. Surya Vardhan [1], Mohan Khedkar [1], Ishan Srivastava [1], Prajwal Thakre [1] and Neeraj Dhanraj Bokde [2,3,*]

[1] Department of Electrical Engineering, Visvesvaraya National Institute of Technology, Nagpur 440010, India
[2] Center for Quantitative Genetics and Genomics, Aarhus University, 8000 Aarhus, Denmark
[3] iCLIMATE Aarhus University Interdisciplinary Centre for Climate Change, Foulum, 8830 Tjele, Denmark
[*] Correspondence: neerajdhanraj@qgg.au.dk

Abstract: Intermittency in the grid creates operational issues for power system operators (PSO). One such intermittent parameter is load. Accurate prediction of the load is the key to proper planning of the power system. This paper uses regression analyses for short-term load forecasting (STLF). Assumed load data are first analyzed and outliers are identified and treated. The cleaned data are fed to regression methods involving Linear Regression, Decision Trees (DT), Support Vector Machine (SVM), Ensemble, Gaussian Process Regression (GPR), and Neural Networks. The best method is identified based on statistical analyses using parameters such as Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Square Error (MSE), R^2, and Prediction Speed. The best method is further optimized with the objective of reducing MSE by tuning hyperparameters using Bayesian Optimization, Grid Search, and Random Search. The algorithms are implemented in Python and Matlab Platforms. It is observed that the best methods obtained for regression analysis and hyperparameter tuning for an assumed data set are Decision Trees and Grid Search, respectively. It is also observed that, due to hyperparameter tuning, the MSE is reduced by 12.98%.

Keywords: short term load forecasting; machine learning; Bayesian optimization; grid search; random search

check for updates

Citation: Vardhan, B.V.S.; Khedkar, M.; Srivastava, I.; Thakre, P.; Bokde, N.D. A Comparative Analysis of Hyperparameter Tuned Stochastic Short Term Load Forecasting for Power System Operator. *Energies* **2023**, *16*, 1243. https://doi.org/10.3390/en16031243

Academic Editor: GM Shafiullah

Received: 24 December 2022
Revised: 19 January 2023
Accepted: 22 January 2023
Published: 23 January 2023

1. Introduction

The continuous modernization of societies has increased electricity demand. Penetration of renewable energy in the grid has led to improper scheduling of power [1,2]. One of the reasons for improper scheduling of power is inaccurate load forecasting. A detailed analysis of variation in peak electricity demand is shown in Figure 1 [3]. As can be seen from Figure 1, India's peak power consumption is always rising, and this is the case for all emerging countries. The ability to meet peak demand has also increased, which can be accounted for by accurate load forecasting and efficient power management [4,5]. The increase in demand has forced the power system operators (PSO) to move away from conventional methods to predict load. The emergence of machine learning techniques has helped power system operators to predict load using various stochastic approaches. In some recently reported work, machine learning methodology is used to solve outage management problems in power distribution systems [6].

Load Forecasting is usually classified based on the duration of its prediction viz. Short Term, Medium Term and Long term. The ranges for these prediction horizons are shown in Table 1.

The most relevant forecasting for the day ahead and real-time power markets is STLF, and this paper focuses on the implementation of STLF using stochastic approaches.

Various researchers have worked on the issue of short-term load forecasting (STLF) using regression analysis. The implementation of STLF using Linear Regression is given in [7]. Although Linear Regression methods are known for their simplicity, the accuracy of Linear Regression depends upon the linearity factor between input and output [8].

The solution to STLF obtained using a Support Vector Machine (SVM) is shown in [9,10]. Non Linearity aspects of the Input and Output of the data can be easily solved using SVM; however, sometimes SVM can be ineffective with large data sets [11]. The approach to STLF using Decision Trees (DT) is demonstrated in [12]. When the data set of the system is large, DTs can produce effective solutions. The disadvantage of using DTs includes false prediction through over-fitting [13]. Hidden relations between data points can be effectively calculated using Ensemble [14,15]. Hardware integration of Ensemble methods is often costly due to its complexities [16,17]. The issue of uncertainties can be dealt with using Gaussian Process Regression (GPR) effectively [18], but lacks a mechanism to find interrelation between parameters [19]. Neural Networks can provide accurate solutions to those problems, where data are changing continuously and can have good communication with Data Base Management Systems (DBMS) [20]. A comprehensive analysis of load forecasting methods using predictive models is given in [21].

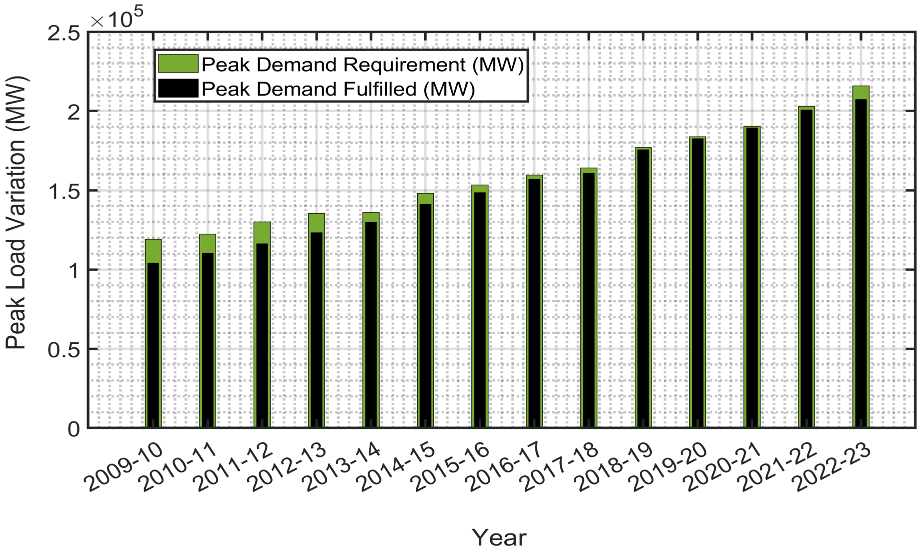


Figure 1. Yearly Peak Demand Requirement versus Peak Demand fulfilled of India in MW.

Table 1. Ranges of different prediction horizons for load forecasting.

Range	Duration
Short term load forecasting (STLF)	1 hour to 1 week
Medium Term Load Forecasting (MTLF)	1 week to 1 year
Long Term Load Forecasting (LTLF)	1 year to 20 years

Hyperparameter tuning plays a crucial role in optimizing the operating parameters of machine learning methodologies. This paper uses three different techniques for hyperparameter tuning using Grid Search, Random Search, and Bayesian Optimization. Grid Search optimization is usually used in medical applications. In this paper, it is utilized in STLF applications. The usage of Grid Search is broadly explained in [22]. The hyperparameter tuning for SVM is reported in [23]. The advantages of Grid Search methodology include its exhaustive nature of search but computational time increases exhaustively when a large number of hyperparameters are tuned [24]. Various studies observed Random Search as an important alternative to Grid Search. In Random Search, arbitrary combinations are used to find the optimal solution, hence to a certain extent it becomes independent of the number of hyperparameters [25]. One of the biggest limitations of Random Search algorithms is their high dependence on their randomness. Hence, different parameters produce different results [26]. The third method used for hyperparameter tuning in this paper is Bayesian Optimization. Bayesian Optimization is based on Bayes theorem, which is based on conditional probability. The goal of Bayesian reasoning is to become “less incorrect" with more data, which these systems do by continuously updating the surrogate probability model after each evaluation of the objective function. Bayesian reasoning is a form of inductive reasoning [27]. Bayesian Optimization is designed in such a way that its current results depend upon past evaluation results, hence sufficient data are required to produce quality results [28]. A detailed analysis of all the hyperparameter tuning methods can be accessed in [29].

Instead of directly forecasting load, load is first classified using classifiers and the classified load is then fed to the regression method to predict load. This advantage of this approach is that, with classifiers, the load band is identified and regression methods are applied to that particular band instead of all the data. This improved methodology is a novel aspect of the manuscript. The following aspects have been discussed in this paper:

- Data are completely analyzed and outliers are detected using the Inter Quartile Range (IQR) technique. After detection, outliers are removed since the data set is large;
- The cleaned data are then fed to various machine learning techniques including SVM, Ensemble, Decision Trees, Neural Networks, and GPR. The best method is proposed using statistical modeling. This statistical modeling is performed using parameters such as Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Square Error (MSE), R^2, and Prediction Speed;
- After a comparative analysis of the statistical methods, the best method is determined. This best method is further optimized by tuning parameters using three methods—Bayesian Optimization, Grid Search, and Random Search—with the objective of minimizing MSE. The method giving the least MSE is identified and proposed.

A detailed flow chart of the proposed methodology is shown in Figure 2. It can be observed from Figure 2 that to select an accurate STLF model, data are first analysed and outliers are removed. Feature scaling of data is performed to normalize the independent variables of the data. After data analyses and feature scaling, data are trained using regression models such as SVM, Ensemble, Decision Trees, Neural Networks, and GPR. The best regression method is obtained by analyzing statistical parameters such as RMSE, MAE, MSE, R^2, and Prediction Speed. The best method is further optimized with the objective of minimizing MSE by tuning hyperparameters.

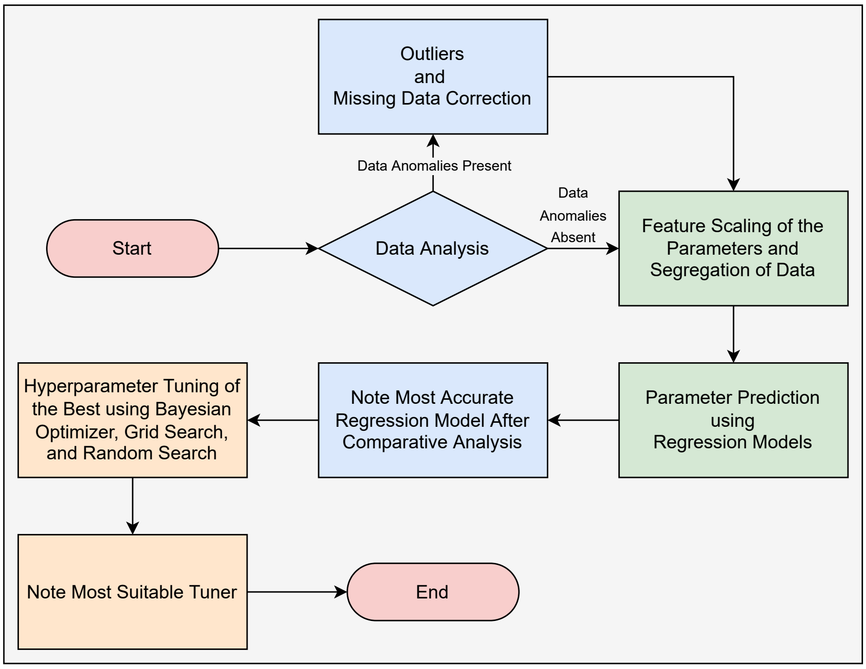

Outliers
and
Missing Data Correction
Data Anomalies Present
Data
Anomalies
Absent
Start
Data Analysis
Feature Scaling of the Parameters and Segregation of Data
Hyperparameter Tuning of the Best using Bayesian Optimizer, Grid Search, and Random Search
Note Most Accurate Regression Model After Comparative Analysis
Parameter Prediction using Regression Models
Note Most Suitable Tuner
End

Figure 2. Proposed methodology for short term load forecasting.

The rest of the paper is organized as follows. Section 2 explains the procedure for the data analysis of Load. Section 3 describes all the methodologies used in the paper. Section 4 deals with hyperparameter tuning. Section 5 presents the results and analysis followed by a conclusion in Section 6.

2. Data Analysis of Load

This section deals with the analysis and cleaning of data.

2.1. Outlier Treatment

A detailed data analysis was performed on the considered load data. The inter quartile method (IQR) method was used to detect outliers. A data entry that was caused due to experimental error, variation in measurement, data anomaly, or any other reason was called an outlier [30]. Although it is difficult to separate bad outliers from good outliers with precision, data that are extremely unfamiliar without any considerable reason are considered outliers. The presence of outliers may produce errors in the output by affecting the mean and standard deviation. In this paper, outliers were detected and removed. The following equations were used to detect outliers in the data.

$$IQR = Quart_3 - Quart_1 \tag{1}$$

$$LB = Quart_1 - 1.5 \times IQR \tag{2}$$

$$UB = Quart_3 + 1.5 \times IQR. \tag{3}$$

The difference between the third quartile (Quart_3) and the first quartile (Quart_1) was determined in (1). This difference was called IQR. The lower bound (LB) and upper bound (UB) of the data using IQR were calculated in (2) and (3). Once outliers that were observed beyond the LB ad UB were detected, they were removed from the system.

2.2. Statistical Parameter Descriptions

After outliers were detected and data were cleaned, the output was fed to various regression methods. A quantitative analysis of each regression method is presented. Quantitative analysis includes the calculation of RMSE, MAE, MSE, R^2, and Prediction Speed. Errors were obtained from the true versus the predicted value of the system [10]. RMSE was obtained by the square root of the mean of all the errors as shown in (4). MAE predicts the average of the modulated difference between actual and predicted, which is computed in (5). The average of the squared difference between average and predicted values are given by MSE which is calculated in (6). R^2 explains the variance in the linear dependency of a predicted value with respect to the true value which is determined in (7). It lies between 0 and 1, where 1 represents a perfect fit model and 0 represents a perfectly unfit model. A detailed explanation of all the statistical models is shown in [15]. After the best method was determined, the MSE was further minimized by tuning or varying hyperparameters such as leaf size. This was implemented by using three optimization techniques—Bayesian Optimization, Grid Search, and Random Search—after which optimal parameters were determined.

$$\text{RMSE} = \sqrt{\sum_{j=1}^{n} \frac{(\overline{y}_j - y_j)^2}{n}} \tag{4}$$

$$\text{MAE} = \frac{\sum_{j=1}^{n} (\overline{y}_j - y_j)}{n} \tag{5}$$

$$\text{MSE} = \frac{1}{n} \sum_{j=1}^{n} (\overline{y}_j - y_j)^2 \tag{6}$$

$$R^2 = 1 - \frac{\text{Anamolic Variation}}{\text{Total Variation}}, \tag{7}$$

where $\overline{y}_j$ and y_j indicate the predicted and true values of the data and n determines the total number of observations.

3. Methodologies

This section analyzes all the methods used for the prediction of STLF in this paper.

3.1. Decision Trees

To implement the Decision Tree (DT) algorithm, a tree-like structure was formed. Each node of the tree was assigned a condition. As per the conditional requirement, the decisions were progressed and arrived at an optimized decision. Various parameters determine the operation of the tree. A tree is composed of nodes, which are further connected by edges. The root node exists at the top of a decision tree, and this node is the origin of decision-making where conditional operation starts. The node where decision-making is performed is called the decision node and it is obtained after separating the root nodes. The node where further splitting is not possible is called the leaf node. Branches represent the flow from question to answer. The number of children at each node is determined by the branching factor. If this value varies, an average branching factor is used for computational purposes. The level of the tree is determined by the total number of parent nodes at that node. A sub tree is formed by a section of the tree. The subgraph is a component of the main graph. Pruning is performed to prevent overfitting; this is done by removing certain nodes. Overfitting can give false and deceptive output. Decision Trees can be used in both classifiers and regression.

Classification is performed to classify components into certain groups; a target variable is set based upon which a decision is taken using the Decision Tree algorithm to obtain the target variable. The set of rules which are obtained to achieve the target variable is noted and used for future simulations. Regression is performed to predict future values of the system. In this paper, decision trees were used to predict future values based on a set of decisions formed to achieve the target value. All the required parameters of a tree are presented in Figure 3. The structure of the tree can be seen in Figure 4.

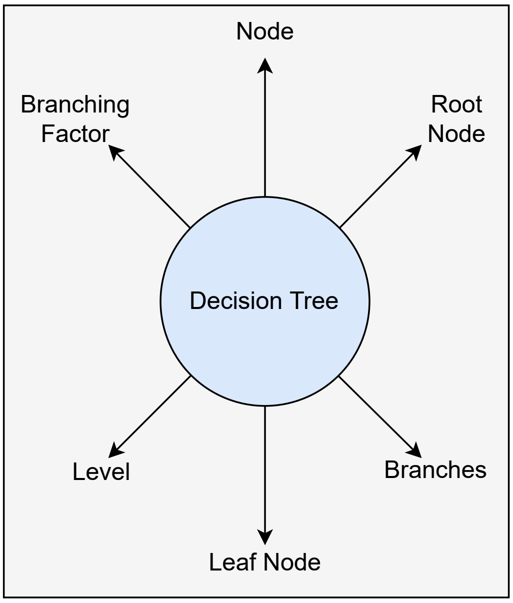


Figure 3. Various parameters of Decision Trees.

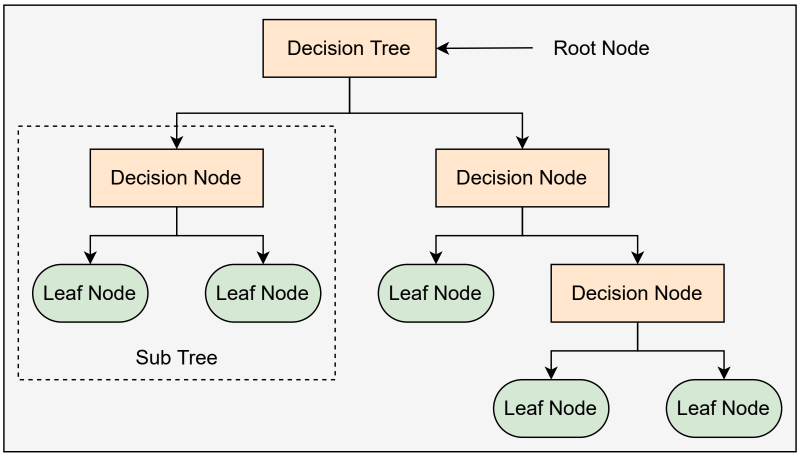


Figure 4. Structure of Decision Tree.

3.2. Linear Regression

Linear Regression is a type of supervised learning. This approach models the relationship between the independent output variable and the dependent input variable. This dependent variable can be one or more.

$$[Y] \approx \beta_0 + \beta_1 [X] \tag{8}$$

$$[Y] = \beta_0 + \beta_1 [X] + \varepsilon, \tag{9}$$

where $[Y]$ represents the output matrix and $[X]$ represents the input matrix. β_0 and β_1 are intercept and slope, respectively, and together are called model coefficients. ε is the error coefficient obtained after approximation. Linear Regression models are simple and provide a clear interpretation. The whole process of Linear Regression is dependent on intercept and slope. Intercept and slope are estimated using Linear Regression algorithms and then compared with actual values. The objective function is formed in such a way that the minimum error is obtained. The set of values related to intercept and slope, which give the minimum error, uses precise model parameters to train. The training time of Linear Regression models is fast and accurate. The main issue with Linear Regression models is linearity. For effective implementation of Linear Regression models, there should be linearity between $[Y]$ and $[X]$, otherwise, ε will be very high.

3.3. Support Vector Machine (SVM)

SVM uses kernels to train the algorithms. Kernels can be termed as a functional relationship between two observations. These functional relationships can be linear—Polynomial and Radial. The following equations determine the functions of different kernels.

$$K(x_i, x_i{}') = \sum_{j=1}^{p} x_{i,j}.x_{i',j} \tag{10}$$

$$K(x_i, x_i') = \left(1 + \sum_{j=1}^{p} (x_{i,j}) \times (x_{i',j})\right)^{p} \tag{11}$$

$$K(x_i, x_i') = \exp\left(-\gamma \sum_{j=1}^{p} (x_{i,j} - x_{i',j})^2\right), \tag{12}$$

where x_i and x_i' are the observations, p is the degree of polynomial and γ is a positive constant. The linear relationship is determined in (10), whereas polynomial and radial relationships are computed in (11) and (12), respectively.

The dominant factor in deciding the rules of SVM is the preciseness of the hyperplane and hyperplane shape. A suitable hyperplane is searched in N-dimensional space. The plane with the best margin is considered to be the best fit. Support vectors are situated close to the hyperplane.

3.4. Gaussian Process Regression (GPR)

GPR is one of the best methods for providing uncertainty measurement. It follows the Bayesian approach to regression. It is an efficient technique for small data sets. GPR follows a nonparametric approach. A nonparametric approach is an approach that is not limited by its functional form. A probability distribution can be defined for a specific function or for all the functions that fit the data. Since GPR has a nonparametric approach, it computes the probability distribution of functions that fit the data. Two main aspects of GPR are Gaussian Process Prior (GPP) and Kernel function. Since GPR depends a lot on the prior values, GPP plays a vital role in computing algorithms for GPR. The equation for GPP can be computed from (13). RBF (Radial Basis Function) is considered as a kernel in this paper. This function is calculated using (14).

$$f(x) \sim GP(m(x), k(x, x^{'})) \tag{13}$$

$$k(x, x^{'}) = \partial_f^2 \exp(-\frac{1}{2l_m^2}\left\|x - x^{'}\right\|^2), \tag{14}$$

where $m(x)$ is mean function, $k(x, x^{'})$ is a covariance function, ∂_f^2 is signal variance and l_m is the length scale of the system.

3.5. Neural Networks

The main inspiration behind neural networks is the working characteristics of neurons in the human brain. Hence, historical data play a key role in working neural networks. The more data, the more robust its learning algorithm will be. It is a function-based approach, where a function is decided based on the operating relationship between data input and data output. The methodology used in this paper is ANN (Artificial Neural Network). Two main components of ANN are the transfer function and activation function. The complex pattern in the data is learned by the activation function. The activation function plays as a selector, i.e., it chooses which signal from one neuron to another neuron should pass. The activation function can be both linear and nonlinear. While transfer functions are used to convert input signals into output signals, activation functions operate on a threshold value that, when crossed, triggers the signal. A detailed diagram of ANN is shown in Figure 5 and the description of the activation function is shown in Figure 6.

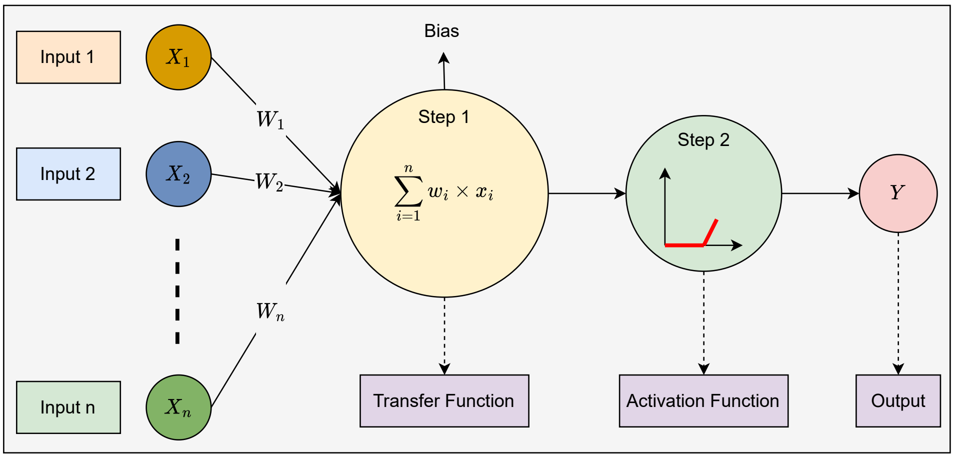


Figure 5. Structure of ANN.

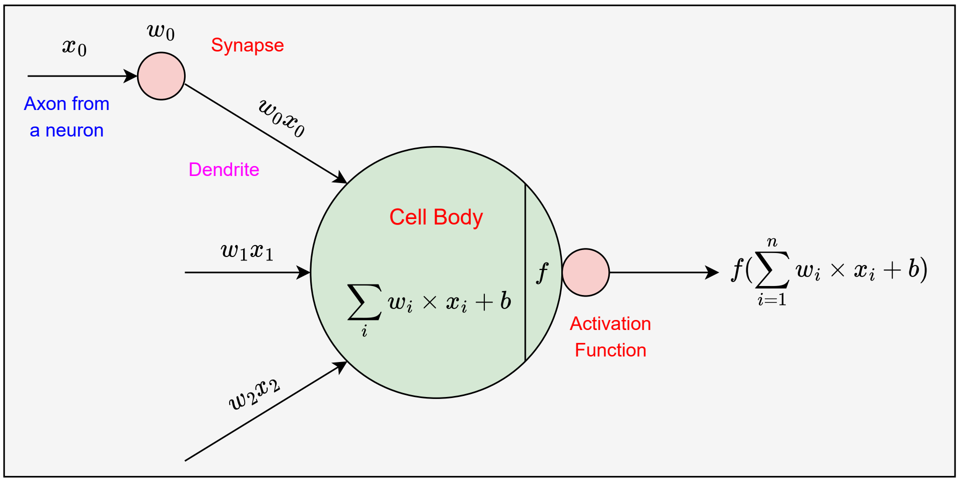


Figure 6. Activation function of ANN.

3.6. Ensemble

Ensemble systems combine two or more functions to give the desired output. Ensembles can be both homogeneous and heterogeneous. To compute Ensemble, a base method is first identified based on number of data points. In this manuscript, Decision Trees were considered a base method because of their ability to handle large data sets. The remaining methods were tested in combination with base methods using trail and error. In this manuscript, the methods tested were SVM, GPR and Linear Regression, in combination with base methods. Homogeneous methods have the same base methods, while heterogeneous methods have different base methods. Ensembles are classified as Bagging, Boosting, and Stacking. The functional approach of each of these methods can be referred to in [14]. The biggest issues of Ensemble techniques are data bias and the quality of base methods. A large data set is divided into groups and then each group is trained based on probability distribution and using a distinct methodology. Finally, all results from each method are accumulated into a single result and are presented as output. It is upon the user to decide a number of parts based on the computation power his system possesses.

4. Hyperparameter Tuning

Hyperparameter tuning is generally performed to find a set of optimal parameters that will help to train the model without compromising on any parameter. Two important components to compute optimization algorithms are objective functions and constraints. In this paper, the objective of reducing MSE is considered and constraints are obtained by training parameters of the methodology. Three methods are used and compared to tuned hyperparameters in this paper.

4.1. Grid Search

Grid Search is the most commonly used method in hyperparameter tuning. Firstly, a set of hyperparameters is formed. If there are too many hyperparameters, some are eliminated based on their importance. The dominant hyperparameters are then combined to form sets and are plotted in the form of a grid. Then, each combination is tested and the most optimum combination is obtained as a result. It is considered an exhaustive algorithm as it checks all possible combinations of the system. The operation of the algorithm is slow because of this exhaustiveness. Checking each point can be complicated and costly. However, the accuracy obtained from this method can be high. A demonstration of the Grid Search algorithm with two hyperparameters is given in Figure 7. It can be observed from Figure 7 that, when two hyperparameters are plotted, every single value is considered

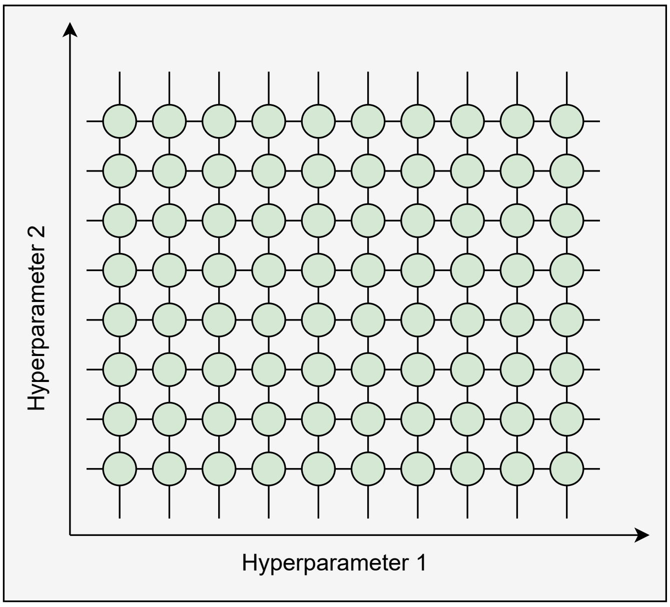


Figure 7. Grid Search sample plotting for two hyperparameters.

4.2. Random Search

Random Search uses randomized combinations to check satisfaction for optimal parameters of the systems. The concept is basically derived from Grid Search, but every parameter is not validated. Only random parameters are validated. Because of this Randomness, the accuracy of the tuning method can be compromised, but the training time of the system is improved. When there are large hyperparameters to optimize, a Random Search can be a decent option. A demonstration of a Random Search algorithm with two hyperparameters is given in Figure 8. It can be observed from Figure 8 that, when two hyperparameters are plotted, only random values are considered.

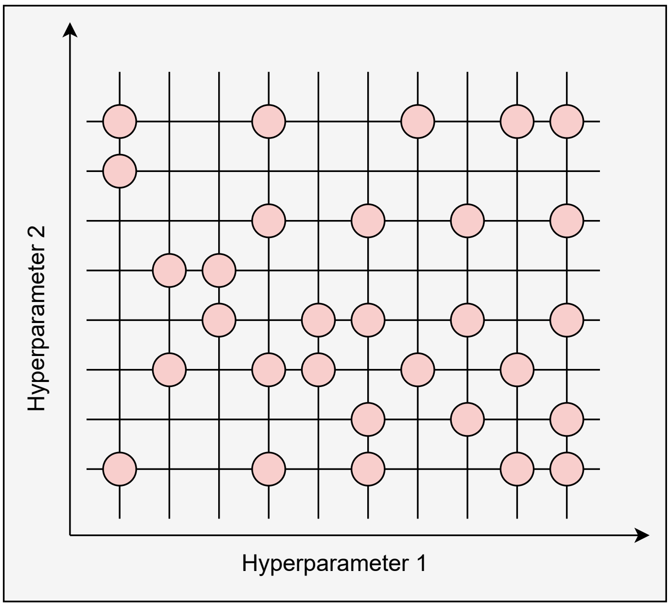


Figure 8. Random Search sample plotting for two hyperparameters.

4.3. Bayesian Optimization

The fundamental theorem behind Bayesian Optimization is the Bayesian Equation. Bayes theorem is used for conditional probability, where the probability of a certain event is known based upon which the probability of another event is calculated. The description of Bayes theorem is shown in (15).

$$p(score|hpm) = \frac{p(score|hpm) \times p(score)}{p(hpm)}, \tag{15}$$

where $p(hpm)$ is the probability of the occurrence of the hyperparameter value and $p(score)$ is the probability of the occurrence of the required value. Bayesian Optimization can be called an informed search technique where the possibility of certain events is known. It considers past values for computational purposes. The pattern extracted from data from past values is taken into account. It focuses on those parameter spaces, which can potentially produce an optimal solution.

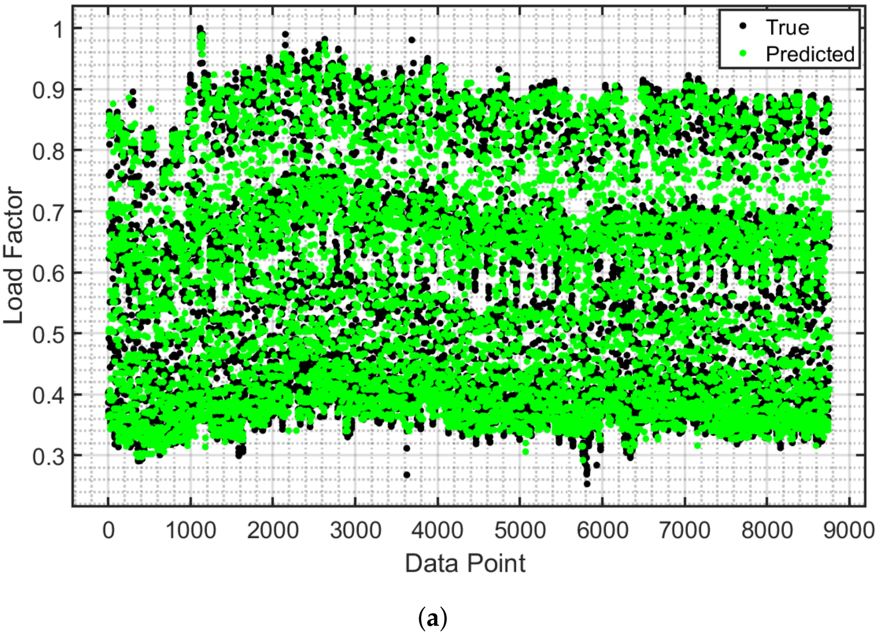
5. Results and Analysis

This Section analyzes the results of all the discussed methods and proposes the best method for STLF.

5.1. Case Studies and Results

The proposed methodology has been implemented using the data set given in [31]. The electricity supply company of the Malaysian city of Johor generated these hourly load data. This data set shows the variation of load corresponding to day, month, year, time and temperature. The best correlation factor is obtained from combining all the input parameters to obtain the output. A combination of all input parameters is 93.8 percent correlated to the output. The methods have also been tested using the data of [32,33] with similar input parameters. While investigating the data, it was found that around 2.5% of the data set was outliers. These outliers were removed. The output load data were scaled between 0 to 1 with the base quantity as the peak value. Hold-out validation was used with 80 and 20 for testing and training, respectively. The data set was trained with algorithms consisting of Ensemble, DTs, Neural Networks, SVM, GPR, and Linear Regressions. The output of DTs is shown in Figure 9. Figure 9a represents the true versus the predicted values of the system, and Figure 9b indicates how fit the model is and is used for the calculation of R^2. If there is an overfitting issue, it has been crosschecked using cross-validation, replacing it with hold out validation. Similarly, the corresponding results of Linear Regression, SVM, Ensemble, GPR and Neural Networks are shown in Figures 10–14, respectively.

Detailed statistical analyses of all the methods are shown in Table 2. It can be observed that the best method was found to be a Decision Tree. The output of the Decision Tree was further optimized with the objective to reduce MSE, and hyperparameters were further tuned. Three different methods—Bayesian Optimization, Grid Search, and Random Search—were simulated and compared. A detailed plot for hyperparameter tuning is presented in Figure 15. Parameters considered for the tuning of Decision Trees were leaf size and number of iterations. The number of iterations is significant with respect to the training time of the method. The optimal hyperparameters are shown in Table 3. It can be observed from Table 3 that the best method for hyperparameter tuning on this data set is Grid Search with a leaf size of 6. Results of all hyperparameter tuning of all the methods can be referred to in Table A1.



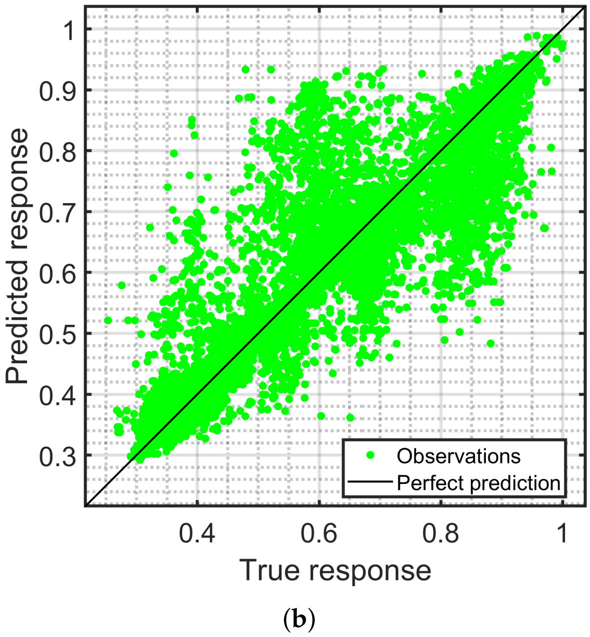


Figure 9. Results with Decision Trees. (**a**) Actual versus predicted values of Decision Trees. (**b**) Linearity analysis of Decision Trees.

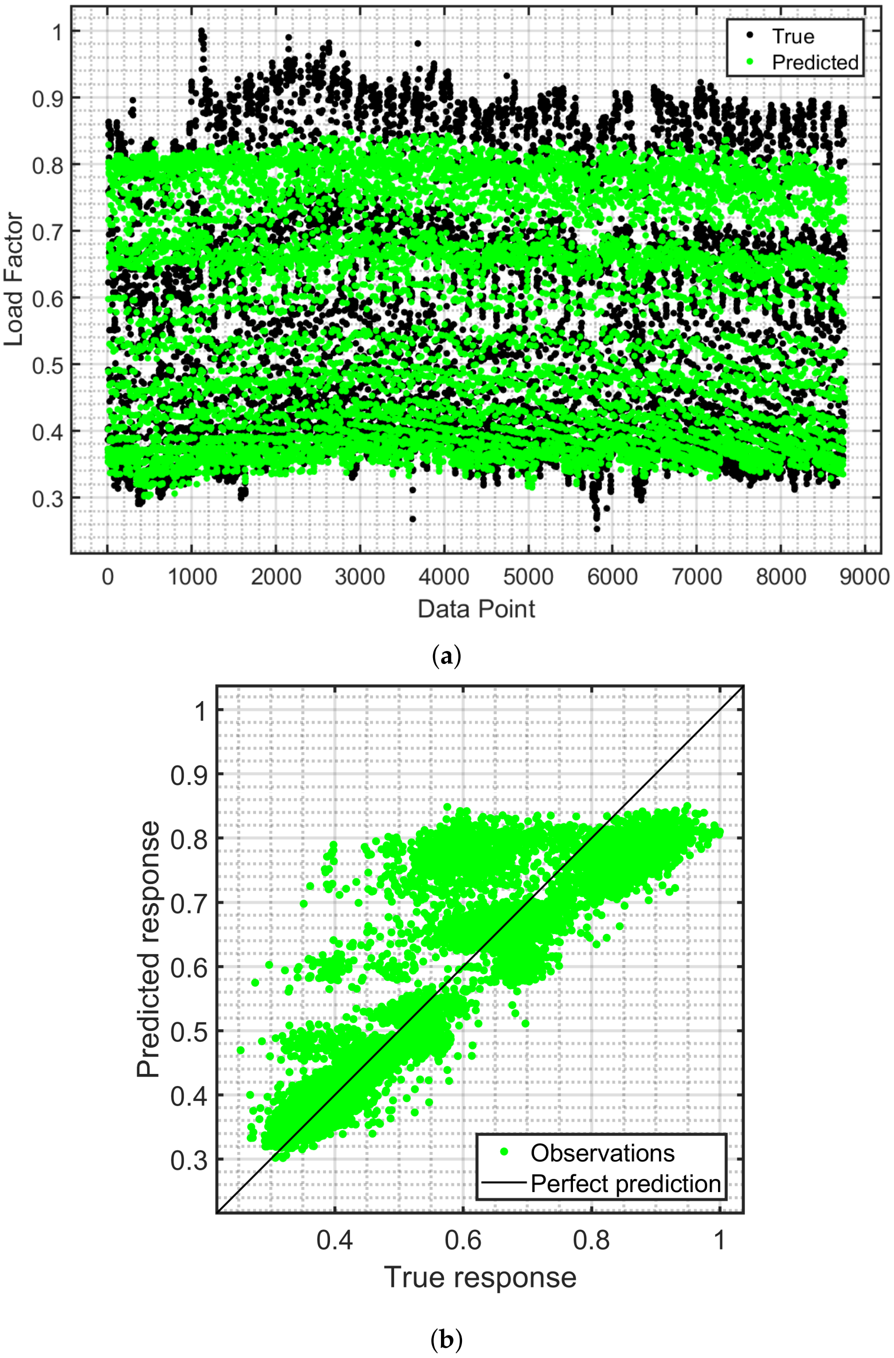
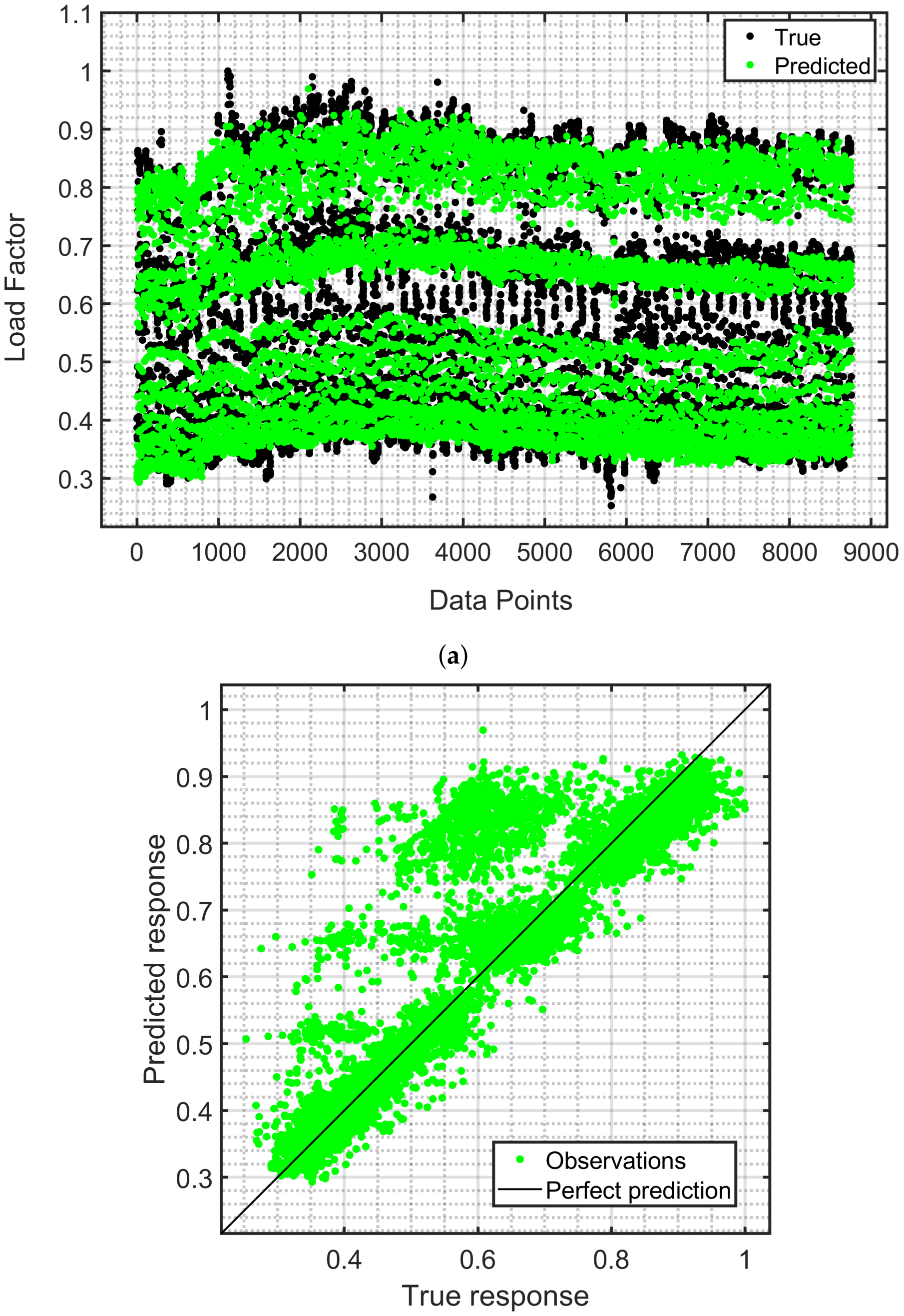


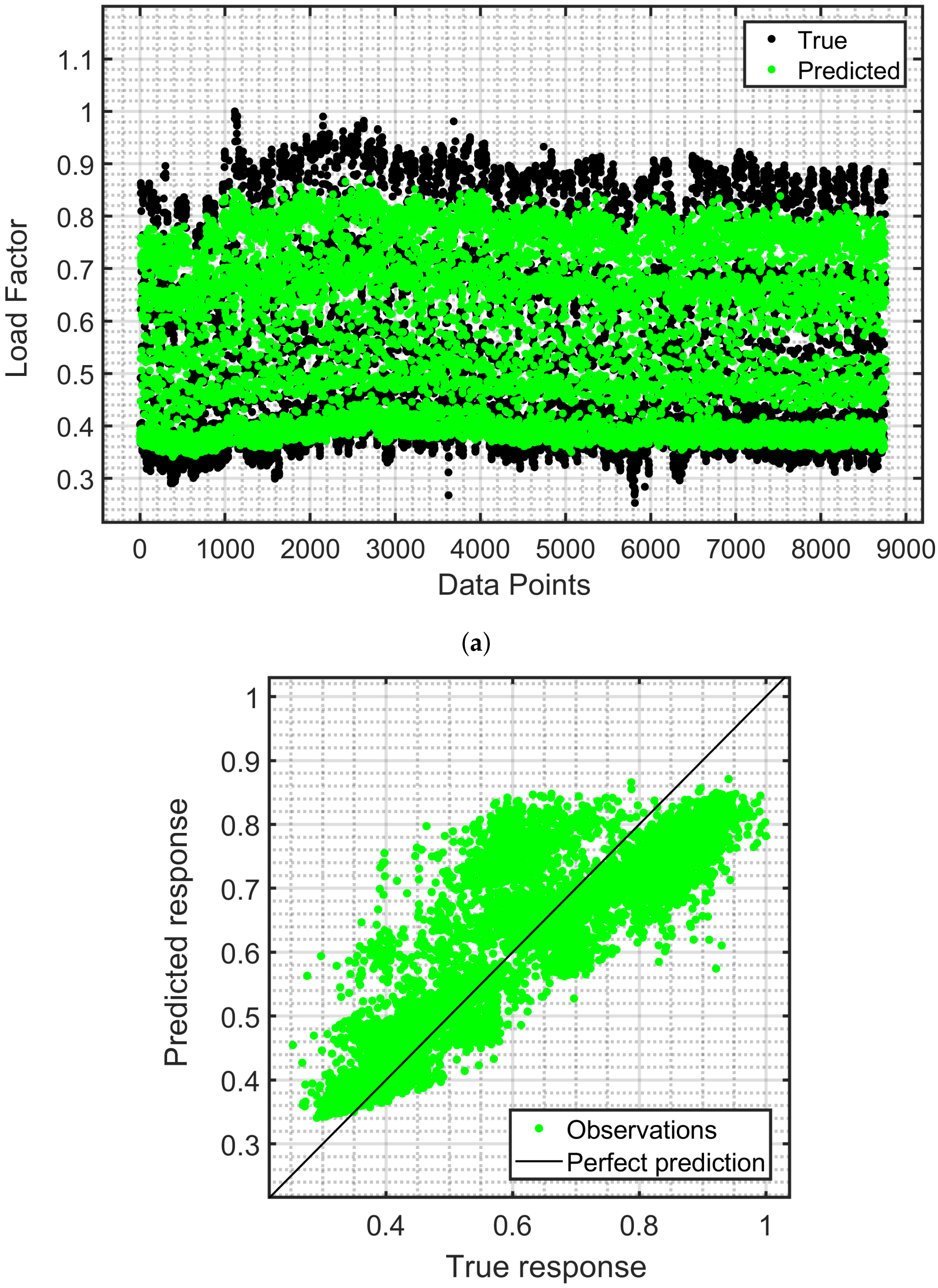
Figure 10. Results with Linear Regression. (**a**) Actual versus predicted values of Linear Regression. (**b**) Linearity analysis of Linear Regression.



(**a**)

(**b**)

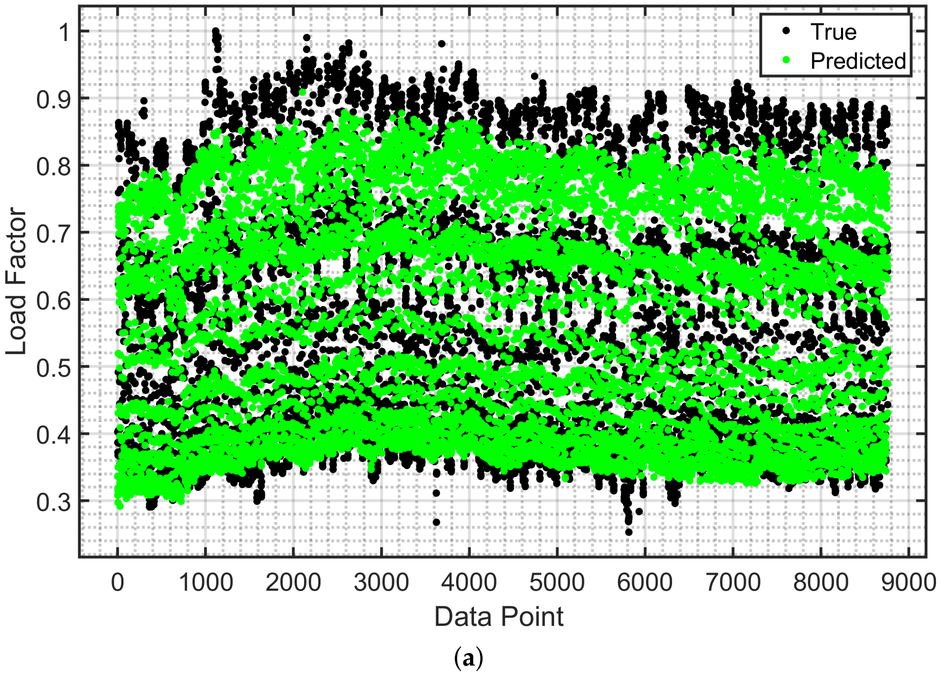
Figure 11. Results with Support Vector Machines. (**a**) Actual versus predicted values of Support Vector Machines. (**b**) Linearity analysis of Support Vector Machines.



(**a**)

(**b**)

Figure 12. Results with Ensemble. (**a**) Actual versus predicted values of Ensemble. (**b**) Linearity analysis of Ensemble.



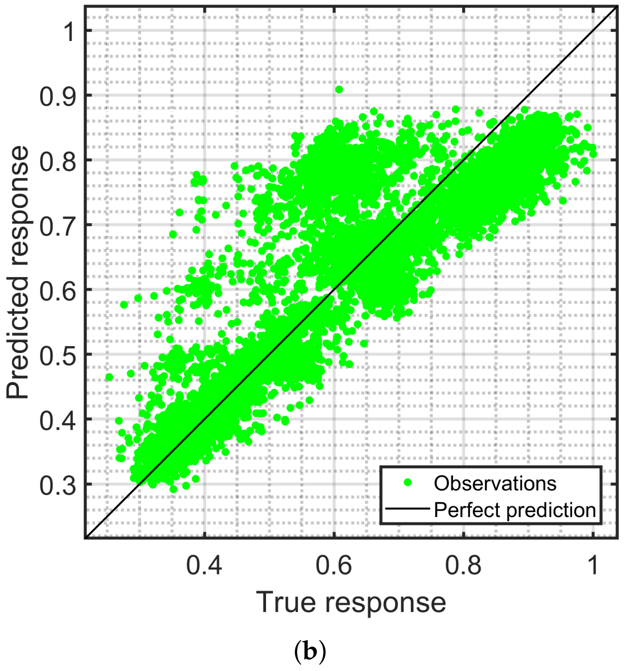
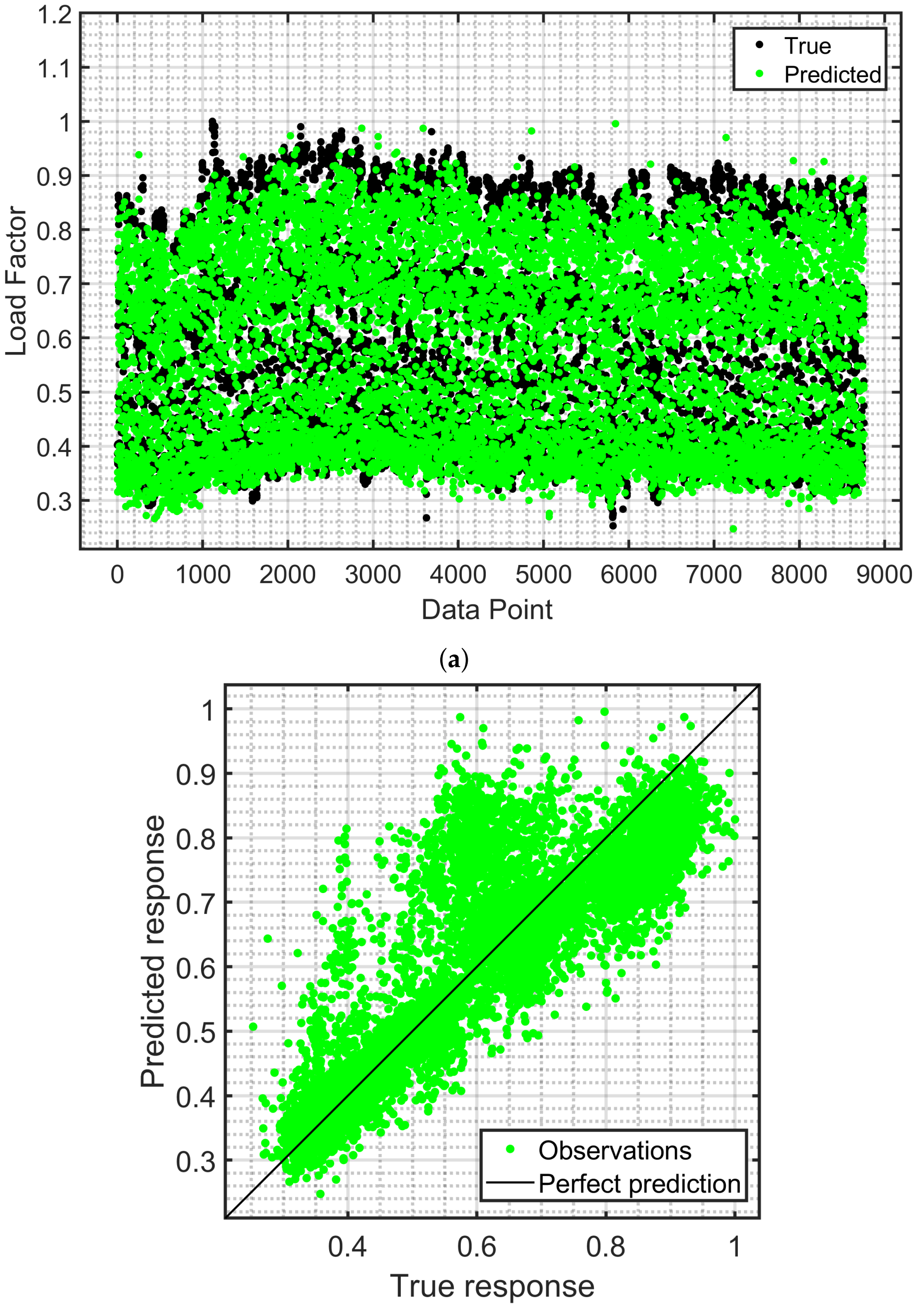


Figure 13. Results with Gaussian Process Regression. (**a**) Actual versus predicted values of Gaussian Process Regression. (**b**) Linearity analysis of Gaussian Process Regression.



(**a**)

(**b**)

Figure 14. Results with Artificial Neural Network. (**a**) Actual versus predicted values of Artificial Neural Network. (**b**) Linearity analysis of Artificial Neural Network.

Table 2. Comparative analysis of forecasting methods.

Method	RMSE	R^2	MSE	MAE	MAPE	Training Time (s)
Decision Tree	0.087	0.85	0.0077	0.05	4.8	1.32
Linear Regression	0.13	0.62	0.019	0.13	12.5	0.72
SVM	0.085	0.78	0.0078	0.053	5.2	12.14
Ensemble	0.1	0.77	0.01	0.067	6.2	1.74
GPR	0.07	0.72	0.005	0.054	5.3	1151.5
Neural Network	0.07	0.82	0.005	0.061	5.9	8.02

Table 3. Hyperparameter tuning results (Decision Tree).

Method	MSE	Leaf Size
Bayesian Optimization	0.0070	3
Grid Search	0.0067	6
Random Search	0.0069	5

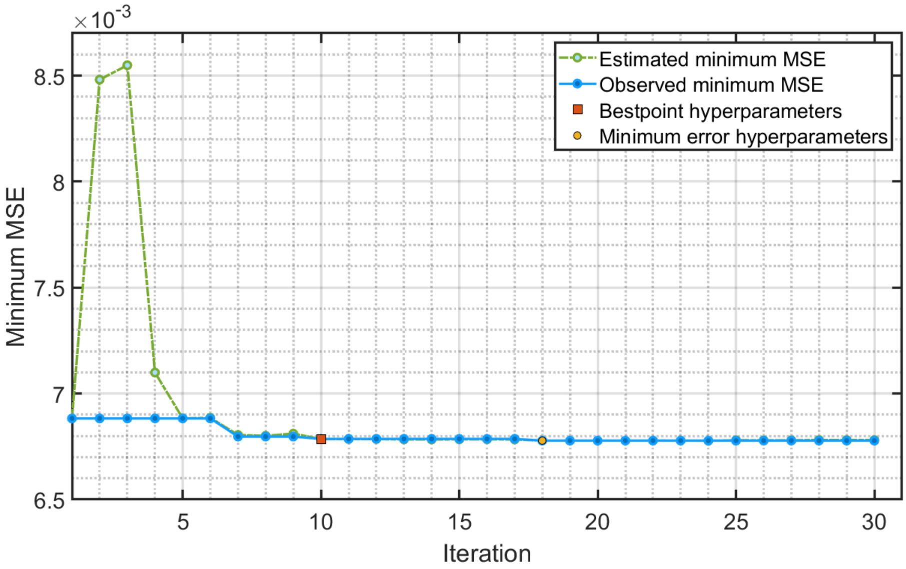


Figure 15. Tuned hyperparameters Graphs of Tree using Grid Search.

Thus, three different methods to tune hyperparameters were analyzed to understand the functionality of tuning.

5.2. Result Interpretation

It can be observed from Table 2 that:

- The performance of Linear Regression methods is not fit—this is because there is no linear relation between data points;
- SVM produces good results but the training time of SVM is relatively high as compared to that of Decision Tree. Additionally, SVM is highly dependent on the accuracy of the hyperplane boundary, which can give inaccurate results during changes in data;
- The time taken to process GPR is the highest, hence this method can be highly ineffective when the load forecasting is integrated with applications such as power trading. Although Neural Networks are producing promising results, when compared to the training time of Decision Tree, it is on the higher side;
- The accuracy of Ensemble is highly dependent on the combinations of the methods assumed. The fitting of the model (R^2) is less as compared to Decision Trees.

Hence, considering all the parameters, it can be concluded that Decision Trees can be an effective method for predicting short term load.

5.3. Comparative Analysis

Various works have been published in the area of STLF. Most of the works have focused on individual methods and the improvement of Individual methods. This paper, instead of focusing on individual methods, focused on numerous methods and on understanding the merit and demerit of each method. In [34], the focus is completely on ANN. ANN is efficient when the data set is large; it is also affected by data sensitivity but it is important to understand that not all load data sets might give consistent patterns, and also there might be data scarcity. So, ANN might not be the best method for specific data sets. The algorithm used in [35,36] is Recurrent Neural Network (RNN). The issue with RNN is the selection of an activation function and the system can be quite slow in producing results. Load data with long sequences might not be processed easily. Since load data are highly dynamic in nature and vary according to geographical location, RNN might not produce the best results. Since RNN follows a deep learning architecture, its operational cost is also quite high. So, as compared to ANN used in [34], and compared to deep learning methods used in [35,36], the methodology used in this paper tries to behave as per the nature of the data set and produce an efficient algorithm. Some of the algorithms produced in this area are highly individualistic in nature; Reference [12,14] focused only on a single method whereas this paper focuses on crucial regression methods. The methodology used in [15] is close to that of this paper but lacks a comparative analysis of important hyperparameter tuning methods, whereas this paper gives a broad comparative analysis of hyperparameter tuning methods.

6. Conclusions

The following objectives have been achieved in this paper:

- Detailed data analyses of the assumed data were performed. It was found that 2.5% of the data contained outliers using the IQR method. These outliers were removed;
- The best regression model for depicting load was found to be Decision Tree with an RMSE, R^2, MSE, MAE and training time of 0.087, 0.85, 0.0077, 0.05, and 1.32, respectively;
- The best optimizer for hyperparameter tuning was achieved using Grid Search with a leaf size of 6 and a reduced MSE of 0.0067861.

A Decision Tree is one of the best methods for solving the issue of STLF because the load has a dynamic nature. It changes with seasons, days, and months, among other parameters. Linear Regression methods did not perform well, with a very low R^2 value, which implies it is difficult to establish linear relations among input and output parameters. Since the data set is large, Neural Networks also produced reasonably good results but, because of their computational cost, they lag behind Decision Trees. As far as hyperparameter tuning is concerned, Grid Search is considered the best method because of its certainty. Bayesian Optimization has more reliability than Grid Search but it requires strong, pre-computed values which might not be available with data sets. Randomized Search for hyperparameter tuning can be highly unpredictable since it depends on random values.

Based on the proposed research work, the following areas can be promising future aspects of the research:

- The forecasted load active power can be scheduled with constraints obtained from Distributed Energy Sources integrated with the grid;
- Calculation of Localized Marginal Pricing (LMP) and Distributed Localized Marginal Pricing (DLMP) can be carried out with forecasted load;
- A more robust comparison with Deep Learning methods can be carried out and the advantages and disadvantages of each individual method can be presented;
- Peer to Peer trading using forecasted load can be analyzed.

Author Contributions: Conceptualization, B.V.S.V., I.S. and P.T.; methodology, B.V.S.V., M.K., I.S., P.T. and N.D.B.; software, B.V.S.V. and I.S.; validation, M.K., P.T. and N.D.B.; formal analysis, B.V.S.V.; investigation, B.V.S.V. and I.S.; resources, B.V.S.V., M.K., I.S. and N.D.B.; data curation, B.V.S.V.; writing—original draft preparation, B.V.S.V., M.K., I.S., P.T. and N.D.B.; writing—review and editing, B.V.S.V., M.K., I.S., P.T. and N.D.B.; visualization, B.V.S.V., I.S. and N.D.B.; supervision, M.K. and N.D.B.; project administration, M.K. and N.D.B.; funding acquisition, M.K. and N.D.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

PSO	power system operators
STLF	short-term load forecasting
DT	decision tree
SVM	support vector machine
GPR	gaussian process regression
RMSE	root mean square error
MAE	mean absolute error
MSE	mean sqaure error
DBMS	database management system
IQR	interquartile range
RBF	radial basis function
ANN	artificial neural network
LMP	localized marginal pricing
DLMP	distributed localized marginal pricing

Appendix A. Hyperparameter Tuning

Table A1. Results of Hyperparameter tuning for methods used in the study.

Method	Tuning Method	MSE
Decision Trees	Bayesian	0.007
	Grid Search	0.0067
	Random Search	0.0069
Linear Regression	Bayesian	0.0182
	Grid Search	0.017
	Random Search	0.0185
SVM	Bayesian	0.0066
	Grid Search	0.0065
	Random Search	0.0067
Ensemble	Bayesian	0.009
	Grid Search	0.008
	Random Search	0.0095
GPR	Bayesian	0.0045
	Grid Search	0.004
	Random Search	0.0048
Neural Network	Bayesian	0.0048
	Grid Search	0.0045
	Random Search	0.0049

References

1. Vardhan, B.S.; Khedkar, M.; Srivastava, I. Effective energy management and cost effective day ahead scheduling for distribution system with dynamic market participants. *Sustain. Energy Grids Netw.* **2022**, *31*, 100706. [CrossRef]
2. Bokde, N.D.; Pedersen, T.T.; Andresen, G.B. Optimal Scheduling of Flexible Power-to-X Technologies in the Day-ahead Electricity Market. *arXiv* **2021**, arXiv:2110.09800.
3. Ministry of Power, India. Annual Report, 2021-22. Available online: https://powermin.gov.in/sites/default/files/uploads/MOP_Annual_Report_Eng_2021-22.pdf (accessed on 5 October 2022).
4. Shewale, A.; Mokhade, A.; Funde, N.; Bokde, N.D. An overview of demand response in smart grid and optimization techniques for efficient residential appliance scheduling problem. *Energies* **2020**, *13*, 4266. [CrossRef]
5. Shewale, A.; Mokhade, A.; Funde, N.; Bokde, N.D. A Survey of Efficient Demand-Side Management Techniques for the Residential Appliance Scheduling Problem in Smart Homes. *Energies* **2022**, *15*, 2863. [CrossRef]
6. Srivastava, I.; Bhat, S.; Vardhan, B.V.S.; Bokde, N.D. Fault Detection, Isolation and Service Restoration in Modern Power Distribution Systems: A Review. *Energies* **2022**, *15*, 7264. [CrossRef]

7. Kumar, S.; Mishra, S.; Gupta, S. Short Term Load Forecasting Using ANN and Multiple Linear Regression. In Proceedings of the 2016 Second International Conference on Computational Intelligence Communication Technology (CICT), Ghaziabad, India, 12–13 February 2016; pp. 184–186. [CrossRef]
8. Aprillia, H.; Yang, H.T.; Huang, C.M. Statistical Load Forecasting Using Optimal Quantile Regression Random Forest and Risk Assessment Index. *IEEE Trans. Smart Grid* **2021**, *12*, 1467–1480. [CrossRef]
9. Yang, L.; He, M.; Zhang, J.; Vittal, V. Support-Vector-Machine-Enhanced Markov Model for Short-Term Wind Power Forecast. *IEEE Trans. Sustain. Energy* **2015**, *6*, 791–799. [CrossRef]
10. Vardhan, B.V.S.; Khedkar, M.; Shahare, K. A Comparative Analysis of Various Stochastic approaches for Short Term Load Forecasting. In Proceedings of the 2022 International Conference for Advancement in Technology (ICONAT), Goa, India, 21–22 January 2022; pp. 1–6. [CrossRef]
11. Ding, C.; Bao, T.Y.; Huang, H.L. Quantum-Inspired Support Vector Machine. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 7210–7222. [CrossRef]
12. Xie, Z.; Wang, R.; Wu, Z.; Liu, T. Short-Term Power Load Forecasting Model Based on Fuzzy Neural Network using Improved Decision Tree. In Proceedings of the 2019 IEEE Sustainable Power and Energy Conference (iSPEC), Beijing, China, 21–23 November 2019; pp. 482–486. [CrossRef]
13. Ray, P.K.; Mohanty, S.R.; Kishor, N.; Catalão, J.P.S. Optimal Feature and Decision Tree-Based Classification of Power Quality Disturbances in Distributed Generation Systems. *IEEE Trans. Sustain. Energy* **2014**, *5*, 200–208. [CrossRef]
14. Malekizadeh, M.; Karami, H.; Karimi, M.; Moshari, A.; Sanjari, M. Short-term load forecast using ensemble neuro-fuzzy model. *Energy* **2020**, *196*, 117127. [CrossRef]
15. Vardhan, B.V.S.; Khedkar, M.; Srivastava, I. Cost Effective Day -Ahead Scheduling with Stochastic Load and Intermittency Forecasting for Distribution System Considering Distributed Energy Resources. *Energy Sources Part A Recover. Util. Environ. Eff.* **2021**, 1–26. [CrossRef]
16. Krannichfeldt, L.V.; Wang, Y.; Zufferey, T.; Hug, G. Online Ensemble Approach for Probabilistic Wind Power Forecasting. *IEEE Trans. Sustain. Energy* **2022**, *13*, 1221–1233. [CrossRef]
17. Agenis-Nevers, M.; Bokde, N.D.; Yaseen, Z.M.; Shende, M.K. An empirical estimation for time and memory algorithm complexities: Newly developed R package. *Multimed. Tools Appl.* **2021**, *80*, 2997–3015. [CrossRef]
18. Gan, L.K.; Zhang, P.; Lee, J.; Osborne, M.A.; Howey, D.A. Data-Driven Energy Management System With Gaussian Process Forecasting and MPC for Interconnected Microgrids. *IEEE Trans. Sustain. Energy* **2021**, *12*, 695–704. [CrossRef]
19. Nejati, M.; Amjady, N. A New Solar Power Prediction Method Based on Feature Clustering and Hybrid-Classification-Regression Forecasting. *IEEE Trans. Sustain. Energy* **2022**, *13*, 1188–1198. [CrossRef]
20. Hosein, S.; Hosein, P. Load forecasting using deep neural networks. In Proceedings of the 2017 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), Washington, DC, USA, 23–26 April 2017; pp. 1–5. [CrossRef]
21. Mamun, A.A.; Sohel, M.; Mohammad, N.; Haque Sunny, M.S.; Dipta, D.R.; Hossain, E. A Comprehensive Review of the Load Forecasting Techniques Using Single and Hybrid Predictive Models. *IEEE Access* **2020**, *8*, 134911–134939. [CrossRef]
22. Pirjatullah.; Kartini, D.; Nugrahadi, D.T.; Muliadi.; Farmadi, A. Hyperparameter Tuning using GridsearchCV on The Comparison of The Activation Function of The ELM Method to The Classification of Pneumonia in Toddlers. In Proceedings of the 2021 4th International Conference of Computer and Informatics Engineering (IC2IE), Depok, Indonesia, 14–15 September 2021; pp. 390–395. [CrossRef]
23. Srivastava, I.; Bhat, S.; Thadikemalla, V.S.G.; Singh, A.R. A hybrid machine learning and meta-heuristic algorithm based service restoration scheme for radial power distribution system. *Int. Trans. Electr. Energy Syst.* **2021**, *31*, e12894. [CrossRef]
24. S, G.; Brindha, S. Hyperparameters Optimization using Gridsearch Cross Validation Method for machine learning models in Predicting Diabetes Mellitus Risk. In Proceedings of the 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT), Chennai, India, 10–11 March 2022; pp. 1–4. [CrossRef]
25. Bergstra, J.; Bengio, Y. Random Search for Hyper-Parameter Optimization. *J. Mach. Learn. Res.* **2012**, *13*, 281–305.
26. Jerrell, M.E. A random search strategy for function optimization. *Appl. Math. Comput.* **1988**, *28*, 223–229.[CrossRef]
27. Nguyen, V. Bayesian Optimization for Accelerating Hyper-Parameter Tuning. In Proceedings of the 2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), Sardinia, Italy, 3–5 June 2019; pp. 302–305. [CrossRef]
28. Joy, T.T.; Rana, S.; Gupta, S.; Venkatesh, S. Hyperparameter tuning for big data using Bayesian optimisation. In Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 4–8 December 2016; pp. 2574–2579. [CrossRef]
29. Yu, T.; Zhu, H. Hyper-parameter optimization: A review of algorithms and applications. *arXiv* **2020**, arXiv:2003.05689.
30. Shende, M.K.; Feijóo-Lorenzo, A.E.; Bokde, N.D. cleanTS: Automated (AutoML) tool to clean univariate time series at microscales. *Neurocomputing* **2022**, *500*, 155–176. [CrossRef]
31. Guimaraes, F.; Javedani Sadaei, H. Load Data in Malaysia. 24 March 2019. Available online: https://data.mendeley.com/datasets/f4fcrh4tn9/1 (accessed on 2 November 2022). [CrossRef]
32. Zhang, J.; Feng, C. Short-term load forecasting data with hierarchical advanced metering infrastructure and weather features. *IEEE Dataport* **2019**. [CrossRef]
33. Farrokhabadi, M. Day-Ahead Electricity Demand Forecasting: Post-COVID Paradigm. *IEEE Dataport* **2020**. [CrossRef]

34. Baliyan, A.; Gaurav, K.; Mishra, S.K. A Review of Short Term Load Forecasting using Artificial Neural Network Models. *Procedia Comput. Sci.* **2015**, *48*, 121–125. [CrossRef]
35. Bianchi, F.M.; Maiorino, E.; Kampffmeyer, M.C.; Rizzi, A.; Jenssen, R. An overview and comparative analysis of recurrent neural networks for short term load forecasting. *arXiv* **2017**, arXiv:1705.04378.
36. Zhang, B.; Wu, J.L.; Chang, P.C. A Multiple Time Series-Based Recurrent Neural Network for Short-Term Load Forecasting. *Soft Comput.* **2018**, *22*, 4099–4112. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.