NIPES

Journal of Materials Engineering, Structures and Computation

Journal homepage: www.nipesjournals.org.ng

A Review of Short-Term Electrical Load Forecasting Using Ensemble Stacking Generalization with Artificial Neural Network

Are, M. T[1], Okorie, P. U.[2]*, Olarinoye, G. A.[3]

[1,2,3] Department of Electrical Engineering, Faculty of Engineering, Ahmadu Bello University, Zaria-Nigeria.

*corresponding mail-patrickubeokorie@yahoo.com

ARTICLE INFORMATION

Article history:
Received 15 February 2023
Revised 26 February 2023
Accepted 08 March 2023
Available online 22 March 2023

Keywords:
electricity load forecasting, machine learning, STLF, ANN, EVA, GBRM, SNN, RF

https://doi.org/10.5281/zenodo.7760014

https://nipesjournals.org.ng

ABSTRACT

Electric load forecasting has gained much attention in electricity production due to its important role in electric power system management. Short-term load forecasting (STLF) uses the perception of ensemble learning approaches as a general scheme for educating the prognostic skill of a machine learning model (MLM). STLF is subjected to numerous errors /problems like high bias and variance. This prompts the need for the employment of ensemble stacking generalization with artificial neural networks (ANN) to ensure an improved performance with accurate results. This approach combined four models namely random forest (RF), generalized boosted regression model (GBRM), Evolutional Algorithm (EvA), and artificial neural network (ANN). The inner mechanism of the stacked EvA-RF-GBRM-ANN model involves creating meta-data from EvA, RF, and GBRM models to calculate the final estimates using ANN. This work proposes a stacked neural network for short-term load forecasting through a view of dropping predicting faults besides their discrepancy associated with sole-based models and stacked neural networks (SNN).

1. Introduction

To supply electric energy to the customer securely and economically, an electric company faces many economical and technical challenges in operation [1]. Stable and uninterrupted high-quality electric energy provides a guarantee for the stable operation of industry and society [2].
Therefore, to ensure the stable operation of the power system and provide economic and reliable power for the market, it is necessary to accurately predict the change of load when planning the power system [2]. Load Forecasting is also one of the most emerging fields of research for this important and challenging field in the last few years [1]. Load forecasting has been a topic of interest for many decades and the literature is plenty with a wide variety of techniques [3]. Forecasting methods can be divided into three different categories: time-series approaches, regression-based, and artificial intelligence methods [3]. Electricity load forecasting is a process of predicting future load changes by analyzing historical load data [2]. Load forecasting can be carried out using conventional or artificial intelligence/machine learning (ML) based methods [4]. There are two types of load forecasting used for electricity distribution systems namely, spatial and temporal forecasting. Temporal forecasting means forecasting the electricity load for a specific supplier or collection of consumers for future related particular times like hours, days, based on the time horizon of prediction, load forecasting can be classified into four categories: long-term forecasting, medium-term forecasting, short-term forecasting, and ultra-short-term forecasting [2].

STLF has become vitally important to be considered, to guarantee safe dispatch scheduling, to enable high performance of the power systems, and to determine profits for shareholders and consumers [5]. Short-term load forecasts are required for the control and scheduling of power systems [6]. Hence, an accurate STLF model is essential to avoid unnecessary generation and operation cost increments [7]. For STLF, different methods have been developed such as time series, regression models, fuzzy logic, neural networks, etc. [5]. There are two main approaches to forecasting energy consumption: Conventional methods, and more recently, methods based on Machine learning [8]. Several machine learning or computational intelligence techniques have been applied in the field of Short-Term Load Forecasting [3].

Ensemble methods have been widely deployed for forecasting applications due to their ease of implementation [9] and are derived from seven individual machine learning models, which include random forest, among others [3]. ML-based intelligent methods can model nonlinear relationships [4], and new machine learning (ML) approaches are emerging, motivating the exploration to update the forecasting tools with the most efficient and robust methods to minimize errors. This paper is aimed at improving STLF through ensemble stacking generalization with Ann which has excellent performance, accuracy and flexibility.

In the related areas, recently many researchers have carried out various works in the subject matter to mention but a few among others. Xin *et al.*, [2] proposed electricity load forecasting as a process of predicting future load changes by analyzing historical load data. Jihoon *et al* [10] presented and discussed stable power supply and management of power infrastructure. Ribeiro *et al.*, [11] explored short-Term Load Forecasting as critical for reliable power system operation, and the search for enhanced methodologies has been a constant field of investigation, particularly in an increasingly competitive environment where the market operator and its participants need to better inform their decisions. Khawaja *et al.*, [4] used artificial neural networks (ANNs) based ensemble machine learning for improving short-term electricity load forecasting. Sulandari *et al.*, [12] proposed electricity play a key role in human life. Ribeiro *et al.*, [11] explored short-Term Load Forecasting as critical for reliable power system operation, and the search for enhanced methodologies has been a constant field of investigation, particularly in an increasingly competitive environment where the market operator and its participants need to better inform their decisions. Massaoudi *et al.*, [13] and Aguilar & Antonio [14] proposed an effective computing framework for Short-Term Load Forecasting (STLF) and planning committees. Load forecasting is the underpinning of control scheme procedure and scheduling. Accurate load forecasting can secure the safe and reliable operation of the power system, cut power generation costs, and increase economic benefits [15]. Sun *et al.*, [16], transient stability prediction is critically essential to the fast online assessment and maintaining the stable operation in power systems. The phasor measurement units (PMUs) help the progress of data-driven methods for momentary strength valuation. The wind power industry has called for precise and steady wind speed guessing, on which dependable wind power cohort systems depend greatly [17]. Miguel *et al.*, [18] proposed electricity load forecasting as an essential tool for effective power grid operation and energy markets. Electricity demand forecasting has been a real challenge for power system scheduling in different levels of energy sectors [19]. Yang *et al.*, [20] proposed short-term load forecasting (STLF) in improving the economy and security of electric system operations. Fathi *et al.*, [3] load forecasting models are of great importance in Electricity Markets and a wide range of techniques have been developed according to the objective being pursued.

1.2 Short Term Load Forecasting

Forecasts are required for proper scheduling activities, such as generation scheduling, fuel purchasing scheduling, maintenance scheduling, investment schedule, and for security analysis [21]. Load forecasting may be defined as the measure of exactness of the difference between the actual and predicted value of future load demand Baliyan *et al*, [1], and an estimation of how much electricity is needed in the future. STLF is a load forecasting scheme with a foremost time of one hour to several days, which is essential for acceptable preparation and process of power systems. It is vital also to understand the general trend of researchers' interest in searching and investigating the time progress of developed models of electricity load forecasting over time to improve the existing results and applications [22]. Figure 1 covered the period from 2003 to 2019 publications papers presented so far on STLF on electricity.

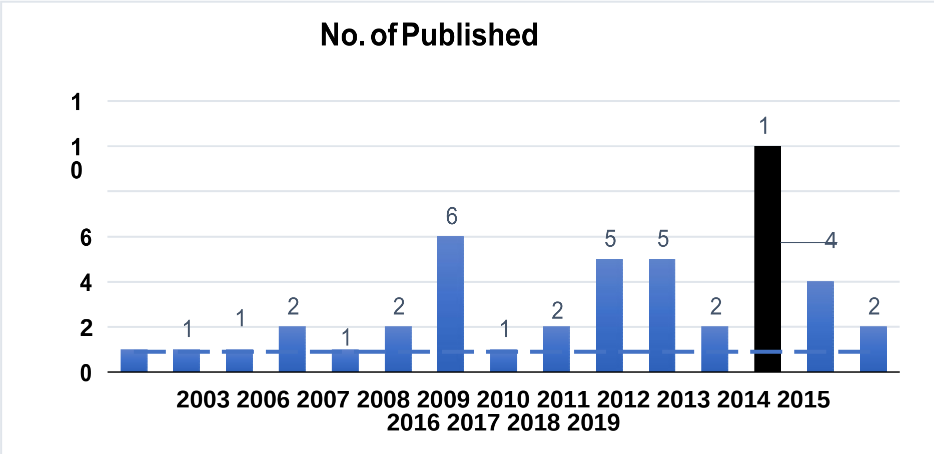


Figure 1. The Publication Pattern in the Field of Electric Load Forecasting [22]

The forecasting horizon distribution through the different reviewed papers is shown in Table 1. The results reveal that short-term and long-term predictions have contributed to the highest percentage within the reviewed papers by 44.4% and 22.2% respectively applications [22]. In comparison, very short-term and mid-term guesses are not exceedingly signified inside the belongings.

Table 1: The forecasting horizon sharing over the studied papers

Time Frame	Number of Papers (Journal & Conference)	Distribution Percentage
Very Short-Term	1	2.22%
Short-Term	20	44.44%
Mid-Term	5	11.11%
Long-Term	10	22.22%
None	9	20 %
Total	45	---

Figure 2 presents a different analysis of the forecasting model. A clear orientation is observed in the use of forecasting models [22]. The ANN is the most widely used and is followed by the regression model as shown in Figure 2 (27 and 19 papers respectively.

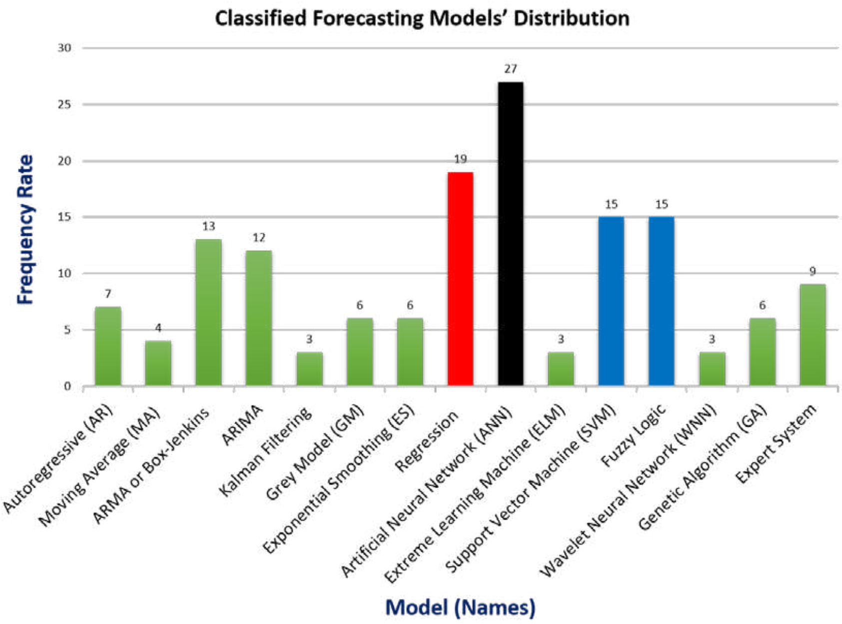


Figure 2. The distribution of the different analyzed forecasting models [22].

The world energy demand is increasing day by day, it is estimated that the world energy consumption will increase from 549 quadrillion British thermal units (Btu) in 2012 to 629 quadrillion Btu in 2020, a further 48% increase (to 815 quadrillion Btu) is expected by 2040 [23]. Henceforward, dynamism companies must focus powerfully on load forecast and constraints to be presented on power booster into conduction lines and revolving reserves [24]. Figure 3 illustrates the tree graph of these four predicting approaches. As can be seen, each method can be carried out via multiple strategies [25]. There are various methods to forecast a hierarchical arrangement including bottom-up, top-down, ensemble, and weighted arrangement. The full explanation of these four known groups of STLF methodologies is presented in the following paragraphs with specimens of numerous case studies.

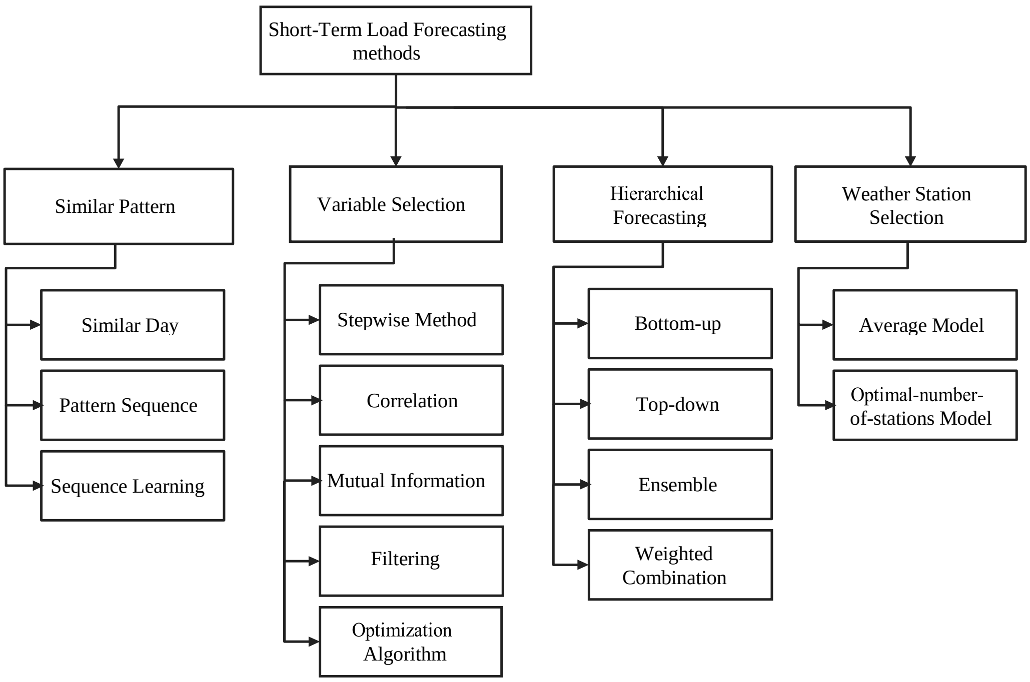


Figure 3. Tree diagram of the STLF methods [25].

The selection of a forecasting method relies on several factors including the relevance and availability of historical data, the forecast horizon, the level of accuracy for weather data, desired prediction accuracy, and so forth [25]. A large variety of methods and ideas have been tried for load forecasting. There are two main approaches to forecasting energy consumption, conventional methods and more recently, methods based on machine learning [24].

2. Machine Learning Methods

Machine learning (ML) is defined as 'the study of computer programs that leverage algorithms and statistical models to learn through inference and patterns without being explicitly programmed. At present, various types of ML algorithms are being used in different applications. ML techniques and other applications including '*supervised, semi-supervised, unsupervised, and reinforcement learning' will be detailed further in this subsection*. Furthermore, ML methods such as 'artificial neural networks (ANN), deep learning (DL), multi-layer perceptron (MLP), support vector machine (SVM), extreme learning machine (ELM), self-organizing map (SOM), decision tree (DT), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM) and abductive networks' have been utilized to predict STLF of Micro-grid. Machine learning (ML) is the practice of programming computers to learn from data [26]. In ML, different trained models can give different solutions and fail under different conditions [27]. Machine learning strategies, in contrast to traditional methods, are also suitable for non-linear cases. Machine learning algorithms are widely used in a variety of applications like digital image processing (image recognition), big data analysis, Speech Recognition, Medical Diagnosis, Statistical Arbitrage, Learning Associations, Classification, and Prediction etc, [28]. Figure 4 shows the machine learning algorithm technique.

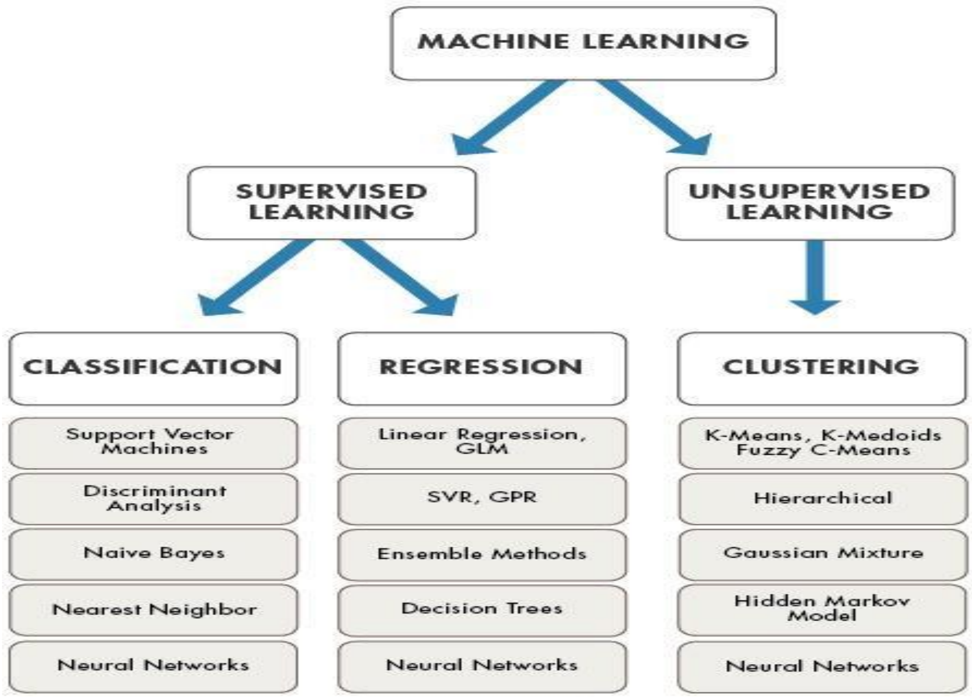


Figure 4. Machine learning algorithms [28].

Machine learning tasks are typically classified into three broad categories, depending on the nature of the learning "signal" or "feedback" available to a learning system. Neeraj *et al*, [28]: Supervised Learning; Unsupervised Learning and Reinforcement Learning.

2.1 Supervised Learning

Supervised learning means building a parametrized model that can divide the data domain, and then optimizing the parameters using training, validation, and testing algorithms [29].

Supervised learning is where you have input variables (X) and an output variable (Y) and you use an algorithm to learn the mapping function from the input to the output [30].

$$Y=f(X) \quad (1)$$

Supervised learning mechanisms by serving the machine sample data with numerous structures (represented as "X") and the precise value output of the data (represented as "Y"). (In this type of machine-learning system, the data that you feed into the algorithm, with the desired solution, are referred to as "labels." Figure 5 shows the supervised learning process.

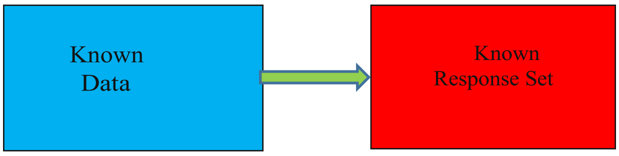


Figure 5: Supervised learning (models and algorithms)

As far as supervised learning is concerned, every example is a mainstay containing an input object (which is usually a vector quantity) and an enforced output value (may also be referred to as a supervisory signal) [31].
However, in big data, the data domain is controlled by the three common parameters: volume, variety, and velocity, and the modelling definition is presented [29]. Supervised learning has two main objectives: parameterization objectives and optimization objectives, and these objectives may be defined and differentiated using the continuous and discrete nature of the response variables [29]. Therefore, the objectives of supervised learning can be divided into the following four steps Suthaharan, [29] viz: (a). tuning model parameters, (b). generating algorithms for tuning, (c). improving the models to work with unseen data, and (d). applying efficient quantitative and qualitative measures for tuning. The process of applying supervised ML to a real-world problem is described [31]. Figure 6 shows the process of supervised ML

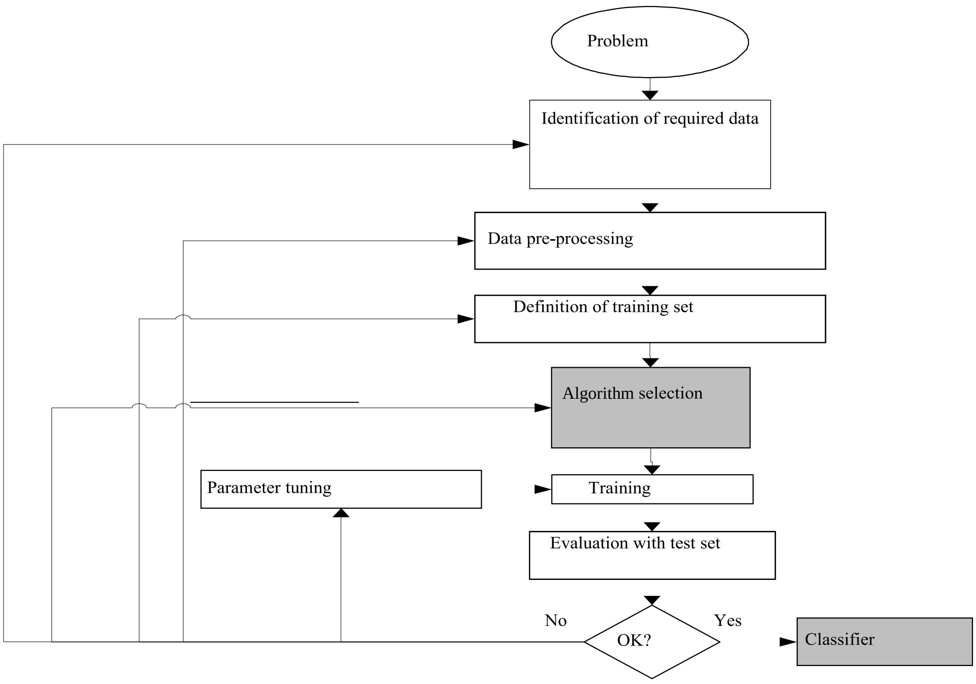


Figure 6. The process of supervised ML (Praveena *et al.,* [31]

2.2 Unsupervised Learning

Unsupervised learning is where you only have input data (X) and no corresponding output variables [30]. This is the ML task that gathers a function to depict buried structures from "unlabeled" data [31]. Since the examples specified to the learner are unlabeled, there is no assessment of the accuracy of the structure that is output by the relevant algorithm—which is one way of distinguishing unsupervised learning from supervised learning and reinforcement learning [31]. This is called unsupervised learning because unlike supervised learning above there are no correct answers and there is no teacher, and unsupervised learning problems can be further grouped into clustering and association problems [30].

2.3 Reinforcement Learning

A processor program relates to an animated setting in which it must execute a certain goal. The program is provided feedback in terms of rewards and punishments as it navigates its problem space [31]. The conditions wherever you have a huge volume of involvement data (X) and only some of the data is labelled (Y) are termed semi-supervised learning problems. These situations sit in between both supervised and unsupervised learning [30].

Similarly, one of the goals of machine learning is to build an intellectual arrangement. The two main components that can help machine learning approaches achieve this goal are learning models and learning algorithms [29]. Learning models (LM) and learning algorithms (LA) remain, in one way or the other, pattern acknowledgement apparatuses. Then the machine-learning problem may be defined as how to fit a model between them and how to train and validate the model to learn the system's characteristics from data [29]. Figure 7 shows demonstrates a data domain and response set:

A parameter B

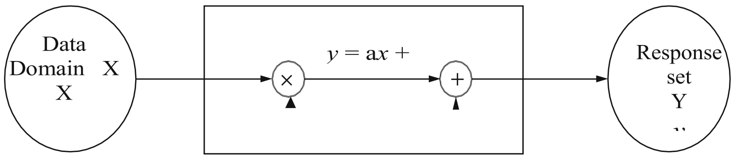


Figure 7: A linear (*straight line*) demonstrating between a data domain and a response set.

Load forecasting methods should have low variance and bias, which ensures consistent and accurate performance, respectively, this goal is achieved by combining a group of predictive models, known as ensemble learning [4].

4 Ensemble Learning

Load forecasting methods should have low variance and bias, which ensures consistent and accurate performance, respectively, this goal is achieved by combining a group of predictive model. An ensemble method is a Machine Learning concept in which the idea is to build a prediction model by combining a collection of "N" simpler base learners [3]. Ensemble methods have been successfully applied for solving pattern classification, regression and forecasting in time series problem models, known as ensemble learning [4]. These methods are designed to reduce bias and variance concerning a single-base learner [3]. Ensemble methods have been successfully applied for solving pattern classification, regression and forecasting in time series problem models, known as ensemble learning [4]. Ensemble learning is also called committee-based learning or learning multiple classifier systems [32]. The idea of ensemble learning is to train multiple models, each to predict or classify a set of results [33]. In a nutshell, ensemble learning is a procedure where multiple learner modules are applied to a data set to extract multiple predictions; such predictions are then combined into one composite prediction [23]. Usually, two phases are employed, in the first phase a set of base learners are obtained from training data, while in the second phase, the learners obtained in the first phase are combined to produce a unified prediction model [23]. Most ensemble methods use a single base learning algorithm to produce homogeneous base learners, i.e., learners of the same type, leading to homogeneous ensembles, but there are also some methods which use multiple learning algorithms to produce heterogeneous learners, i.e., learners of different types, leading to heterogeneous ensembles [32]. The oversimplification capability of an ensemble is frequently greatly tougher than that of improper learners. Ensemble methods are appealing mainly because they can boost weak learners which are even just slightly better than a random guess to strong learners who can make very accurate predictions [32]. Ensemble methods have become a major learning paradigm since the 1990s, with great promotion by two pieces of pioneering work [32]. One is empirical, in which it was originated that forecasts completed through the grouping of a conventional of classifiers are frequently further precise than forecasts finished by the greatest sole classifier. Ensembles of weak learners were mostly studied in the machine-learning community [32]. Generally, an ensemble is constructed in two steps, i.e., generating the base learners, and then combining them [32]. The concrete illustration of the ensemble architecture is shown in Figure 8, known as ensemble learning [4].

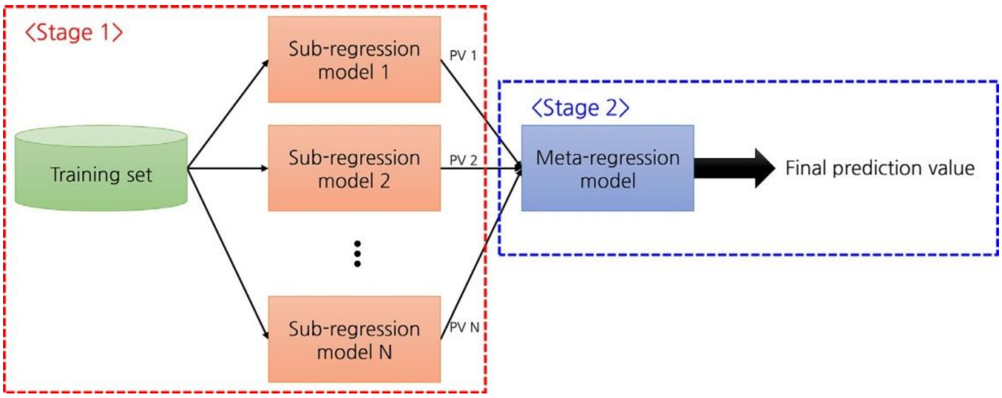


Figure 8. Conceptual diagram of the ensemble architecture

The most used and well–known of the basic ensemble methods are three (3) [23].
a) Bagging,
b) Boosting and,
c) Stacking.

4.1 Bagging Ensemble Method

In the case of bagging (bootstrap aggregating), the collection of "N" base learners to the ensemble is produced by bootstrap sampling on the training data [3]. The name Bagging came from the abbreviation of Bootstrap AGGregatING [32]. As the name implies, the two key ingredients of Bagging are bootstrap and aggregation [32]. The simplest approach with bagging is to use a couple of small subsamples and bag them, if the ensemble accuracy is much higher than the base models, it's working; if not, use larger subsamples Zhang *et al*, [33] of ensemble methods, that is, sequential ensemble methods where the base learners are generated sequentially, with AdaBoost as a representative, and parallel ensemble methods where the base learners are generated in parallel, with Bagging as are presentative [32]. Trendy precise, once we smear shooting to regression trees, a separately specific tree takes tall change, but short partiality. Averaging the resulting prediction of these N trees reduces the variance and substantially improves accuracy [3]. Taking approves the utmost general policies for combining the productivities of the base learners that is, elective for classification and be an average of for regression. To predict a test instance, taking classification for example, bagging feeds the instance to its base classifiers and collects all of their outputs, and then votes the labels and takes the winner label as the prediction, where ties are broken arbitrarily [32]. Figure 9 shows the algorithm of bagging as follows:

Input: Data set $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$;
Base learning algorithm $\mathfrak{L}$;
Number of base learners T.

Process:
1. **for** $t = 1, \ldots, T$:
2. $h_t = \mathfrak{L}(D, \mathcal{D}_{bs})$ % $\mathcal{D}_{bs}$ is the bootstrap distribution
3. **end**

Output: $H(\boldsymbol{x}) = \underset{y \in \mathcal{Y}}{\arg\max} \sum_{t=1}^{T} \mathbb{I}(h_t(\boldsymbol{x}) = y)$

Figure 9. The Bagging Algorithm [32]

4.2 Boosting Ensemble Method

The stretch boosting denotes a household of procedures that are talented to change weedy learners to robust learners. Intuitively, a weak learner is just slightly better than a random guess, while a strong learner is a very close to perfect performance [32]. Boosting is like bagging, but with one theoretical alteration. Instead of assigning equal weighting to models, boosting assigns different weights to classifiers, and derives its ultimate result based on weighted voting [23]. Figures 10 and 11 present the boosting algorithm and a common ensemble architecture respectively.

Input: Sample distribution D;
Base learning algorithm L;
Number of learning rounds T.
Process:
1.D $_1$=D.
%Initializedistribution2.**for**t
= 1,...,T:
3.h_t=L(D$_t$); %Train a weak learner from distribution D_t
4.$_t$=$P_{x\sim D_t}(h_t(x)f(x))$; %Evaluate the error of h_t
5.D$_{t+1}$=*Adjust Distribution*(D$_t$, $_t$)
6.**end**
Output:$H(x)$=*Combine Outputs*($\{h_1(x),\ldots,h_t(x)\}$)

Figure 10: The Boosting Algorithm [32].

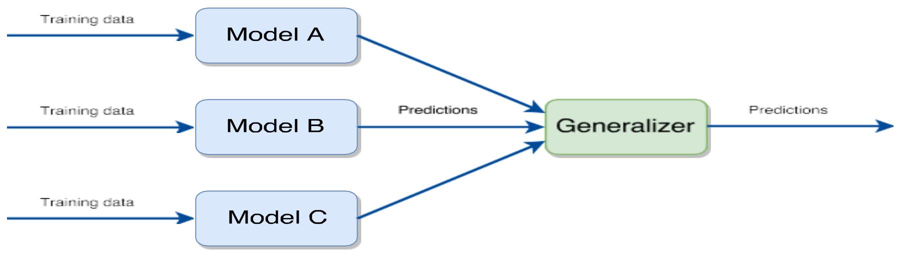


Figure 11. A common ensemble architecture [34]

4.3 Stacking Ensemble Method

Stacking is a general procedure where a learner is trained to combine the individual learners, [32]. Here, the individual learners are called the first-level learners, while the combiner is called the second-level learner, or meta-learner [32]. Stacking shapes its reproductions using diverse learning algorithms and then a combiner algorithm is skilled to make the final forecasts using the forecasts made by the base algorithms. This combiner can be any ensemble technique [23]. In stacking, the result of a set of different base learners at level-0 is combined with a meta-learner at level-1 [35]. The character of the meta learner is to determine how finest to associate the output of the base learners. The composition of Stacking is of two phases. In the first phase, diverse representations are learned and built on a dataset. Then, the output of each model is collected to create a new dataset [35]. In the novel dataset, a piece example is associated with the real value that it is made up to forecast. Secondly, that dataset is used with a learning algorithm, the so-called meta-learning algorithm, to provide the final output [36]. Figures 12 and 13 present the stacking algorithm and conceptual diagram of the stacking ensemble model respectively. **Input**: Dataset D = {(x1,y1),(x2,y2),...,(xm, ym)};

First-level learning algorithms L1,...,LT;

Second-level learning algorithm L.

Process:

1. **for** t =1,...,T: %Train a first-level learner by applying the
2. **ht** = Lt(D); %first-level learning algorithm Lt
3. **end**
4. D' = ∅; %Generate a new dataset
5. **for** i =1,...,m :
6. **for** t =1,...,T:
7. zit = ht(xi);
8. **end**
9. D' = D'∪((zi1,...,ziT),y i);
10. **end**
11. h'= L(D);

%Train the second-level learner hby applying

%the second-level learning algorithm L to the

%new data set D.

Output: H(x)=h'(h1(x),...,hT(x))

Figure 12. The Stacking Algorithm [32].

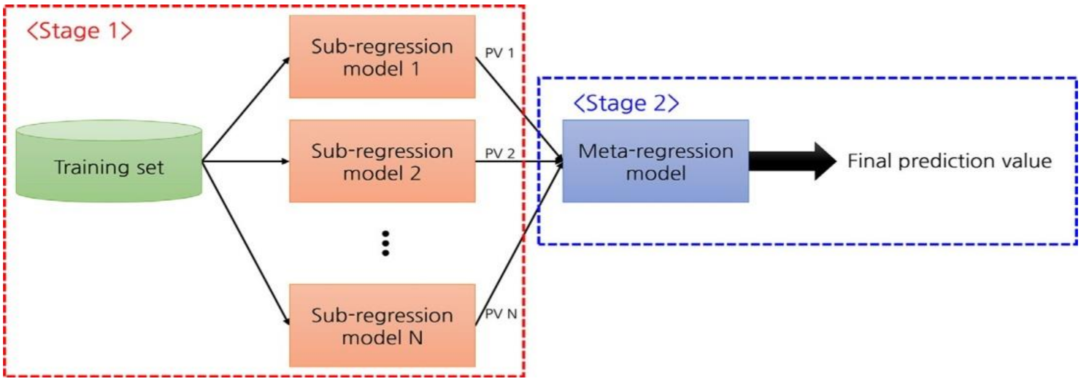


Figure 13. Conceptual diagram of the stacking ensemble model [37]

5. Model of Stacked Generalization of Load Forecasting System

The basic idea is to train the first-level learners using the original training data set, and then generate a new data set for training the second-level learner, where the outputs of the first-level learners are regarded as input features while the original labels are still regarded as labels of the new training data [32]. The first-level learners are often generated by applying different learning algorithms, and so, stacked ensembles are often heterogeneous, though it is also possible to construct homogeneous stacked ensembles [32]. In particular, three base learning methods were employed and formed (evolutionary algorithm, gradient boosting model and random forest) then the highest technique (Artificial Neural Networks). The straightforward knowledge approaches are regression trees based on Evolutionary Algorithms, Generalized Boosted Regression Models. On the highest equal, one must use the Artificial Neural Networks in demand to the association the forecasts formed by the lowest equal. Figure 14 shows components of the stacked generalization of the load forecasting system.

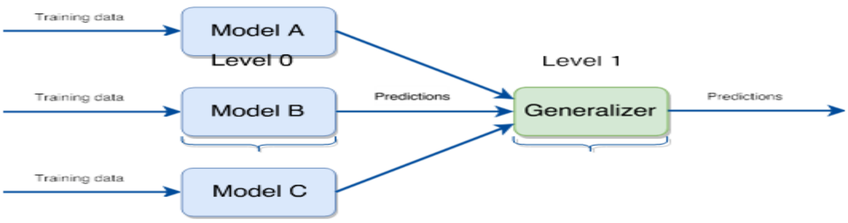


Figure 14: Components of Stacked Generalization of load forecasting system [25]

The employed scheme is graphically shown in the Figure 15:

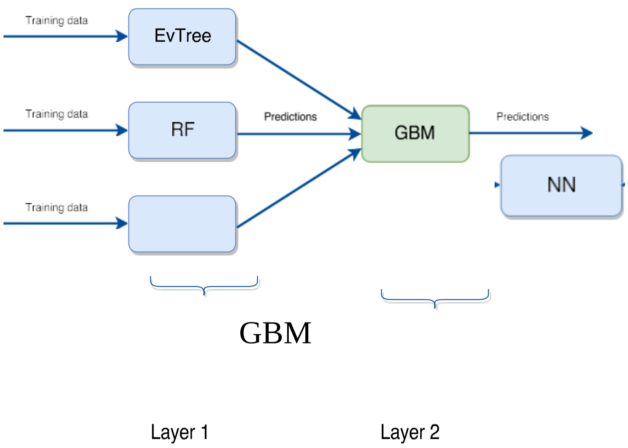
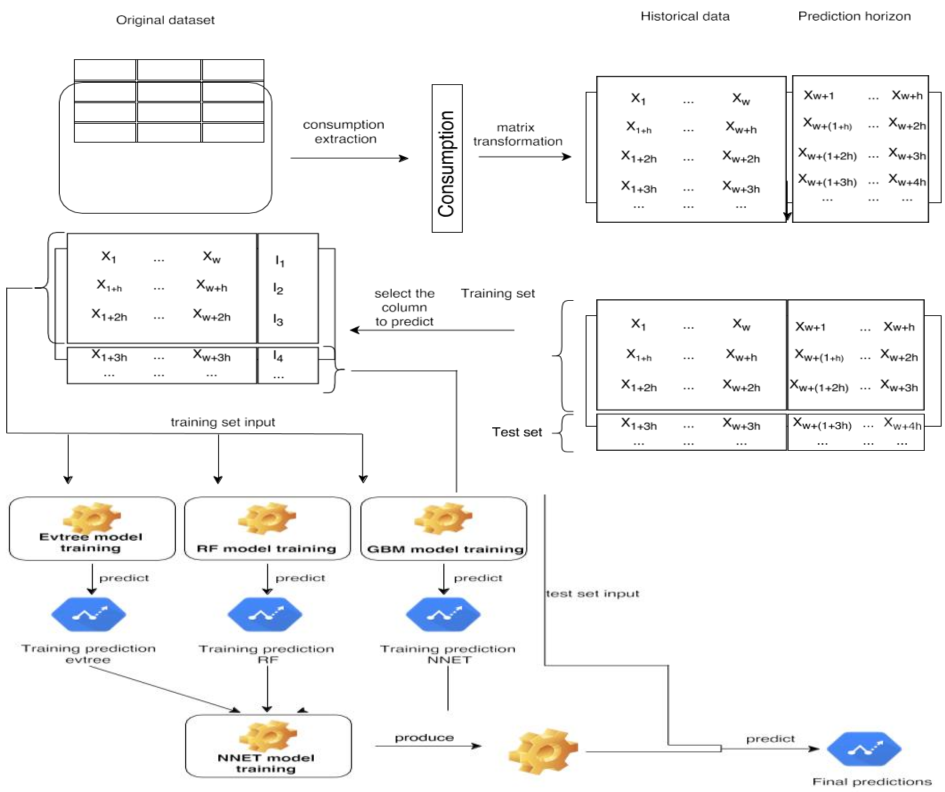


Figure 15. A graphical representation of the ensemble scheme [25]

The ending ensemble arrangement projected is shown in Figure 16. The exercise set is second-hand in the direction to get the forecasts of the improper close, comprising of RF, GBM and EVTree. The gotten forecasts are then used through the upper layer (ANN) to yield the final forecasts for respectively problematic.



(ANN Model)

Figure 16: A scheme of the ensemble learning strategy used, *w* determines the size of the historical window used, while *h* determines the prediction horizon [38].

6. Evolutionary Algorithms (Eas) For Regression Trees

EAs remain population-based policies that use procedures enthused by evolutionary natural science such as legacy, transformation, assortment a border. Each distinct I of the inhabitants signifies an applicant's resolution to a given problem and is allocated a suitability purpose, which is an amount of the superiority of the answer signified by i. Characteristically EAs jump after an early populace containing arbitrarily reset entities. Individually is estimated to regulate its suitability worth. Formerly a collection device is used to select several individuals. Generally, the choice is based on fitness, so fitter entities have more likelihood of being selected. Selected entities produce descendants, i.e., novel resolutions, using the request of crossover and mutation operatives. This procedure is reiterated over several peers or until a respectable sufficient answer is established. The knowledge is that improved and better results will be established at each generation. Furthermore, the use of stochastic machinists, such as mutation, allows EAs to discharge after local goals. For the difficulty undertaken in this paper, each separate encodes a regression tree. A regression tree is a decision response variable Y by a vector of P predictor variables $X = (X1, XP)$ [39]. It is not much different; regression tree is similar to a classification tree. Both classification and regression trees aim at modelling a for classification trees, Y is qualitative and for regression trees Y is quantitative. In mutual cases, Xi can be constant and/or definite variables. Regression trees remain usually used in regression-type problems, where we attempt to forecast the values of a constant variable from one or additional constant and/or definite forecaster variables. A benefit of using regression trees is that outcomes can be easier to understand. Additional materialistic approaches have been used to attain regression trees, for example. The principal experiment of such approaches is that the exploration space is usually huge, rendering full-grid searches computationally infeasible Isberg, [40]. Due to their search competencies, EAs have proven that they can overwhelm this constraint.

7. Random Forests (Rf)

Random Forest was familiarized by Breinman and Cutle, then denotes a set of resolution trees which procedure an ensemble of forecasters. Thus, RF is an ensemble of decision trees, where each tree is trained separately on an independent randomly selected training set Liu *et al.,* [41] and [42]. The aforementioned tails that each tree be contingent on the standards of an input dataset experimented autonomously, with a similar supply for all trees. Similarly, the trees produced are changed since they are gotten from different drill sets from bootstrap subsampling and dissimilar random subsets of structures to splitting at each tree node. Individually tree is completely full-grown, to achieve low-bias trees. Furthermore, at the identical time, the arbitrary subsets of structures outcome in a low association between the separate trees, so the algorithm yields an ensemble that can attain both, low bias and low variance. Instead of arrangement, an individual tree in the RF casts a component ballot for the greatest standard class at the input. The final result of the classifier is determined by a majority vote of the trees. Instead of regression, the ultimate forecast is typical of the forecasts from the set of choice trees. The technique is less computationally exclusive than others tree-based classifiers that accept capturing plans since each tree is produced by taking into justification only a portion of the feedback topographies.

8. Generalized Boosted Regression (Gbr)

GBR technique iteratively sequences a set of choice trees. Gradient boosting involves three elements [44]:

1. A loss function to be optimized. Such a function is problem dependent. For instance, for regression, a squared error can be used and for classification, we could use logarithmic loss.
2. A weak learner to make predictions. Regression trees are used for this aim and a greedy strategy is used to build such trees. This strategy is based on using a scoring function used each time a split point has to be added to the tree. Other strategies are commonly adopted to constrain the trees. For example, one may limit the depth of the tree, the number of splits or the number of nodes.
3. An additive model to add trees to minimize the loss function. This is done sequentially, and the trees already contained in the model built so far are not changed. To minimize the loss during this phase, a gradient descent procedure is used. The procedure stops when a maximum number of trees has been added to the model or once there is no improvement in the model.

Overfitting is mutual in gradient boosting, and usually, some regularization approaches are recycled to decrease it. These approaches essentially penalize numerous parts of the algorithm. Typically some devices are used to execute restrictions on the assembly of decision trees, for instance, limit the distance of the trees, the number of nodes or leaves or the quantity of observation per split. Another mechanism is shrinkage, which is weighting the contribution of each tree to the sequential sum of the predictions of the trees [23]. This is complete to decelerate the learning rate of the algorithm. As a consequence, the training takes longer, since more trees are added to the model. In this way, a trade-off between the learning rate and the number of trees can be reached.

9. Artificial Neural Networks (Anns)

ANNs are computational models inspired by the structure and functions of biological neural networks [25]. The basic unit of computation is the neuron, also called the node, which receives input from other nodes or an external source and computes an output [25]. Naturally enough, a network of neurons is the composition of the nonlinear functions of two or more neurons [45]. In direction to calculate such productivity, the node applies a utility f called the *Activation Function*, which has the drive of presenting non-linearity into the output. Furthermore, the output is produced

only if the inputs are above a certain threshold [25]. Essentially, an ANN makes an association between input and output values and is collected of consistent nodes gathered in numerous layers. Amongst such layers can differentiate the outer ones, called input and output layers, from the "internal" ones, called hidden layers. Neural networks come in two classes: feedforward networks and recurrent (or feedback) networks [45]. A feedforward neural network is a nonlinear function of its inputs, which is the composition of the functions of its neurons [45]. The most general neural network architecture: recurrent neural networks, whose connection graph exhibits cycles [45]. In that graph, there exists at least one path that, following the connections, leads back to the starting vertex (neuron); such a path is called a cycle [45]. Since the output of a neuron cannot be a function of itself, such an architecture requires that time be explicitly taken into account: the output of a neuron cannot be a function of itself at the same instant of time, but it can be a function of its past value(s) [45].
In contrast to biological neuron networks, ANNs usually consider only one type of node, to simplify the model calculation and analysis [25]. The intensity of the connection between nodes is determined by weights, which are modified during the learning process [25]. Therefore, the learning process consists in adapting the connections to the data structure that model the environment and characterizes its relations [25]. Conferring to the assembly, there are dissimilar types of ANN. The fittingness of the construction is contingent on numerous aspects, for instance, the quality and the volume of the input data. The modest type of ANN is called a 'feed-forward neural network'. In such networks, nodes from adjacent layers are interconnected and each connection has a weight associated with it [25]. The data moves accelerative from the input to the output layer over the concealed nodes. There is only one node at the output layers, which provides the final results of the network, being it a class label or a numeric value [19]. The present ensemble of trees is recycled to forecast the value of the individual training sample. The forecast mistakes are then projected, and deprived forecasts are attuned so that in the following repetitions the preceding errors are modified. Figure 17 shows a modest neuron arrangement in an ANN.

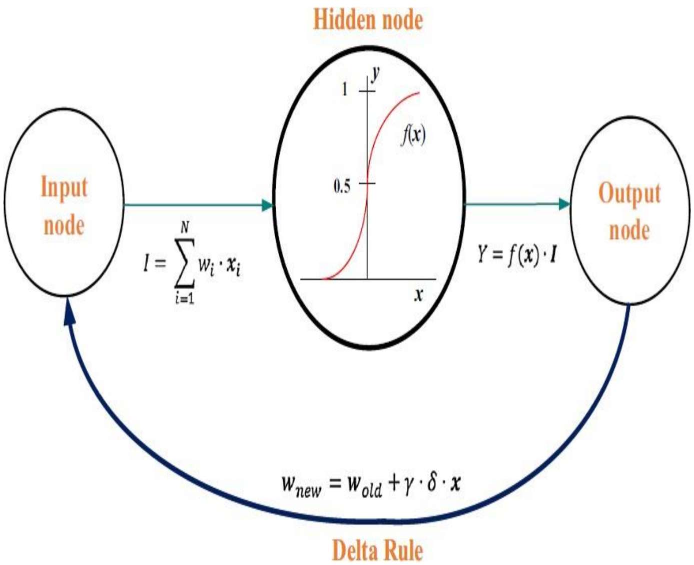


Figure 17: A simple neuron scheme in an ANN [22].

10. Evaluation Measure
This subset defines numerous features of the evaluation of the dissimilar models; the evaluation characteristics include the assessment of fault rates and pairwise contrasts of classifiers/ensembles.
10.1 Performance Evaluation

To assess the performances of both the ensemble scheme and the base methods, we used five measures commonly used in regression: the mean relative error (MRE), the mean absolute error (MAE), the symmetric mean absolute percentage error (SMAPE), the coefficient of determination *R*2, and the root mean squared error (RMSE), which are defined (Fallah *et al.*, [25] as in equations (2-6), and were used as criteria for error evaluation to analyze model prediction performance.

$$MRE = \frac{1}{n}\sum \frac{|Y - Y'|}{Y} \qquad (2)$$

$$MAE = \frac{1}{n}\sum |Y - Y'| \qquad (3)$$

$$sMAPE = \frac{1}{n}\sum \frac{2\,|Y-Y'|}{|Y+Y'|} \qquad (4)$$

$$R^2 = 1 - \sum \frac{(Y_i - Y')^2}{|Y_i + Y^-{}_i|} \qquad (5)$$

$$RSME = \sqrt{\frac{1}{n}} \sum (Y_i - Y_i^-)^2 \qquad (6)$$

11. Conclusion

Load forecasting shows an important character in the controlling of power systems, and it can boot out the restrictions produced by the deficiency of electricity. This paper presented a broad appraisal of the ensemble model with numerous depictions to predict the short-term electricity load. Subsequently studying the obtainable models and their consequences, the finding achieves that most of the models have been used for estimating energy demand and electrical load. Measurable methods include statistical analysis and predict the future based on the material of past data. The AI method was favourite more normally than the quantitative one for predicting the electric load and renewable energies. Though, methods extended to help forecast energy demand are significantly more effective than those formed by AI. Investors can use LF models to accomplish the significance of prevailing and potential energy policy. The application of well-organized forecasting approaches to maintain and deliver sector predictions is simple to the accuracy of STLFs. Numerous variables choose the best applied to predict method. Unique of the best dangerous causes of technique choice is the range and nature of the research issue and the objective of the study. In prospective research work, diverse deep learning approaches ought to be examined to influence their predicting benefits.

Reference

[1] Baliyan, A., Gaurav, K., & Mishra, S. K. (2015). A review of short term load forecasting using artificial neural network models. Conf. Procedia Computer Science, 48, 121-125.
[2] Gao, X., Li, X., Zhao, B., Ji, W., Jing, X., & He, Y. (2019). Short-term electricity load forecasting model based on EMD-GRU with feature selection. Energies, Vol. 12(6), 1140.
[3] Fathi, S., Srinivasan, R., Fenner, A., & Fathi, S. (2020). Machine learning applications in urban building energy performance forecasting: A systematic review. Renewable and Sustainable Energy Reviews, 133(2020), p110287.
[4] Khwaja, A. S., Anpalagan, A., Naeem, M., & Venkatesh, B. (2020). Joint bagged- boosted artificial neural networks: Using ensemble machine learning to improve short-term electricity load forecasting. Electric Power System Research, 179(2020), p106080.
[5] Khuntia, S. R., Rueda, J. L., & van Der Meijden, M. A. (2016). Forecasting the load of electrical power systems in mid-and long-term horizons: a review. IET Generation, Transmission & Distribution, Vol. 10(16), 3971-3977.
[6] Taylor, J. W. (2008). An evaluation of methods for very short-term load forecasting using minute-by-minute British data. International journal of forecasting, Vol. 24(4), 645-658.

[7] López, C., Zhong, W., & Zheng, M. (2017). Short-term electric load forecasting based on wavelet neural network, particle swarm optimization and ensemble empirical mode decomposition. Energy Procedia, 105(2017), 3677-3682.

[8] Wang, J., Zhang, L., & Li, Z. (2022). Interval forecasting system for electricity load based on data pre-processing strategy and multi-objective optimization algorithm. Applied Energy, 305(2022), 117911.

[9] Shen, Y., Yang, F., Habibullah, M. S., Ahmed, J., Das, A. K., Zhou, Y., & Ho, C. L. (2021). Predicting tool wear size across multi-cutting conditions using advanced machine learning techniques. Journal of Intelligent Manufacturing,Vol. 32(6), 1753-1766.

[10] Jihoon, Yongsung, Minjae and Eenjun (2018). Hybrid Short-Term Load Forecasting Scheme Using Random Forest and Multilayer Perceptron. Energies, Vol.11(12), 3283.

[11] Ribeiro, A. M. N., do Carmo, P. R. X., Endo, P. T., Rosati, P., & Lynn, T. (2022). Short-and Very Short-Term Firm-Level Load Forecasting for Warehouses: A Comparison of Machine Learning and Deep Learning Models. Energies, Vol. 15(3), 750.

[12] Sulandari, W., Lee, M. H., & Rodrigues, P. C. (2020). Indonesian electricity load forecasting using singular spectrum analysis, fuzzy systems and neural networks. Energy, 190(2020), 116408.

[13] Massaoudi, M., Refaat, S. S., Chihi, I., Trabelsi, M., Oueslati, F. S., & Abu-Rub, H. (2021). A novel stacked generalization ensemble-based hybrid LGBM-XGB-MLP model for Short-Term Load Forecasting. Energy, 214(2021), 118874.

[14] Aguilar Madrid, E., & Antonio, N. (2021). Short-term electricity load forecasting with machine learning information, Vol.12(2), 50

[15] Tang, X., Chen, H., Xiang, W., Yang, J., & Zou, M. (2022). Short-Term Load Forecasting Using Channel and Temporal Attention Based Temporal Convolutional Network. Electric Power Systems Research, 205(2022), 107761.

[16] Sun, P., Huo, L., Liang, S., & Chen, X. (2022). Fast Transient Stability Prediction Using Grid-informed Temporal and Topological Embedding Deep Neural Network. arXiv preprint arXiv:2201.09245.

[17] Li, D., Jiang, F., Chen, M., & Qian, T. (2022). Multi-step-ahead wind speed forecasting based on a hybrid decomposition method and temporal convolutional networks. Energy, 238(2022), 121981.

[18] López, M., Sans, C., Valero, S., & Senabre, C. (2018). Empirical comparison of neural network and auto-regressive models in short-term load forecasting. Energies, Vol. 11(8), 2080.

[19] Fan, S., & Hyndman, R. J. (2011). Short-term load forecasting based on a semi-parametric additive model. IEEE transactions on power systems,Vol. 27(1), 134-141.

[20] Yang, L., & H. (2019). Analysis of different neural networks and a new architecture for short-term load forecasting. Energies,Vol. 12(8), 1433.

[21] Guo, H., Davidson, M. R., Chen, Q., Zhang, D., Jiang, N., Xia, Q., ... & Zhang, X. (2020). Power market reform in China: Motivations, progress, and recommendations. Energy Policy, 145(2020), 111717.

[22] Hammad, M/ A., Jereb, B., Rosi, B., & Dragan, D. (2020). Methods and models for electric load forecasting: A comprehensive review. Logist. Sustain. Transp, Vol.11(1), 51-76.

[23] Divina, F., Gilson, A., Goméz-Vela, F., García Torres, M., & Torres, J. F. (2018). Stacking ensemble learning for short-term electricity consumption forecasting. Energies, Vol.11(4), 949.

[24] He, L., Lu, Z., Zhang, J., Geng, L., Cai, Y., & Li, X. (2021). Economic dispatch of multi-area integrated electricity and natural gas systems considering emission and hourly spinning reserve constraints. International Journal of Electrical Power & Energy Systems, 132(2021), 107177.

[25] Fallah, S. N., Ganjkhani, M., Shamshirband, S., & Chau, K. W. (2019). Computational intelligence on short-term load forecasting: A methodological overview. Energies, Vol.12(3), 393

[26] Dimiduk, D. M., Holm, E. A., & Niezgoda, S. R. (2018). Perspectives on the impact of machine learning, deep learning, and artificial intelligence on materials, processes, and structures engineering. Integrating Materials and Manufacturing Innovation, Vol.7(3), 157-172.

[27] Khwaja, A. S., Zhang, X., Anpalagan, A., & Venkatesh, B. (2018). Boosted neural networks for improved short-term electric load forecasting. Electric Power Systems Research, 143(2018), 431-437.

[28] Neeraj, K. N., & Maurya, V. (2020). A review on machine learning (feature selection, classification and clustering) approaches of big data mining in different area of research. Journal of Critical Reviews,Vol. 7(19), 2610-2626.

[29] Suthaharan, S. (2016). Decision tree learning. In Machine Learning Models and Algorithms for Big Data Classification 36(2016), 237-269. Springer, Boston, MA.

[30] Gygli, M., Norouzi, M., & Angelova, A. (2017, July). Deep value networks learn to evaluate and iteratively refine structured outputs. In International Conference on Machine Learning (pp. 1341-1351). PMLR.

[31] Praveena, M., & Jaiganesh, V. (2017). A literature review on supervised machine learning algorithms and boosting process. International Journal of Computer Applications, Vol. 169(8), 32-35.

[32] Li, M., Zhang, H., Wu, R., & Zhou, Z. H. (2012). Sample-based software defect prediction with active and semi-supervised learning. Automated Software Engineering, Vol.19(2), 201-230.

[33] Zhang, Y., Miao, D., Wang, J., & Zhang, Z. (2019). A cost-sensitive three-way combination technique for ensemble learning in sentiment classification. International Journal of Approximate Reasoning, 105(2019), 85-97.

[34] Ullah, I., & Mahmoud, Q. H. (2020, May). A scheme for generating a dataset for anomalous activity detection in iot networks. In Canadian Conference on Artificial Intelligence, Proceedings 33, (pp. 508-520). Springer, Cham.

[35] Patil, P. R., & Sivagami, M. (2020). Forest cover classification using stacking of ensemble learning and neural networks. In Artificial Intelligence and Evolutionary Computations in Engineering Systems (pp. 89-102). Springer, Singapore.

[36] Nicosia, G., Pardalos, P., Umeton, R., Giuffrida, G., & Sciacca, V. (Eds.). (2020). Machine Learning, Optimization, and Data Science: 5th International Conference, LOD 2019, Siena, Italy, September 10–13, 2019, Proceedings (Vol. 11943). Springer Nature.

[37] Moon, J., Kim, J., Kang, P., & Hwang E. (2020). Solving the cold-start problem in short-term load forecasting using tree-based methods. Energies. Vol.13(4):886.

[38] Ayala, J., García-Torres, M., Noguera, J. L. V., Gómez-Vela, F., & Divina, F. (2021). Technical analysis strategy optimization using a machine learning approach in stock market indices. Knowledge-Based Systems, 225(2021), 107119.

[39] Bartz-Beielstein, T., Branke, J., Mehnen, J., & Mersmann, O. (2014). Evolutionary algorithms. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery,Vol. 4(3), 178-195.

[40] Isberg, S. (2022). Design and analysis of computer experiments: large datasets and multi-model ensembles (Doctoral dissertation, University of British Columbia).

[41] Liu, M., Wang, M., Wang, J., & Li, D. (2013). Comparison of random forest, support vector machine and back propagation neural network for electronic tongue data classification: Application to the recognition of orange beverage and Chinese vinegar. Sensors and Actuators B: Chemical, 177(2013), 970-980.

[42] Li, Y., Yan, C., Liu, W., & Li, M. (2018). A principle component analysis-based random forest with the potential nearest neighbor method for automobile insurance fraud identification. Applied Soft Computing, 70(2018), 1000-1009.

[43] Benbelkacem, S., & Atmani, B. (2019, April). Random forests for diabetes diagnosis. In 2019 International Conference on Computer and Information Sciences (ICCIS) (pp. 1-4). IEEE.

[44] Zounemat-Kermani, M., Batelaan, O., Fadaee, M., & Hinkelmann, R. (2021). Ensemble machine learning paradigms in hydrology: A review. Journal of Hydrology, 598(2021), 126266.

[45] Rostami, Z., & Jafari, S. (2018). Defects formation and spiral waves in a network of neurons in presence of electromagnetic induction. Cognitive Neurodynamics, Vol.12(2), 235-254.