IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 26 July 2024, accepted 6 August 2024, date of publication 8 August 2024, date of current version 21 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3440631

SURVEY

Short-Term Load Forecasting: A Comprehensive Review and Simulation Study With CNN-LSTM Hybrids Approach

KALEEM ULLAH[1], MUHAMMAD AHSAN[2], SYED MUHAMMAD HASANAT[1], MUHAMMAD HARIS[3], HAMZA YOUSAF[4], SYED FARAZ RAZA[2], RITESH TANDON[5], SAMAIN ABID[2], AND ZAHID ULLAH[6], (Graduate Student Member, IEEE)

[1]Centre for Advanced Studies in Energy, University of Engineering and Technology, Peshawar (UET Peshawar), Peshawar 25000, Pakistan
[2]Department of Computer Science, University of Alabama at Birmingham, Birmingham, AL 35294, USA
[3]Institute for Insight, J. Mack Robinson College of Business, Georgia State University, Atlanta, GA 30303, USA
[4]Stetson-Hatcher School of Business, Mercer University, Macon, GA 30341, USA
[5]School of Informatics, Computing, and Engineering, Indiana University, Bloomington, IN 47405, USA
[6]Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, Italy

Corresponding author: Zahid Ullah (zahid.ullah@polimi.it)

This work was supported by the Politecnico di Milano for providing Open Access within the CRUI CARE Agreement.

ABSTRACT Short-term load forecasting (STLF) is vital in effectively managing the reserve requirement in modern power grids. Subsequently, it supports the grid operator in making effective and economical decisions during the power balancing operation. Therefore, this study comprehensively reviews STLF methods, including time series analysis, regression-based frameworks, artificial neural networks (ANNs), and hybrid models that employ different forecasting approaches. Detailed mathematical and graphical analyses and a comparative evaluation of these methods are provided, highlighting their advantages and disadvantages. Further, the study proposes a hybrid CNN-LSTM model comprised of Convolutional neural networks (CNN) for feature extraction of high dimensional data and Long short-term memory (LSTM) networks to boost the model's efficiency for temporal sequence prediction. This study assessed the model using a comprehensive dataset from Pakistan's NTDC national grid. The analysis revealed superior performance in short-term load prediction, achieving enhanced accuracy. For single-step forecasting, the model yielded an RMSE of 538.71, MAE of 371.97, and MAPE of 2.72. In 24-hour forecasting, it achieved an RMSE of 951.94, MAE of 656.35, and MAPE of 4.72 on the NTDC dataset. Moreover, the model has outperformed previous models in comparison using the AEP dataset, demonstrating its superiority in enhancing reserve management and balancing supply and demand in modern electricity networks.

INDEX TERMS Short-term load forecasting, convolution neural network, hybrid LSTM-CNN network, NTDC Pakistan, power balancing operation.

I. INTRODUCTION

Forecasting load demand is central to power system management and planning; thus, the accuracy of available demand forecasting models is crucial for effectively managing power grids. From this perspective, short-term load forecasting (STLF) enables the proper management of energy resources and the accurate control of energy storage. The accurate and timely predictions of possible future load variations by STLF also enhance the work of utilizing the available resources, hence improving the overall functionality of the systems [1]. STLF constitutes a fundamental element in energy management [2]. These models are made for forecasting power requirements for the short-term timeframe ranging from an hour ahead to days or weeks at most but typically up to a maximum of one month only. These models are invaluable for utility companies and grid operators as they help in planning the resources using better foresight to manage the responses to the ever-changing energy requirements that ensure the stability and availability of the power supply systems. However,

The associate editor coordinating the review of this manuscript and approving it for publication was Akshay Kumar Saha.

in the progress made before the big data era, various STLF methodologies have appeared in the literature, each with inherent features and limitations [3]. These frameworks cover a spectrum of computation methods, including artificial neural networks (ANNs), time series analysis, regression-based methods, and a combination of both (hybrid systems). The time series methods utilize historical load information using variant statistical tools, including ARIMA models and different types of exponential smoothing, to identify and then temporally analyze load trends to improve load forecasts' accuracy [4]. Contrarily, ANNs are a complex set of machine learning structures capable of learning input-output mapping that otherwise has a non-linear relationship. These models are trained across various data set assemblages using backpropagation networks, enabling them to adapt to information complex formation [5]. Regression-based models utilize linear or non-linear regression techniques to map causal associations with load and other features such as humidity, temperature, and time of day [6].

Contrarily, integrated models blend the strengths of different modeling approaches, such as ANNs and time series analysis, with the explicit intention of improving accuracy and increasing resistance to interference [7]. Though, each class of model has its unique difficulties and oddities. For example, in time series models, challenges often arise in explaining enduring trends and temporary fluctuations, load data in particular, reacting more sharply to anomalies and gaps in information. However, there are drawbacks associated with the computational complexity of ANNs, which subsequently decreases their levels of interpretability; this is coupled with their inherent tendency to fit over. Further, ANNs suffer from noise and bias in training data and are also sensitive to variations [8]. A major drawback of the regression-based models is that they are generally restricted to identifying linear and additive relations within input/output data. In addition, the models are not accustomed to handling categorical or non-numerical data. These challenges threaten the model as it cannot effectively employ graph structures to represent or analyze complex data [9].

Alternatively, some drawbacks of hybrid models include the contingency of the Extended Model with increased model complexity and an increased number of parameters, issues in parameter tuning, and primary issues of overfitting and data redundancy of the model. In this regard, the necessary steps must be taken to tackle these issues by comparing and assessing the different STLF models. This should be done regarding some of the actual load data that has been evaluated and verified using a wide range of measures and methods for validation purposes to understand their efficiency and reliability better. This way, considering one's advantages and challenges makes it possible to evaluate and compare various model types and assess their applicability in sustainably addressing the complexity of load forecasting in practical operational environments. Given the current fast-paced development in AI and data processing methods, there is an ongoing need to engage the sources that present the most contemporary syntheses of STLF models. These updates are important in keeping up with the current technological advancements and improving the prediction factors. To address this research gap, this review manuscript discusses the preceding points and explores advanced or relatively new STLF models that the earlier reviews may not have captured well. It also proposes a novel integration approach of LSTM for time series analysis and CNN for features from high-dimension data to enhance precision in prediction. By evaluating the result of the model, the NTDC Pakistan National Grid offered a data set for the study.

A. LITERATURE SURVEY

The construction of STLF models is a challenging and complicated activity because it has to deal with various concerns connected to the dynamic features of load data. These data are time-dependent and affected by factors such as the climate state, economic characteristics, and buyers' behaviour, whether they are individuals or companies [10]. Consequently, there is a need for STLF models to incorporate the skills that allow them to learn these dynamic data generation processes. Apart from controlling transient information, dealing with uncertainties and risks poses major challenges [11]. Other conventional approaches offer some forecast, more often, just a point estimation while failing to incorporate corresponding variability or risk. Tremendous efforts need to be made to bridge this divide and implement more advanced models that provide probabilistic forecasts, as demonstrated by the models that utilize the quantile regression approach. These reports employ advanced methodologies as a way of addressing issues of variability that are often associated with the data.

Moreover, incorporating wind and solar power as a part of renewable energy also adds fluctuations and randomness to the power supply, negatively impacting the precision and efficiency of STLF models [12]. Similar challenges relate to data characteristics in studying current trends; the data field is high-dimensional, and various input variables such as weather conditions, vacation plans, and economic activity indicators exist. This implies that identifying and selecting relevant predictors for regression models is a significant step toward avoiding overfitting and improving forecasting precision [13]. Moreover, it can be found that non-Gaussian and highly salience distributions are usually present in load data [14], and these characteristics are not suitable for most of the traditional approaches for forecast-based models since they violate the basic assumptions. This means that effective models need to incorporate such relation patterns, and therefore, the structure includes load data and other aspects such as climates and economic factors. The temporal and, mainly, spatial structure of load data adds further challenge because the load often exhibits different patterns depending on the geographic location and time of day or year, and this information must be incorporated into models by recognizing and providing for spatial correlations and variations in load.

Further, the clarity and interpretability of STLF models are critical as these models play a crucial role in driving key decisions in the energy sector, such as Load management [15]. Those undertaking decision-making roles in energy systems must have comprehensible model results to act promptly and efficiently. However, a need for data, which is both scarce and of questionable quality, is the last but very important matter that hinders the creation of accurate STLF models. The high-quality data needed to train and validate the different models points to the significant need to address data issues if the accuracy of the climate forecasts is to be enhanced.

These challenges have, for instance, been elaborated in detail to understand the best methodological approaches to conduct research in the STLF and other solutions towards creating innovative strategies to solve them within the field. Similarly, for coping with the non-stationary data, new models that are time-varying, such as TVARMA, and for having better learning algorithms, such as DNNs, have been developed [11], [14]. To enhance dealing with uncertainty and risk, various statistical methodologies like quantile regression and scenario-based forecasting have been employed [16], [17]. The integration of STLF with the forecasts of renewable energy intends to enhance accuracy. It is further supported by demand response and energy storage to cope with fluctuations in the renewable source [12]. Methods used for feature selection and dimensionality include feature selection and dimensionality reduction, which is crucial in handling the huge and complicated nature of STLF data. At the same time, robust models and distribution approaches address non-Gaussian distributions of data [18]. Nonparametric and kernel-based models are preferred for nonlinearity relations since they are more effective than parametric models [14]. Spatial and temporal models and clustering methods are properly employed to account for and promote spatial properties dependency. Causal inference methods are used to enhance the trust in models for energy management as transparency and interpretability of the models are essential for informed decision-making in the corresponding domain [14]. The aforementioned data availability and quality concerns are also discussed, as well as other methods, including data augmentation, transfer learning, and domain adaptation. These methods enhance the reliability and versatility of the STLF models and extend the horizons of forecasting trends to the energy industry [19], [20].

B. PAPER CONTRIBUTION

The study comprehensively reviews STLF methods, including time series analysis, regression-based frameworks, artificial neural networks (ANNs), and hybrid models that employ different forecasting approaches. Detailed mathematical and graphical analyses and a comparative evaluation of these methods are provided, highlighting their advantages and disadvantages. Furthermore, the research suggests a combined model of CNN and LSTM for deeper characteristics extraction and temporal sequence analysis, respectively, to enhance prediction precision. This work uses a possessive dataset of the national network of NTDC Pakistan to test and validate the model. The proposed model demonstrates exceptional accuracy in single-step and 24-hour forecasting, achieving significantly lower error metrics than the reference methods. For single-step forecasting, the NTDC dataset shows an RMSE of 538.71, MAE of 371.97, and MAPE of 2.72, while the AEP dataset achieves an RMSE of 126.35, MAE of 94.27, and MAPE of 0.64. For 24-hour forecasting, the NTDC dataset reports an RMSE of 951.94, MAE of 656.35, and MAPE of 4.72, and the AEP dataset shows an RMSE of 702.97, MAE of 507.97, and MAPE of 3.39.

C. STRUCTURE OF THE PAPER

The paper is divided into two main sections, as shown in Figure 1: a comprehensive review and development of a practical hybrid model for STLF. The review examines various STLF models, including statistical, intelligent, and hybrid models, each subject to detailed analysis. The (2^{nd}) section of the paper presents an innovative hybrid approach using LSTM and CNN for single-stage and multi-stage load forecasting.

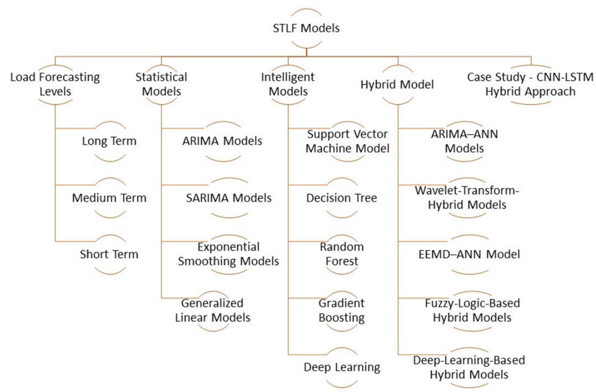


FIGURE 1. Organization of the paper.

II. LOAD FORECASTING LEVELS

Load forecasting operates at different levels, each serving specific managerial goals and employing various methods. Micro-level forecasting involves analyzing smaller segments within a region to estimate load patterns, which is crucial for managing energy plants and grids.

Macro-level forecasting looks at broader areas like cities or countries without dividing them into smaller parts. Regarding the time of the forecast for the load, they are divided into three levels, including short, medium, and long forecasting levels, as shown in Fig. 2 [21]. Short-term forecasting predicts hourly loads, important for energy plant scheduling and grid stability. Mid-term forecasting covers monthly to yearly projections for managing peak consumption and resource planning. Long-term forecasting helps with major decisions like investing in infrastructure and planning for future energy needs. Despite using different methods, all forecasting levels

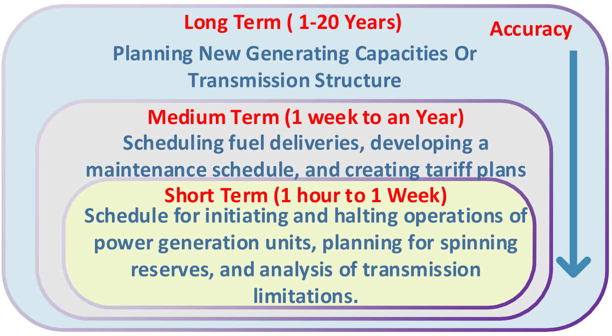


FIGURE 2. Time frames for the load forecasting levels.

consider factors like population growth and energy prices to make accurate predictions. In this study, the STLF models are emphasized.

III. STLF MODELS

Traditional methods for STLF rely on historical data related to electricity consumption, weather conditions, and calendar events. Conversely, the proliferation of emerging data channels, such as social media analytics and granular telemetry from smart measurement infrastructures, represents an excellent opportunity to improve the accuracy and robustness of STLF paradigms. Using sophisticated computational methods, including deep neural architectures and adaptive reinforcement learning algorithms, is expected to significantly optimize STLF systems' precision. Deep learning enables the automatic extraction of intricate data patterns and relationships, while reinforcement learning facilitates optimizing actions based on environmental feedback [22]. Further, converting the forecasts into point forecasts enables power system operators to make the right decisions when the actual uncertainty is experienced by including the probability estimation of the point forecasts. STLF models cover statistical, intelligent, and hybrid systems, all of which have their merits and areas of application [23].

A. MODELS BASED ON STATISTICAL TECHNIQUES

Statistical tools grounded on analysis concepts of time series load demand profiles are particularly useful in explaining the temporal dynamics contained in load demand profiles. Based on the bibliography found in the course of the research, the primary types of statistical structures used in the field of STLF are as follows: ARIMA models, their seasonal counterparts, SARIMA, generalized linear models, and exponential smoothing models [24]. ARIMA models crucially assume that current load requirements are formed from past load sequences and a stochastic term. SARIMA models enhance this structural plan by incorporating fluctuations of a seasonal nature, which are almost always present in energy activity systems where loading demand varies cyclically: diurnal or menstrual. This integration helps improve the cyclical temporal dependencies to recur at specific intervals to improve precise prediction models where the load is dominated by well-defined seasonal behaviour [25]. In turn, exponential smoothing (ES) models forecast the future load demand based on seasonal factors by providing certain weights to the previous values. Despite being accurate, quick to compute, and relatively easy to implement, statistical models have several limitations stemming from their basic structure; they may not delineate the nonlinear links and dependencies in power systems and, consequently, could provide less accurate predictions. EMS for power systems employs STLF as a foundational component for advanced energy applications [26]. Hence, a model that can predict accurate STLF is of utmost significance in optimizing the power system's energy supply and demand balance, leading to increased efficiency in utilizing energy, reduced costs, and system reliability. A detailed explanation of some of these models and a literature review are presented in the following section.

1) AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA) MODELS

ARIMA models find extensive application in STLF owing to their straightforward nature and capacity to capture temporal patterns within load data. These models consist of three main components: the autoregressive (AR) component, which quantifies the influence exerted by predecessor values; the integrated (I) component, which incorporates differentiation to achieve stationarity within the data set; and the moving average (MA) component, which includes the influence of prior stochastic deviations [27]ARIMA frameworks provide considerable adaptability through their parametric customization options, allowing for meticulous calibration of the AR and MA sequences, alongside the differentiation magnitude in the I segment. Such configurability facilitates the model's alignment with the dataset's peculiarities and the precise forecasting prerequisites [28]. Nonetheless, ARIMA models operate assuming that load data adheres to a stationary process, a condition that might not consistently hold in power system contexts. Moreover, these models may fail to capture the intricate nonlinear connections between load patterns and various influential factors like weather conditions and occupancy levels.

Figure 3 gives the algorithm for ARIMA models. The notation represents the ARIMA model as ARIMA(p,d,q); p refers to the number of the autoregressive term 'AR,' d refers to the degree of differencing applied to make a time series stationary, and q refers to the number of moving average term 'MA.' These parameters are explained by comparing the specific features of the investigated times series inherent characteristics employing advanced stochastic analysis methods, including the maximum likelihood estimation, to guarantee the reliability of the fit between the suggested model and observed data. After assessing these parameters, the ARIMA model is ready to forecast new values in the time series. Its mechanism uses past data to guess the next entry in the series with the help of the AR and the MA parts. Subsequently, the forecasted value is used to adjust the error term, using which an improved estimate of the temporal increment is derived for the subsequent temporal segment. This process repeats itself routinely and produces a final forecast covering pre-designated future periods.

The AR facet requires that today's observation in the time series depends on past observations in the same time series. An AR configuration of order 'p' is designated as AR(p), with its respective formulation explicated as follows:

$$Y(t) = \mathtt{C} + \varphi_1 Y(t-1) + \varphi_2 (t-2) \ldots\ldots\ldots\ldots\ldots + \varphi_p (t-p)\, \mathcal{E}(t) \tag{1}$$

In (1), $Y(t)$ represents the value of the time series at time t, reflecting the current state of the series. c is a constant term representing any fixed, unchanging component within the time series. $\varphi_1\varphi_2 \ldots\ldots\ldots \varphi_p$ denote the autoregressive coefficients, each capturing the influence of past values of the time series on the present value. These coefficients are integral to understanding the temporal dependencies within the series. p signifies the order of the AR model, indicating the number of past observations considered in predicting the current value.

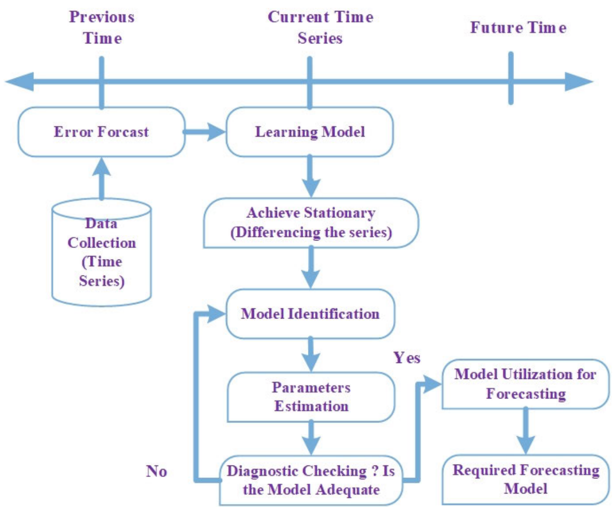


FIGURE 3. Algorithm of the ARIMA model.

A higher p-value implies a more extensive consideration of past values in the prediction process. $(\varepsilon)(t)$ represents the error term at time t, encapsulating the discrepancy between the predicted value generated by the AR model and the actual observed value at that time step. This term reflects the unpredictability or randomness inherent in the time series data. In the context of the ARIMA model, the parameter 'd' signifies the process of differencing. The time series for the first-order difference is:

$$\Delta\mathfrak{Y}(t) = \mathfrak{Y}(t) - \mathfrak{Y}(t-1) \tag{2}$$

Additional stratifications of differencing can be achieved by iteratively executing the differencing maneuver on a dataset previously modified by an initial differencing, thereby intensifying the transformation to ensure enhanced statistical constancy. For example, the mechanism of second-order differencing materializes by applying differencing anew to a series that has already been subjected to a first-order differencing process.

$$\Delta^2\mathfrak{Y}(t) = \mathfrak{Y}^t(t) - \mathfrak{Y}^t(t-1) \tag{3}$$

The MA component embodies the influence of prior error terms on the current value of the time series. An MA model, characterized by an order 'q,' is denoted as MA(q), with its equation expressed as follows:

$$\mathfrak{Y}(t) = r + \in(t) - \psi_1 \in(t-1) - \psi_2 \in(t-2) - \ldots - \psi_1 \in(t-\varrho) \tag{4}$$

In (4), $\mathfrak{Y}(t)$ denotes the value of the time series at a time t, reflecting the present conditionsof the series. The term r represents a constant factor that remains invariant across the temporal span. In the equation, $\in(t)$ signifies the error term at time t, encapsulating the discrepancy between the observed value and the value predicted by the model at that specific time. The coefficients $\psi_1\psi_2\psi_3 \ldots\ldots \psi_\varrho$ represent the moving average parameters, determining how much past error terms influence the current series value. These coefficients delineate the temporal interdependencies in the series attributed to antecedent errors. The variable ϱ specifies the order of the MA model, representing the count of prior error terms factored into the forecast of the current value. An elevated ϱ value suggests an expansive inclusion of historical error terms in the predictive framework.

By integrating the AR, I, and MA components, we can represent an ARIMA(p,d,ϱ) model as:

$$\begin{aligned}\Delta^d\mathfrak{Y}(t) &= \mathtt{C} + \varphi_1\Delta^d Y(t-1) \ldots\ldots \varphi_p\Delta^d Y(t-p) + \in(t) \\ &\quad - \psi_1\in(t-1) - \ldots - \psi_q\in(t-q)(t) \\ &= \Delta\mathfrak{Y}(t) = \mathfrak{Y}(t) - \mathfrak{Y}(t-1)\end{aligned} \tag{5}$$

2) SEASONAL AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (SARIMA) MODEL

SARIMA models, building upon the foundation laid by ARIMA models, extend their capabilities to accommodate seasonal variations inherent in various datasets. These models introduce additional parameters explicitly designed to capture and interpret the recurring seasonal patterns present within the data. Within this array of parameters are considerations on the seasonal interval, with the capability to proficiently model and predict seasonal variances across various data contexts [29]. While SARIMA models prove advantageous for STLF endeavours within power systems characterized by pronounced seasonal variations, such as load periods with daily or weekly peaks, they are not without limitations. Analogous to ARIMA constructs, SARIMA configurations are predicated on a stationary sequence and demonstrate deficiencies in comprehensively delineating the nonlinear interdependencies between load dynamics and various contributory elements [30]. Additionally, the model demands an extensive corpus of historical data to accurately ascertain seasonal parameters, which could pose difficulties for nascent power system infrastructures. The schematic representation of SARIMA models is illustrated in Figure 4.

These models are delineated by three core parameters: $\wp, \partial, q$ governing the non-seasonal facets, alongside $\mathcal{P}, \mathcal{D}, \mathcal{Q}$, which dictate the seasonal components. This schema $\wp$ is allocated to represent the autoregressive order, ∂ is employed to specify the extent of differencing, and q is utilized to define the moving average order for the non-seasonal segment. Similarly, within the SARIMA architecture, the parameters $\mathcal{P}$, $\mathcal{D}$, and $\mathcal{Q}$ correspond to the order of autoregression, the degree of differencing, and the order of the moving average, respectively, each tailored to address the seasonal nuances of the model. The process of fitting a SARIMA model encompasses a series of methodical steps. Initially, model parameters undergo maximum likelihood estimation, a sophisticated statistical method aimed at discerning the most appropriate values for $\wp, \partial, q\, \mathcal{P}, \mathcal{D}, \mathcal{Q}$ that optimize the possibility of the perceived dataset [49]. After parameter estimation, the model is fitted to the dataset utilizing advanced forecasting techniques.

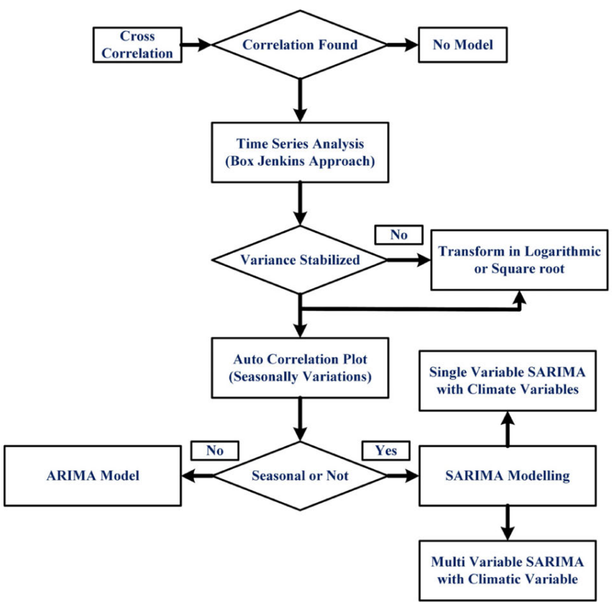


FIGURE 4. SARIMA Model approach.

This tactical methodology leverages the model to forecast future time intervals utilizing historical datasets. The following are the formulas delineating the facets of the SARIMA model, which include the Integrated (I), Seasonal Moving Average (SMA), MA, Seasonal Integrated (SI), AR, and Seasonal Autoregressive (SAR), alongside the composite equation.

- Component for the Autoregressive (AR):

$$\vartheta\,(\mathfrak{B})\,\mathcal{X}_t = \mathfrak{c} + \Psi(\mathfrak{B})\epsilon_t \tag{6}$$

In expression (6), $\vartheta\,(\mathfrak{B})$ is delineated as the autoregressive operator, with $\mathfrak{B}$ signifying the backshift operator. Here, $\mathcal{X}_t$ denotes the sequential data at time t, whereas $\mathfrak{c}$ encapsulates a fixed coefficient. Additionally, $\Psi\,(\mathfrak{B})$ is delineated as the moving average operator, while εt embodies the stochastic error constituent.

- Component for the Differencing (I):

$$(1-\mathfrak{B})^d\mathcal{X}_t = \mathbb{Y}(t) \tag{7}$$

In this formulation, the notation $(1-\mathfrak{B})^d$ characterizes the differencing operator, while $\mathbb{Y}(t)$ denotes the time series that has been transmuted via differencing.

- Component for MA

$$\mathcal{X}_t = \mathcal{U} + \sigma_1\zeta\,(t-1)\ldots\ldots + \sigma_\varrho\zeta\,(t-\varrho) \tag{8}$$

Within this schema, $\mathcal{U}$ designates the arithmetic mean of the series while σ_i encapsulates the coefficients correlated with the moving average (MA) component.

- Component for SAR

$$\Theta_s(\mathcal{B}^s)\mathcal{Y}_t = \psi_s(\mathcal{B}^s)\varepsilon_t \tag{9}$$

Within this delineation, $\Theta_s(\mathcal{B}^s)$ is identified as the seasonal autoregressive operator, whereas $\psi_s(\mathcal{B}^s)$ characterizes the seasonal moving average operator.

- Component for Seasonal differencing

$$(1-\mathcal{B}^s)^{\mathfrak{D}}\,\mathbb{Y}_t = \mathcal{Z}_t \tag{10}$$

where $\mathcal{Z}_t$ denotes the seasonally differenced time series, the seasonal differencing operator is represented by $(1-\mathcal{B}^s)^{\mathfrak{D}}$.

- Component SMA

$$\mathbb{Y}_t = \Phi_s\varepsilon_t\,(t-s) + \ldots + \mathcal{Z}_t\Phi_s\mathcal{Q}\varepsilon(t-\Phi_s) \tag{11}$$

where Φ_{si} are the coefficients for the seasonal moving average component.

Combining these elements, the equation for the SARIMA-based model can be articulated as follows:

$$\vartheta\,(\mathfrak{B})\,\vartheta_s\left(\mathfrak{B}^s\right)(1-\mathfrak{B})^{\mathfrak{D}}(1-\mathfrak{B}^s)^{\mathfrak{D}}\mathcal{X}_t = \mathfrak{c} + \Psi(\mathfrak{B})\Psi_s(\mathfrak{B}^s)\varepsilon t \tag{12}$$

• ARIMA VS SARIMA MODEL

While comparing both models, it may be noted that, in short-term load forecasts, both ARIMA and SARIMA models have their own merits and demerits [31]. ARIMA proves superior because it is far more economical with fewer parameters, particularly when the load data is non-seasonal or has minimal seasonality. It captures the relative change over a period, trends, cycles, and even random fluctuation, helping to give a very good forecast in stable and non-periodic environments. On the set of its drawbacks, the main disadvantage can be considered an inability of ARIMA to model seasonality, which can become an important problem while considering such load patterns that demonstrate clear cyclical fluctuations in the loading and demand during a day or a week.

SARIMA is a little specialized, with several extra parameters, particularly for seasonality. Due to its ability to model

TABLE 1. Comparison analysis of ARIMA and SARIMA models [31].

Features	ARIMA	SARIMA
Seasonality	It does not explicitly model it	Explicitly models it
Components	AR, I, MA	AR, I, MA, SAR, SI, SMA
Use Case	Non-seasonal data	Seasonal data
Complexity	Simpler	More complex
Forecast Accuracy	Higher for non-seasonal data	Higher for seasonal data

and factor in seasonal variations, SARIMA is highly applicable in short-term load forecasting scenarios where load data contains strong seasonality characteristics, such as peak loading during certain hours or certain days. Since these recurring seasonal variations can be modeled, SARIMA models are generally more accurate than ARIMA models in similar cases. However, due to the complexity of SARIMA, the parameters need to be tuned, and sufficient data should be obtained to avoid overfitting.

3) MODELS WITH EXPONENTIAL SMOOTHING

Exponential Smoothing (ES) models are predictive algorithms within the time series domain, utilizing exponentially weighted averages of antecedent observations to project forthcoming values. Adjustability within these models is achieved by manipulating the smoothing constant, which governs the weight allocations to historical data points. ES models encompass a multitude of sophisticated variants, such as Holt's Linear Exponential Smoothing (Holt), Simple Exponential Smoothing (SES), and Holt-Winters Seasonal Exponential Smoothing (HW) [32]. SES models apply just one index of smoothing, which helps attain load data based on observations. Holt models include another coefficient, g, that deals with the series's linear trends. Whereas HW models define the trend and contain the parameters that account for seasonality, the approach used in the model is superior when analyzing periodic fluctuations in the load data. These HW models are especially suitable for STLF in systems characterized by consistent trends and definite seasonal patterns [33]. It can be used in real-time prediction and is not as computationally demanding as the ARIMA and SARIMA models; ES models use a relatively small amount of historical data. They can also capture nonlinear associations with load data and other effects like weather conditions and population density. However, load data is assumed to remain stationary in the case of ES models, implying that the accuracy of ES models when it comes to non-stationary load in power system is affected.

The ES models' operation process flow is depicted in Fig. 5. The ES framework is characterized by two key parameters: alpha and beta. The first paper selected for the analysis is the work of Fountain and colleagues, published in the Journal of Midwifery & Women's Health in 2018. The value of the alpha parameter defines the primary importance of the last value, while the beta parameter sets the importance of the trend factor in the given model. The procedure for calibrating an ES model encompasses multiple stages. Initially, the baseline and trend estimates are derived from the early observations in the time series. Subsequently, these estimates are recurrently refined for each additional observation by employing the equations outlined below:

$$\mathcal{L}(t) = \sigma \times \mathcal{Y}(t) + (1-\sigma) \times (\mathcal{L}(t-1) + \mathfrak{I}(t-1)) \tag{13}$$

$$\mathcal{T}(t) = \beta(t) \times ((\mathcal{L}(t) - \mathcal{L}(t-1)) + (1-\beta) \times \mathfrak{I} \tag{14}$$

Here, $\mathcal{L}(t)$ captures the level of the data at a particular time and translates the current status of the phenomenon under study, whereas $\mathcal{T}(t)$ depicts the trend, describing the rate of change of a certain occurrence.

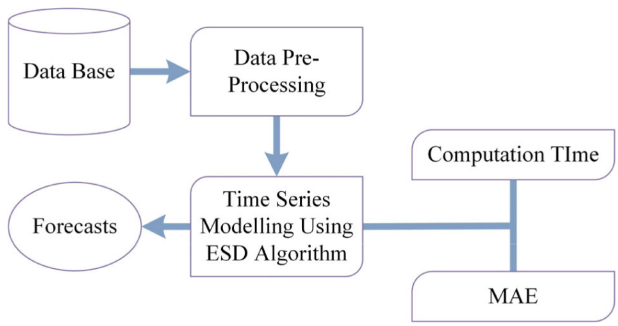


FIGURE 5. ES model principal.

Therefore, once the level and trend estimates have stabilized, it can forecast future periods with even more accurate and precise ES models. The projection for the upcoming period $t+1$ is derived by summing the current level and trend components, as per the equation: The projection for the upcoming period $t+1$ is derived by summing the current level and trend components, as per the equation:

$$\mathfrak{F}(t+1) = \mathcal{L}(t) + \mathcal{T}(t) \tag{15}$$

4) MODELS FOR GENERALIZED LINEAR STRUCTURE

GLM is a large and systematic class of advanced statistical techniques exploring the weak nonlinear relation between load profile and influential factors. Various extensions of GLMs exist, including the Poisson regression model, the Negative Binomial regression model, the Gamma model, and others [34]. GLMs extend the linear regression model by providing the ability to manage the non-normal dependency of the response. The benefits of GLMs are that they can handle highly flexible load patterns due to the change in load behaviour with meteorological parameters and/or occupancy patterns based on the link function. This function provides an estimated value of the dependent variable as $E(y) = f(X_1, X_2, \ldots\ldots\ldots\ldots X_k$ from the predictor variables $X_1, X_2, \ldots\ldots\ldots\ldots X_k$ [35]. Likewise, in STLF, the primary focus of most GM applications is the Poisson regression model, the most common type of GLM, where the load count is the dependent variable to the relevant control variables. The Poisson regression model presupposes that the described response variable is Poisson distributed and maps the mean of this variable through a logarithmic link function to the set of predictors. Indeed, the Poisson regression is particularly adapted to power systems that involve counting data, which means the number of working or existing appliances or devices. In its own right, one must acknowledge the function that PCA serves as a powerful tool for capturing complex nonlinear relationships between loads' variability and a range of potential drivers [36]. The Poisson regression model, employing a log-linear framework, formulates the relationship between the predictor variables λ and the anticipated count X. This relationship can be expressed as The Poisson regression model, employing a log-linear framework, formulates the relationship between the predictor variables λ and the anticipated count X. This relationship can be described as:

$$\mathcal{L}og\xi = \alpha_0 + \alpha_1\mathcal{X}_1 \tag{16}$$

In this context, α_0 refers to the intercept, whereas α_1 refers to the coefficient estimate for $\mathcal{X}_1$. Another advanced GLM that is employed in STLF is the Negative Binomial Regression model, which follows the formula: Derived from Poisson regression models, it is most appropriate in working with over-dispersed count data by including a prevalence parameter by thus improving preciseness and reliability in conditions of high data variability. This is because the variance of the response variable is generally more than its mean and is quite common for load datasets considering power systems due to high variability [37]. The logarithmic link function used in the negative binomial regression model is an advanced statistical method useful in accurately measuring the complex relationship of the response variable with the predictor variables in case the expected value of the response cannot be assumed to be a linear function of the predictor variables. This relationship is explained proficiently in linear algebra by describing the association between the predictors and the outcome in terms of a linear combination, thereby providing a rich appreciation of how each predictor contributes to the prediction. The difference in this model lies in including one more dispersion parameter to reduce the density of the response variables. This parameter contributes additional robustness to the method because it is inherently possible to have variability within the datasets used in real-world applications. Indeed, incorporating the third variable, which accounts for dispersion, enhances the model's capabilities by ensuring that the model can capture fine details of the data sets. The articulation of the dispersion model is as follows: The articulation of the dispersion model is as follows:

$$\mathcal{L}og\,(\varphi) = \alpha_0 + \alpha_1\mathcal{X}_1 \tag{17}$$

In this context, φ belongs to the mathematical expectation of the response variable, mean, which is a primary measure of the central tendency of any set of data. On the other hand, the symbol α_0 represent the model intercept representing the response variable's baseline level, whereby all the predictor variables have been set to zero. Moreover, α_1 refers to the coefficient that corresponds to the predictor variable, which is $\mathcal{X}_1$ in this case, hence showing the degree of its impact on the mean of the response variable. Novelty is deemed the dispersion parameter κ, which brings variability to the model. It can either be made fixed so it only offers a fixed measure of dispersion or can be estimated depending on the predictor variables, which provides the possibility of changing in response to changes in the data used to create the model. When κ is modeled as a function of the predictors, it follows a logarithmic equation: When κ is modeled as a function of the predictors, it follows a logarithmic equation:

$$\mathcal{L}og\,(\kappa) = \rho_0 + \rho_1 z_1 \tag{18}$$

Here, ρ_0 term provides the intercept for the dispersion component whereby the value of ρ signifies the general basal level of dispersion. At this stage, ρ_1 represents the estimator of z_1 which shows its effect on the variability of the response variable. Gamma regression is identified in detail as one of the significant tools in the Generalized Linear Models category specifically utilized in Short-Term Load Forecasting during power systems. Since the response variable is an instance of the gamma distribution, this regression method brilliantly transforms continuous load data and the dependent variable in terms of the predictor variable.

$$\mathcal{G}\,(\mathcal{U}) = \eta \tag{19}$$

In (19), it also defines a complex interaction between the mean of the response variable $\mathcal{U}$ and the linear predictor η via an appropriate link function $\mathcal{G}$. The integration of the link function with the linear predictor results in a comprehensive equation that encapsulates the complex interplay between the response variable and the predictor variables: The integration of the link function with the linear predictor results in a comprehensive equation that encapsulates the complex interplay between the response variable and the predictor variables:

$$\mathcal{G}\,(\mathcal{U}) = \rho_0 + \rho_1\mathcal{X}_1 + \rho_2\mathcal{X}_2 + \ldots + \rho_P\mathcal{X}_P \tag{20}$$

GLMs boast computational efficiency and the flexibility to accommodate diverse predictor variables, encompassing both categorical and interaction terms. Furthermore, GLMs furnish interpretable coefficients, facilitating the identification of pivotal predictor variables and quantifying their impact on the load [38]. However, a significant drawback of GLMs lies in their assumption regarding the distribution of the response variable, which may not universally apply to power system contexts [39]. Additionally, the accurate estimation of model parameters via GLMs often necessitates a substantial volume of historical data, presenting challenges for newly established power system installations with limited data availability [40].

B. MODELS WITH INTELLIGENT TECHNIQUES

Sophisticated methodologies in STLF encompass a range of prognosticative techniques that employ cutting-edge computational strategies, including artificial intelligence, machine learning, and various optimization algorithms, to anticipate near-future electrical demands. These advanced models are intricately designed to discern and interpret complex patterns, nonlinear interdependencies, and multifaceted relationships within the load data, thereby enhancing the precision and dependability of forecasts [41]. Figure 1 delineates several prominent intelligent models employed in STLF, which have been discussed in detail.

1) METHODS WITH SUPPORT VECTOR MACHINE APPROACH

A critical machine learning archetype for STLF is the Support Vector Machine (SVM), an advanced supervised learning framework proficient in addressing linear and nonlinear associations between the electrical load and its influencing determinants. SVMs are remarkably versatile in handling an extensive spectrum of predictor variables, which include categorical variables and interaction terms [42]. Furthermore, these models are adept at mitigating issues related to noisy data by applying a kernel function that transposes the data into a higher-dimensional realm for enhanced analysis. SVMs have been widely employed in predicting load dependent on the/power system variables like temperature, humidity, and time. In the context of the domain STLF, it is unobtrusively incorporated to predict the future load or demand on electricity for up to one week. However, other intervals, such as a few hours, can also be used. In general, an SVM aims to find the best hyperplane that offers the maximum likely separation of two classes of regression data sets. In the two-dimensional case, it appears as a hyperplane, which in this context represents a line that separates the data distinctly into two separate categories. In ad-dimensional geometries, a hyperplane is a sub-space born of the containment space and is reduced by one dimension. A margin represents the distance between the hyperplane and the neighbours of distant classes of points referred to as support vectors. These vectors are essential in the structure and position of the hyperplane as they provide its support. The fundamental objective of an SVM revolves around optimizing this margin to ensure the precise categorization of data elements. Within a dataset containing labelled data pairs (x_i, y_i) where x_i represents a vector in n-dimensional real space $\mathcal{R}_n$ and y_i designates the binary class labels $(-1, 1)$, the delineation of the hyperplane is established mathematically [43].

$$w.x + b = 0 \tag{21}$$

In this context, w signifies the vector of coefficients, x encapsulates the vector of input features, and b is identified as the intercept. The scalar product $w.x$ calculates the extent of the projection of x in the direction oriented by w. The task is to determine the most advantageous values for w and b that effectively maximize the separation, or margin, between the two categorically distinct classes. The discriminative criterion for categorization is

$$f(x) = sign(w.x + b) \tag{22}$$

where sign represents the signum function, which assigns the class label according to the sign of its argument. The margin for each data point is determined in the following manner:

$$y_i(w.x_i + b) \tag{23}$$

where, In the world of binary-defined classes y_i represents the ith data point's classification, often as -1 or $+1$. In turn, w denotes the weight vector that is orthogonal to the defining hyperplane, whereas x_i refers to features for the ith data sample. Moreover, b_i the bias term is the important function that shifts the delineating boundary from the origin. The notation $w.x_i$ represents the dot product between the network weight vector w and the feature vector x_i, which is a representation of the projection of x_i about the vector w. In the context of the SVM paradigm, the fundamental aim, therefore, is to understand how to obtain the biggest margin and, at the same time, ensure all pattern vectors are classified correctly. This optimization is made by controlling the size of the weight vector, which is represented as $|w|$, where it is known that the distance between the hyperplane and the data point is inversely proportional to $|w|$. Building on this concept, the SVM is meant to look for the perfect hyperplane that not only partitions the data points according to their classes but also does it with the largest margin, measured as the distance of this hyperplane from the nearest data point of any of the class. Thus, by minimizing $|w|$, the classifying hyperplane of the SVM increases the said margin, and this, in turn, improves the overall performance of the model in terms of its ability to generalize. This process makes it possible for the SVM to fulfil the optimal condition of maximizing the margin while minimizing the classification errors and, hence, arriving at an efficient and accurate classification system.

$$M = \frac{1}{|w|} \tag{24}$$

The optimization conundrum for an SVM can be articulated as a constrained optimization quandary.

$$Min = \frac{1}{2} \times |w|^2 \tag{25}$$

Subject to the condition: $y_i\,(w.x_i + b) \geq 1$ for each i from 1 to i. This situation leads to a convex quadratic programming problem with linear constraints imposed on it. This task is a challenge that is solved with the help of finding extrema by using Lagrange multipliers, which thus leads to what is called the dual problem. The dual formulation presents a more practically applicable approach to deal with the problem of optimization integrated in SVM, particularly in cases where kernel functions are introduced in nonlinear situations [44]. The steps involved in STLF are as follows, and a flow diagram is given in Figure 6.

- The first step is to collect past load demand data on electricity, other systemic parameters, external environmental conditions, and days of the week and time. It is then followed by a cleanse and preprocess step to eliminate or minimize any inconsistencies, noise, or missing values in the data. This

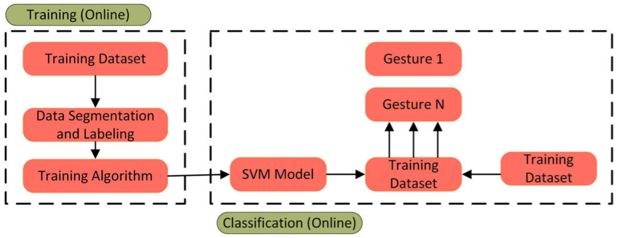


FIGURE 6. **Machine system based on support vector mechanism.**

is usually done through the normalization of standardization methods to enhance the performance and quality of the SVM.

• In the next step, the attributes relevant to the forecasting enterprise are identified for further assessment solely. This stage is paramount, particularly if unnecessary functions negatively affect the model's performance. Also, there are RFE, correlation reviews, or PCA to identify the features of the problem that would be optimal to focus on.

• To elaborate further, the selected model of SVM is carefully trained on the refined data and the chosen variables only. SVMs aim to obtain the best hyperplane that distinguishes the studied data sets into distinct categories or classes. The model is given for predicting continuous values to be the electrical load, given that STLF is naturally a regression problem. It should be understood that these kernel functions are designed to map the input data to a space of higher dimensionality. It helps when finding an appropriate hyperplane in this expanded space of slowly growing dimensionality to separate the points.

• The SVM model is initially trained using preprocessed data, as discussed in Section III, and the most appropriate attributes are selected. SVMs, as a technology, aim to classify the data as aptly as possible into well-defined classes or categories. For example, while STLF is a regression problem, the model is set to forecast the continuous values for electrical load. To achieve this, the SVM employs what is known as kernels (which could be linear, polynomial, or the radial basis kernel function,) which transforms the input data into the higher dimensional space, making it easier to identify a plane of best fit that separates the classes.

• To ensure everyone knows that the SVM model developed to solve different problems performs well, it is tested using measures like cross-validation. Splitting the data into the training and the validation set is performed in this process, and the former Mode is trained while the latter mode is tested. This helps to determine the model's performance apart from the test dataset and check for the model's ability to perform well across the other dataset.

• Moreover, the present enhancements include the cost factor, the type of kernel function to be utilized, and the parameters associated with the kernels that are incorporated in the best estimation for solving the specific STLF challenge.

• Once a particular model, such as the SVM model, has been trained and calibrated, its performance in terms of prediction can then be evaluated on another data set that was not used during the training. An examiner can assess the model based on factors like standard error, R square, coefficient of determination, and accuracy using MAE, MSE, or MAPE.

• It is worth mentioning that, with new input data, the proposed SVM model that is trained, validated, and tested can provide a five-minute STLF. The developed model calculates the required attributes for the forecast period and gives the expected demand for electricity load.

2) DECISION TREE

Another suitable approach in business for the STLF is the Decision Tree (DT), a statistical model specializing in segmenting predictor variables into categorical or continuous categories, as shown in Fig. 7. DTs function by dividing the predictor variables into subsets known as the leaf node. The trees are connected based on load patterns within the load prediction or mere hierarchy [45]. Decision trees have also demonstrated favorable performance in power system-related studies, especially in forecasting load variations from weather characteristics, occupancy ratios, and time factors. Therefore, divisions, lame, and splitting standards are at the heart of the decision tree [46], [47].

Regarding classification tasks, data binning through metrics such as the Gini index or information gain is used, making it easy to classify the different classes within the dataset.

• Gini Impurity: This metric assesses the heterogeneity of a node (mix of classes), where lower figures indicate a higher degree of uniformity. The Gini impurity for a node expressed through class probabilities $\mathcal{p}_i$, is formulated as:

$$\mathrm{G.I} = 1 - \sum (\pi)^2 \tag{26}$$

• Information Gain: Predicated on the concept of entropy, which evaluates the degree of disorder or indeterminacy within a group, Information Gain is computed for a node with class probabilities π as

$$Ent = -\sum (\mathcal{p}_i \times \log 2)(\pi) \tag{27}$$

Information gain (IG) is the entropy difference before and after the data elements or nodes split.

$$\mathrm{IG} = \mathrm{Entropy(parent)} - \text{ weighted sum(Entropy(children))} \tag{28}$$

To increase the information sought to facilitate the formation of equally proportioned descendant nodes, the most frequent technique used in regression efforts IX is using Mean Squared Error (MSE) to split. MSE is expressed as the mean of the squared difference between actual and estimated targets; it serves as a key criterion underlying each partition's goal to minimize error. To train the decision tree model for predictive analysis, it is crucial to correctly obtain previous electricity load data and other independent variables like weather parameters, day and month of the week, and time, and address any anomalies or missing values. Pre-processing

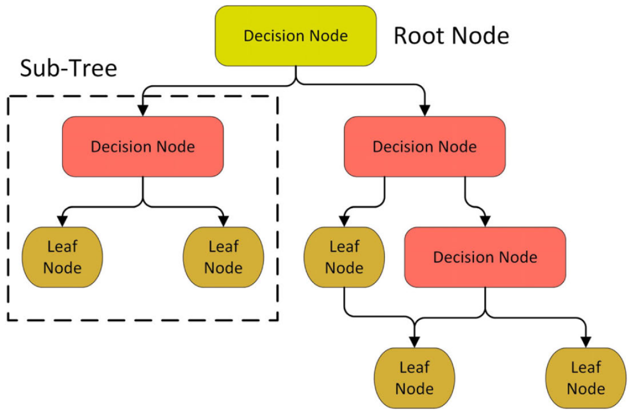


FIGURE 7. Block decision tree diagram.

techniques such as selection by using RFE, correlation analysis, or Information gain help remove attribute features that may distort the classification result. The splitting is done stepwise in the process of modeling, where the data is divided into segments to minimize impurity, as measured by MSE, until the model reaches the predetermined condition for terminating the process of building a tree, such as the maximum depth of the tree being achieved. Due to the nature of the overfitting of decision trees, indicators for validation and pruning, like cross-validation or cost complexity, are used to enhance the stability and performance of the tree. After extensive training and subsequent pruning, the final model goes through specific performance evaluation metrics such as MAE, MSE, or MAPE. Once the decision tree model is validated, the process of short-term load forecasting begins by transforming the input data into valid decision-making criteria and passing it through the tree structure, moving from the root of the tree to a node or leaf point at the end of a decision path, which is the forecasted electricity load.

3) RANDOM FOREST AND GRADIENT BOOSTING (GB)

Random Forest (RF) is a refined version of DT developed to reduce sensitivity to data noise and enhance accuracy in load forecasting. RFs use the bootstrap aggregating method, that is, many decision trees are built using bootstrap samples of the training dataset for arriving at a final decision from all these individual trees, and outputs from all the trees are averaged to arrive at the final prediction. This technique is used in certain power system paradigms to forecast load changes due to specific factors like temperature, humidity, and sunlight [48]. The simplified working model of an RF algorithm is depicted below in Figure 8: In each tree node, for a given dataset, 'm' number of features are randomly selected from the total features. Many trees are then regrew, leading to different predictive classifications coming from each of them. The last prediction is the overall consensus, either a simple average or the decision from most of the above-mentioned classification types [48]. GB is a sophisticated ensemble machine learning paradigm known for its capability to refine the precision of load forecasts. The method systematically incorporates successive decision trees, each tasked with rectifying the errors identified in its predecessors, thereby constructing a comprehensive model capable of capturing the intricate nonlinear dependencies between the load and its influencing variables [49].

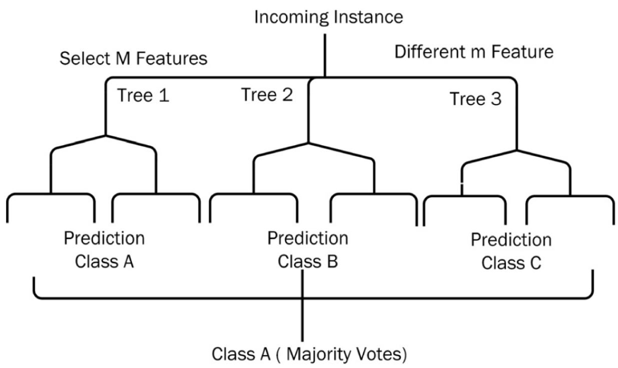


FIGURE 8. Random forest algorithm.

Figure 9 illustrates the gradient boosting algorithm process, which leverages training data to pinpoint weak learners, enhancing the accuracy of subsequent predictions. The sequential operation of the random forest and gradient boosting algorithms is delineated as follows [50].

- The initial phase aggregates a load of power data alongside variables such as meteorological data, the day of the week, and the diurnal cycle. This collected data undergoes cleansing and preprocessing to eliminate inconsistencies, anomalies, or absent values. Notably, feature scaling is typically superfluous for random forests since the decision trees, which serve as their foundational learners, exhibit a reduced sensitivity to the magnitude of input features.
- The procedural execution of random forest and gradient boosting algorithms proceeds as follows: The most pertinent attributes for the forecasting endeavor are meticulously selected to preclude the detrimental effects of extraneous or superfluous features on the model. While random forests intrinsically possess the capacity to manage many features and autonomously evaluate their significance, applying domain expertise or methodologies such as RFE and correlation analysis can substantially enhance the model's performance.
- The model is trained after feature selection for the best performing random forest model. This algorithm builds many decision trees from a bootstrap sample in the presented data set. Specifically, an independent random sample of features is selected at the split stage of tree construction. It helps increase the tree number intended to improve a model's accuracy and also helps avoid overfitting problems in the gradient-boosting model.
- This confirms that the random forest model is reliable in its performance with contemporary datasets due to the application of the cross-validation technique. The dataset is split into the train and validation sets, wherein the entire

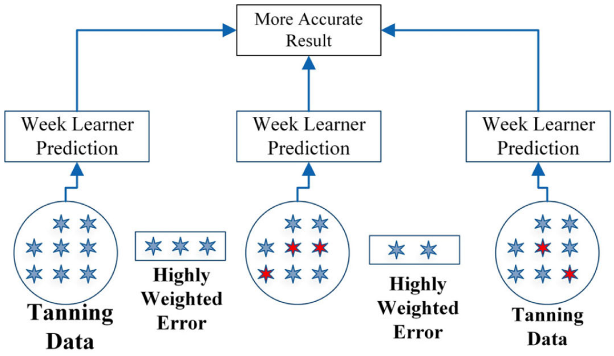


FIGURE 9. Gradient boosting algorithmic diagram.

computation process is performed on the first subset, and the results are tested on the second one. This methodology allows evaluation of the model's performance and generalization ability on a different dataset.

- The random forest algorithm contains various tunable hyper parameters. One can identify the number of trees within the forest, the specified maximum depth of each tree, and the minimum sample size of a node. These hyper parameters can be tuned and fine-tuned through such techniques as grid search, random search, [and] k-fold cross-validation to arrive at the best setup relative to a given STLF task.
- Once the random forest model is thoroughly developed and refined, it can accomplish STLF chap accordingly to the newly supplied input data. For the particular given forecasting time, the model takes the relevant characteristics and forecasts based on each tree in the decision tree. The last forecast is then calculated as the average of all these results to produce a more accurate and less volatile forecast.

4) MULTILAYER PERCEPTRON (MLP) MODEL

An MLP is an extended artificial neural network comprising several linked stages, also known as layers of neurons or perceptron layers. Figure 10 shows the architecture of an MLP; the following are the main components of this nerve [51].

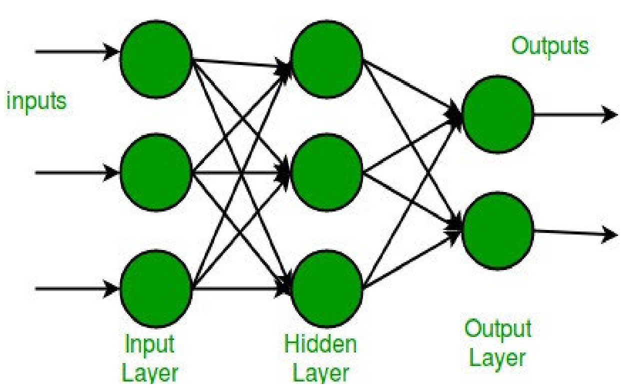


FIGURE 10. Multilayer perceptron model.

- Primary Interface: This MLP architecture's first module involves ingesting the first data received at the platform, acceptability in various forms, including numeracy, imagery, or text. Each node here stands for a specific attribute of the input data.
- Intermediary Layers: Sitting between the input and output parts, these layers incorporate neurons required for translating the complexities into the acceptable format of the system. The architecture of these layers determines the depth of the model and the number of neurons within them, which dictates the ability of the model to understand more complex patterns.
- Terminal Section: This final component of the MLP framework provides the last outcomes and or predictions of the analysis. The use of nodes depends on the type of task to be solved and whether to classify images into ten categories, as it is assumed to have ten nodes.
- Neuronal Units: Every node in this computational architecture takes in input from the earlier layer of nodes, applies a transform on this input, and then passes on the transformed input to the next layer of nodes. Fortunately, containing a non-linear activation function enables these networks to function and model complicated data configurations.
- Synaptic Weights and Biases: connectivity strength is measured in weights, which control the inputs from one neuron to another. These weights are fine-tuned iteratively in the training phase to minimize the difference between predicted outputs and actual input data.
- Model Calibration: These connected neural structures are calibrated using a complicated procedure known as backpropagation. This method operates by systematically adjusting weights and biases to reduce the gap between computed results and reality. The basic idea with Calibration is that the algorithm works continuously on a data set until it achieves better and more accurate network prediction.
- Performance Metric: The network evaluation is based on a loss metric, gauging the model's efficiency. The general structure of this kind of graphic is such that lower values mean better accuracy and effectiveness. The principle during calibration is to reduce this performance measure as much as possible since it jeopardizes the network's capabilities.

Mathematically, the output of a neuron can be represented as [52]:

$$a_{\mathcal{J}} = f \sum (\mathcal{W}_{ij} \times x_i + b_j) \tag{29}$$

In the complex design of an MLP, $a_{\mathcal{J}}$ is the quantity we name the activation, which is the output of a neuron $\mathcal{J}$, is decided by an activation function f. This function is governed by the weighted sum of inputs, which is also a linear summation of inputs, where $\mathcal{W}_{ij}$ is a weight connecting the input i to the neuron j and is the value of the input x_i. Furthermore, the bias term b_j is introduced to each neuron, enabling the model to be more flexible since the weights can be adjusted with the additional input. To denote the above relation in the MLP for each consequent stratum, the matrix construct will make it possible to have computed right across the network layer concurrently. This representation significantly improves activation calculations in many network layers, thereby increasing the effectiveness of deeper learning architectures.

$$\mathcal{A} = \mathit{f} \sum (\mathcal{W}\mathcal{X} + \mathfrak{B}) \tag{30}$$

In this detailed description of an MLP, the matrix $\mathcal{A}$ represents the activation matrix in which a column contains the activation information of a neuron to the input examples. The matrix $\mathcal{W}$ represents the weight matrix and holds coefficients that relate the inputs with the specific neurons to alter the effect of every input feature in the neuron's output. $\mathcal{X}$ can be represented by the symbol, where each column of the matrix comprises a specific input feature vector and contains the input data that is directly supplied to the network. Finally, $\mathfrak{B}$ is the bias matrix integral to the network for adding a bias term to each neuron's weighted input, which assists in shifting the activation function curve, thus offering another degree of freedom for the learning algorithm. After the computation of activations throughout all strata, the terminal stratum delivers the ultimate forecast. In classification endeavours, a softmax function is habitually deployed in the output stratum to transfigure the activations into probabilities:

$$Soft - Max(a_i) = \frac{e^{a_i}}{\sum e^{a_j}} \tag{31}$$

In this framework, sa_i denotes the activation for output neuron, ia_j signifies the activation for output neuron j, and $Soft - Max(a_i)$ is the probability associated with class.

5) ENSEMBLE MODELS

Understanding the integration of machine learning architectures, the last category, ensemble models, integrates the predictions of multiple models to create a single forecast. The basic concept of ensemble modeling is to combine various models, where each model has its unique characteristics and strengths, and use their forecasted result together, which, when developed, forms a clearer and more robust prediction. These models have general applicability in most machine learning processes, including classifications, regression, and clustering [53]. Another benefit associated with the ensemble models is the ability to minimize variance and reduce the extent of overfitting. Variance, in ensemble models, refers to the variation of the standard deviation of the outcomes of the individual models within the ensemble. Overfitting is a model learning problem that happens when excessive detail about the training data is developed, and therefore, it is ineffective in analyzing new data. This characteristic of ensemble models provides a basis for ensuring their stability and increases their applicability on various datasets. Ensemble models do a good job of reducing variances and overfitting because they combine the real estimations of the models, reducing the overall variations that lead to a stronger prediction [54]. However, there is one disadvantage regarding ensemble models: They are highly complex. Each of these situated models is typically less straightforward in their application and analysis than isolated models and often requires higher computing power.

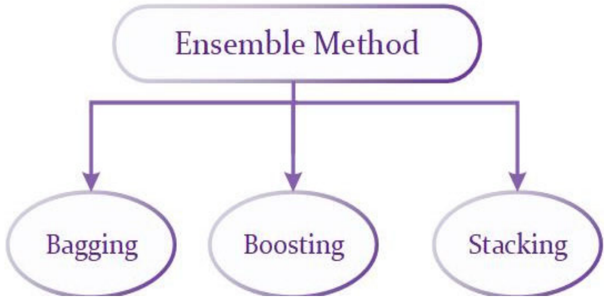


FIGURE 11. Types of the ensemble method.

However, the effectiveness of ensemble models cannot be generalized and depends on the specific combination of integrated models; identifying which configuration is most effective may be difficult. The limitations present in ensemble models include the fact that the accuracy of the models within the ensemble directly impacts the ensemble models' overall accuracy and reliability of the forecast [55]. Substandard models reduce the efficacy of ensemble models, whereby a model may cross-spread problems such as overfitting or under fitting the data. Selecting the models that would be part of the ensemble should be done carefully; ensuring they are all high-quality is important. The ensemble model construction tactics include bagging, boosting, and stacking, as shown in Figure 11. Both methods are suitable for handling data and have specific strengths and weaknesses, and the choice of a method depends on the features of the given application and the dataset. Bagging, also known as bootstrap aggregating, involves using many based models developed on different samples of the base training set to create a final forecast or solution, the weighted average of the individual predictions/solutions. The fundamental idea of bagging lies in constructing several replicas of the initial set, wherein each replica will consist of a distinct yet somehow diverse part of the set [56].

Further, models are trained on each of such executions, and the outcomes are combined to generate the final forecast. In this approach, bagging involves compiling the results of all the models that have been developed to produce a more accurate and reliable prediction. Boosting is the method through which the training of models is set in a sequential form such that each is responsible for correcting the mistakes made by its predecessor. This technique increases prediction accuracy – generally due to increased susceptibility to overfitting compared to the bagging method. According to the principles of boosting, the process begins with a simple model and continues with increasing levels of model complexity added to solve the mistakes of the prior level [57]. These models are then added with a weighted average, where the weights have direct proportionality to accuracy, and errors are gradually focused on previous models, enhancing attempts towards higher accuracy and reliability of the forecast.

The process of hierarchical aggregation, frequently known as stratified synthesis, implies that numerous models are trained successively in the training dataset, and the calculated outputs are used as inputs for the other, more complex models. The higher-level model further learns to accumulate the outcome of the lower-level models to arrive at an outcome. Despite these, the stacking model can enhance the accuracy and reliability of the prediction compared to bagging or boosting but may be harder to apply and analyze than the two [58]. The higher-level model then learns to aggregate the two lower-level predictions into one prediction. Stacking, in general, is a way to obtain a more accurate and stable forecast since, by using several models together, the result is an ensemble that is usually more precise [59]. DL is an advanced ML model classification that performs well when discovering the intricate nonlinear relationship of load by other preceding factors [60]. The DL frameworks have stacked layers of interacting nodes that help convey information using weighted connections [61]. As a significant point, it is possible to emphasize that CNNs are one of the most used DL models for STLF. These networks effectively capture load data based on space and time characteristics. CNNs are very useful in power systems, enabling the possibility of predicting changes in load due to special factors such as weather, occupancy, and time of day. The second key model for TLs specialized in STLF is the RNN, as this type of network can quickly identify load data's temporal characteristics. Sequential Neural Networks (SNNs) possess feedback connections and the ability to move from one time point to the next, thus equipping the model to quantify changes in the load over time. Altogether, compared to all the correlated works on SNNs, the especially suitable for this task is the LSTM network, as it is characterized by a rather high ability to perceive short-time and long-time temporal interdependences in the load data set [62].

C. HYBRID MODELS

As suggested by the title, they encompass both statistical and machine learning methodologies that are integrated to magnify the strengths of still. Among all the hybrids analyzed in STLF studies, ARIMA–SVR and ES–ANN are the most utilized models. The type of composite scheme employed for this particular kind of investigation in this paper is the ARIMA–SVR model, a conjunction of ARIMA and SVR architecture for capturing temporal patterns in load demand and non-linear associations with load demand [63], [64]. On the other hand, due to the integration of ES with ANN, the ES – ANN design can note complexities like changes in season and other non-linear demands in load. Accurate and hardly interpretable models are achieved; hence, the STLF models are improved for use in power systems with the help of hybrid models. However, they require fewer statistical models and are more required than machine learning algorithms, and they can have more problems implementing than purely machine learning models [65]. These biomechanical-fractured constructs, which support different modeling methodologies for assimilation efficiently, have been said to improve load forecast rates. Here are some notable hybrid models: Some hybrids have a strong combination of the two types.

1) HYBRID MODEL WITH ARIMA–ANN APPROACH

Another conventional time series forecasting approach is the ARIMA model, which serves as the basis for capturing linearity in the data. On the other hand, ANNs can categorize complex non-linear relationships. This compound model fully integrates the linear skill of ARIMA for load data and the nonlinear diagnosis capacity of ANNs by realizing linear forecasting of ARIMA & nonlinear diagnosis of ANNs. Indeed, such a synthesis significantly improves the accuracy of STLF [66].

2) HYBRID MODELS BASED ON WAVELET-TRANSFORM

One of the sophisticated analytical tools that allow for the decomposition of a time series into several discrete frequency bands is the wavelet coefficient. The resolution means that the high-frequency section contains noise and other irregularities, whereas the low-frequency section captures steady variations. Since EMD can be combined with different predictive models like the ANN, SVM, or LSTM networks, this technique can be effectively incorporated into developing the hybrid forecasting model. What is more important is that, through splitting the frequency components, this method increases the accuracy of the forecasts by an average of sixty-five percent [67].

3) HYBRID MODEL EEMD ANN APPROACH

The EEMD is a refined signal analysis that helps to recreate a non-stationary time series through a set of IMFs. This study further strengthens the proposed hybrid model by combining ANEMD with ANN to handle non-stationary and non-linear load data. In this process, EEMD filters the load data to extract the IMFs. Finally, the ANN is trained with the extracted IMFs identified in the previous section to build predictive forecasts. These individual forecasts are then summed up to arrive at the STLF output [68].

4) HYBRID MODELS FUZZY LOGIC APPROACH

Fuzzy logic is a mathematical approach designed to handle such uncertainties and impreciseness in data. It can be easily extended with other predictive methods like ANN, SVM, or regression functions. It can create a highly effective combined model to solve the high uncertainty observed in the load data. In this regard, fuzzy logic can filter the input data, represent uncertainties in modeling the forecast unit, or combine the outputs of distinct models to sharpen the forecast accuracy [69].

5) HYBRID MODELS WITH DEEP-LEARNING APPROACH

Deep learning paradigms, including CNNs and long LSTMs, have demonstrated high fitness in STLF due to their ability to uncover hierarchical and temporal informatics. These enhanced learning frameworks can be incorporated with other predictive systems like statistical models, wavelet transform, or fuzzy rules to form a blended model. Such integrative models make the best out of the integrated methodologies of STLF and significantly increase the accuracy level of the models [63].

D. PERFORMANCE COMPARISON OF STLF MODELS

It is, therefore, safe to mention that several antecedent factors determine the effects of STLF models and the volume and caliber of the variables involved. It is well established that intelligent models often have superior predictive power compared to models based solely on pure statistical considerations, and models based on a blend of statistical and intelligent framework components have been found to outperform both [70]. However, these models' somewhat relative performance depends on the specific application and the data used. For instance, statistical models might produce good outcome predictions of stationary and linear values if data is limited in quantity.

On the other hand, intelligent models are known to perform well under nonlinear and non-stationary datasets or environments, particularly if only a large number of datasets are present. When these variables present linear and nonlinear data patterns of relationship, and problems like noise or missing data elements are there, hybrid models are more beneficial. The following summarizes these observations, as given in Table 2.

TABLE 2. Different Model of STLF and their Pros and Cons.

Refs.	Model	Contributions/Advantages	Drawbacks
[71]	Generalized Linear Model	Enlarges the directions of linear regression to allow comparably skewed response distribution	The method has the least assumption and presupposes linearity between covariates and the dependent variable.
[72]	Support Vector Machine	Very efficient for non-linear classification and regression	Closely related with the identification of the model type, one has to pay a lot of attention to the choice of the parameters
[73]	Decision Tree	The method includes a non-parametric, general-purpose algorithm for classification and regression.	Squares may lead to overfitting, hence the need to schedule pruning and parameter tuning for the model.
[74]	Random Forest	An approach that employs a collection of decision trees for classification tasks.	Prone to overfitting and requires careful selection of hyperparameters
[75]	Gradient Boosting	Comprises of a group of relatively weak learning models that continuously be improved to achieve better accuracy in prediction.	Susceptible to overfitting, necessitating meticulous hyperparameter selection.
[76]	Deep Learning	Crucial in advanced applications like image recognition, natural language processing, and speech recognition.	Demands significant computational resources and large datasets for effective training.
[77]	Ensemble Methods	Employed across diverse domains, including finance, marketing, and healthcare, for robust predictive modeling	Requires judicious selection of constituent models and fine-tuning of hyperparameters
[78]	Multilayer Perceptron	A type of artificial neural network applied in fields like finance, healthcare, and image processing.	Susceptible to overfitting; mandates precise tuning of NN parameters.

IV. CASE STUDY

A. PROPOSED MODEL

The proposed model architecture effectively integrates convolutional neural networks (CNNs) with long short-term memory (LSTM) networks to process sequential data characterized by 24-time steps and 17 features per time step. This hybrid architecture leverages the strengths of both CNNs and LSTMs, enabling the model to capture spatial and temporal dependencies inherent in the data.

The model begins with an input layer that accommodates sequences of the specified shape. The first processing block consists of a 1D convolutional layer with 48 filters and a kernel size 3. This convolutional layer is intended to extract local features from the input sequences, capitalizing on the capability of convolutional operations to identify patterns such as trends and periodicities. The output of this convolutional layer is then passed through a ReLU activation function, which introduces non-linearity into the model, enhancing its capacity to capture complex relationships within the data.

Following the activation, the output is flattened, transforming the multi-dimensional tensor into a 1D vector. This flattened representation is subsequently processed by a dense layer comprising 24 nodes. This dense layer aims to further abstract and learn higher-level representations of the features extracted by the convolutional layer. The second convolutional block takes the output from the first activation layer as input and applies another 1D convolutional layer with 32 filters and a kernel size 3. This layer continues the process of feature extraction but at a finer granularity. The activation, flattening, and dense layer operations are repeated as in the first block, ensuring consistent transformation and abstraction of features. The third convolutional block follows a similar structure, utilizing a 1D convolutional layer with 16 filters and a kernel size 3. The progressively decreasing number of filters in subsequent convolutional layers is designed to reduce the dimensionality of the feature space while retaining essential patterns. After the ReLU activation, the output is flattened and passed through another dense layer with 24 nodes.

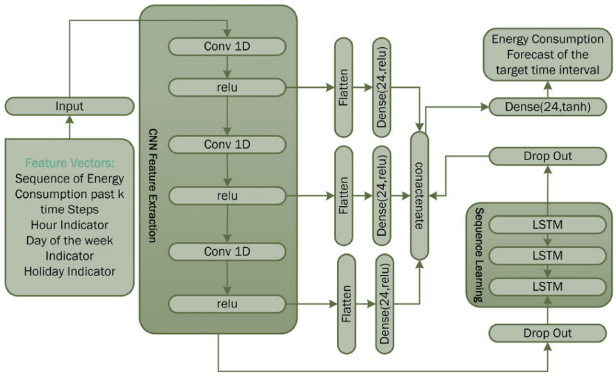


FIGURE 12. Proposed Model of the Network.

Following the third activation layer, a dropout rate of 0.25 is applied to mitigate the risk of overfitting. This regularization technique randomly sets a fraction of input units to zero during training, promoting robustness and generalization of the model. The model then transitions to a series of LSTM layers, beginning with an LSTM layer containing 20 units that return sequences. This layer is instrumental in capturing the temporal dependencies within the data, leveraging the LSTM's ability to retain and utilize information from previous time steps. A second LSTM layer with 20 units and returning sequences further processes the temporal information. The third LSTM layer, with 20 units, returns only the last output in the sequence, encapsulating the temporal dynamics into a fixed-size vector. Following the LSTM layers, another dropout layer with a dropout rate 0.25 is introduced to further reduce overfitting. The output from this layer is then concatenated with the outputs from the three dense layers following each convolutional block. This concatenation operation effectively combines the spatial features extracted by the CNN layers with the temporal features captured by the LSTM layers, providing a comprehensive representation of the input data. Finally, the concatenated features are passed through a dense layer with 24 or 1 node and a tanh activation function, producing the model's final output for single-step and multistep forecasting [79]. However, it has been modified to enhance the performance. The parameter setting of the formulated deep learning framework in this study is captured in Table 3.

It also assists them in reducing the number of overfitting and results in a smaller number of trainable parameters. Every stack included a sequence learning block, where exactly three LSTM layers were used, with 20 neurons per layer. The return sequence is set as the true for the first two LSTM layers to ensure the net makes out the total series of hidden states. Finally, the return sequence is false in the final LSTM layer: It is worth noting that in the final LSTM layer, the network gives only the hidden state (the last time step). Therefore, another dropout layer was invoked before the output layer to reduce cases of overlearning and high chances of overfitting. The input or fully connected layer comprises 20 neurons equivalent to 20 indexes of the irregular array used to cool the neural network during the simulation. The neurons in the output layers are from one to six to cover all the look-ahead periods of 1 up to 3 h for load forecasting. The parameter setting of the formulated deep learning framework in this study is captured in Table 4.

TABLE 3. Various layer configuration of the proposed model.

Proposed Model Parameters		
Convolution 1	Kernels	48
	Size of Receptive field	3
MaxPooling	-	-
tanh	-	-
Convolution 2	Kernels	32
	Size of Receptive field	3
MaxPooling	-	-
ReLU	-	-
Convolution 3	Kernels	16
	Size of Receptive field	3
MaxPooling	-	-
ReLU	-	-
Dropout 1	-	0.25
LSTM1	Hidden Node	20
	Return Sequence	True
LSTM2	Hidden Node	20
	Return Sequence	True
LSTM3	Hidden Node	20
	Return Sequence	True
Dropout 2	-	0.25
Fully Connected	Hidden Node	20
Output	Hidden Node	1/24

TABLE 4. Proposed model setting (Parameters).

Parameter	**Setting**
Optimizer	Adam
Loss function	Mean absolute Error (MAE)
Learning Rate	{0.001}
Learning Rate	Monitor=validation loss, patience=10 Epochs
Adjustment	Factor=0.8,minimum learning rate =1e^{-5}
Batch Size	{128}
Epoch	{150}

To this end, the distinguished optimizer tagged 'Adam' was deployed in this study due to its efficiency and supposedly optimal responsiveness to fluctuating learning rate requirements. To evaluate the accuracy of the mean absolute error, or MAE was utilized as the loss function by calculating the absolute value of the difference between the target and predicted values for each data point, summing all these values, and dividing the total by the overall number of values in a dataset.

B. DATA PRE-PROCESSING

For optimal learning, data must be meticulously pre-processed before model training. A well-prepared dataset enhances

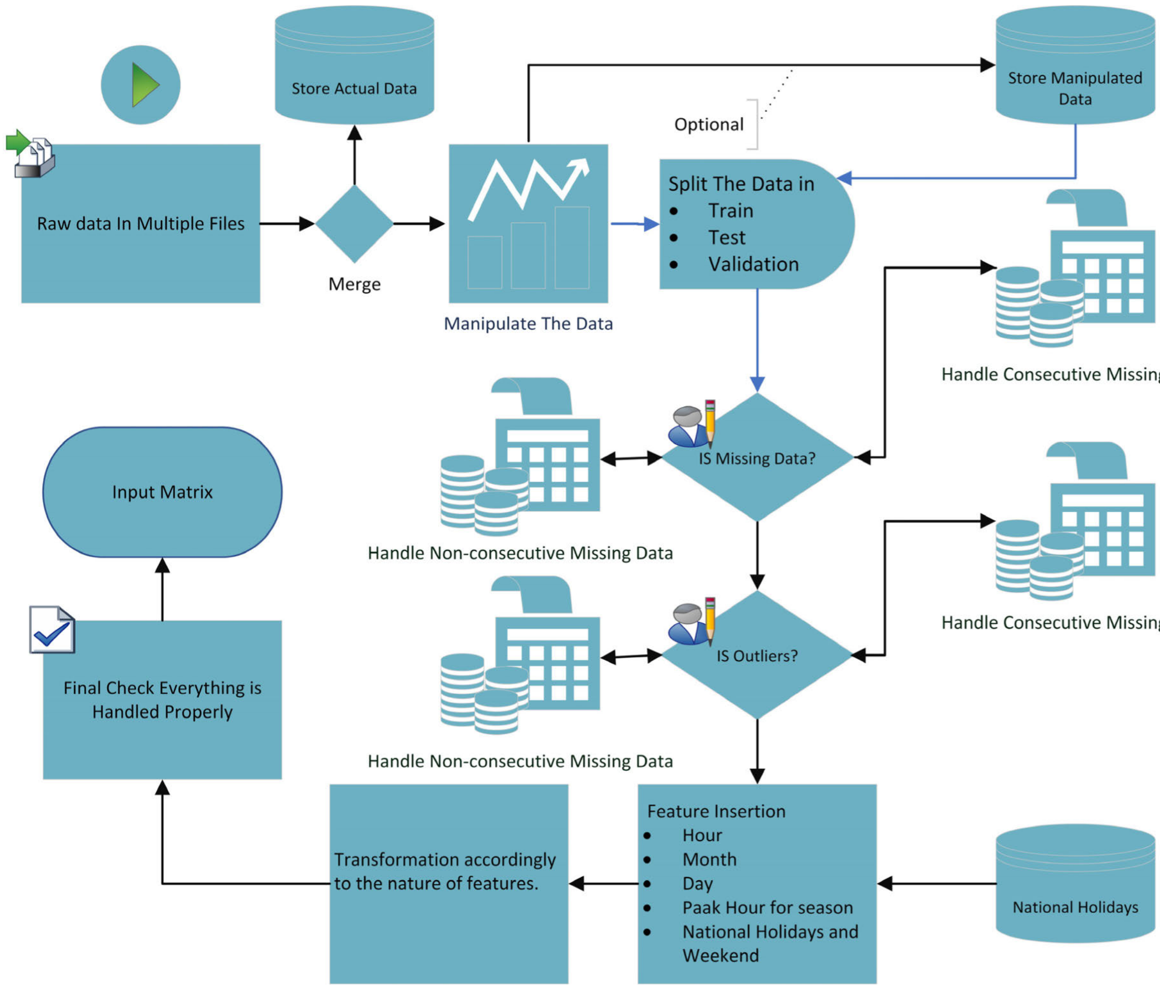


FIGURE 13. **Pre-processing diagram.**

a model's ability to identify trends and relationships. The primary pre-processing steps include train-test splitting, data cleaning, high-level feature extraction, and formatting the data into a 3D structure (Samples, Time Steps, and Features) suitable for model training. Figure 13 illustrates the comprehensive pre-processing flowchart. Utilizing the aforementioned model for pre-processing, the initial data was acquired from the National Transmission and Dispatch Company (NTDC) in Islamabad. This data was initially scattered across multiple files, which posed a challenge for effective handling and analysis. These disparate files were meticulously processed and merged to streamline the process into a cohesive file, facilitating more efficient data management and feature extraction. The comprehensive flowchart detailing the entire pre-processing procedure is presented in Figure 13.

This dataset encompasses a time range from January 1, 2019, at 00:00 hours to May 31, 2023, at 23:00 hours, with data recorded hourly by the National Power Control Center (NPCC). Several manipulation techniques were applied to ensure data privacy while preparing it for analysis. These m anipulations were executed to preserve the overall consumption patterns, introducing only minor adjustments to the data. This approach was essential to maintain the integrity and utility of the dataset for accurate feature extraction and subsequent model training. A detailed description of the dataset, including various attributes and corresponding values, is provided in Table 5.

TABLE 5. **Dataset description.**

Dataset	**Description**
Provided By	NPCC
Start Date	1/1/2019 0:00
End Date	5/31/2023 23:00
Resolution	Hourly
Missing Values	0
Outliers	16
Maximum Load	25792.34 MW
Minimum Load	MW

This comprehensive description clearly explains the data's scope and characteristics, ensuring that all relevant information is available for thorough analysis and model development. Subsequently, the data was divided into training and testing sets. The period from February 2023 to May 2023 was designated as the test data and extracted from the

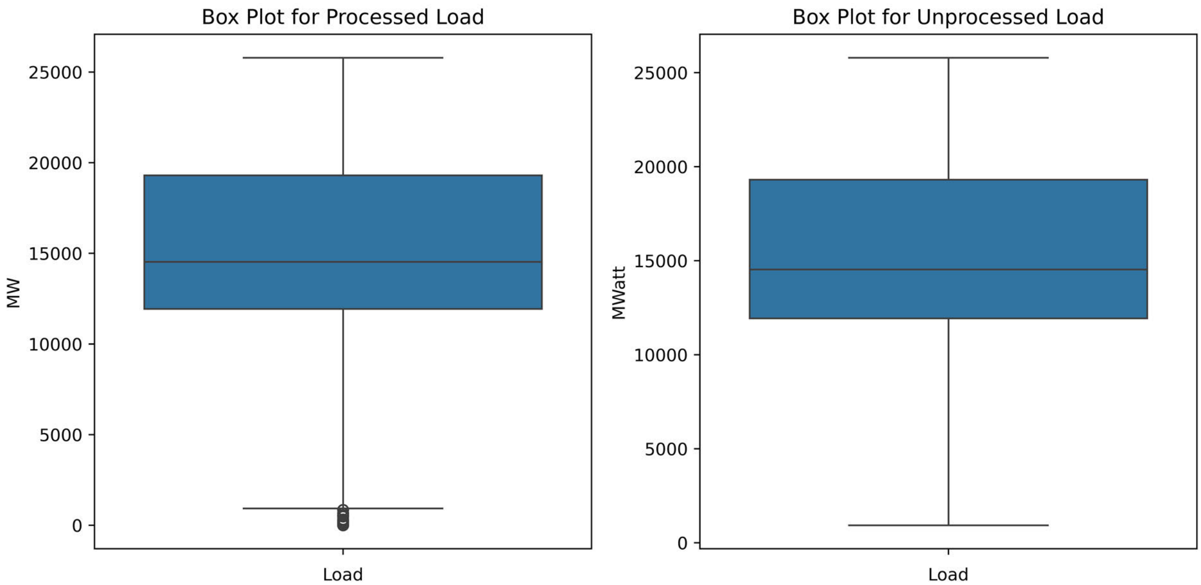


FIGURE 14. Outliers before and after handling.

entire dataset. The remaining data, spanning from January 2019 to January 2023, was partitioned such that 70 percent was utilized for training and 30 percent for validation.

During the analysis process, it was confirmed that there was no missing data; however, outliers were detected within the training dataset, as illustrated in Figure 14. These outliers were identified and addressed to ensure the integrity of the model training process. The analysis assumes no missing data was in the dataset while outliers were identified, especially during off-peak hours. The figure presents the load in megawatts (MW) by categories, including working days, holidays, off-peak, and peak hours during the summer peak period. The median load is slightly lower during holidays than during working days, but a similar IQR follows this and has no extreme points; therefore, the load is similar during both working days and holidays. Contrarily, the load during on-peak is much higher and more consolidated than off-peak, with several low values and higher variability. This implies a high and well-defined increase in energy demand during the on-peak hours. Therefore, there is a need to establish a better way of predicting the variations and stability during the off-peak hours. They are important for energy management, resource planning, and load forecast enhancement. Following this, all potential features that could influence the load consumption pattern were extracted, as depicted in the heat map in Figure 15. Additionally, this study treated all national holidays as weekend holidays to maintain consistency in the data analysis.

The outliers were identified using the following equations:

$$\text{Lower Outliers} = Min \times (Q_1 - 1.5 \times IQR) \tag{32}$$

$$\text{Upper Outliers} = \text{Max} \times (\text{Q}_3 + 1.5 \times \text{IQR}) \tag{33}$$

The outliers were managed similarly to consecutive missing values. Specifically, the data anomalies from 2021-01-02 22:00:00 to 2021-01-09 22:00:00 and from 2021-01-09 06:00:00 to 2021-01-16 06:00:00 were rectified by substituting them with the mean values computed over 24-hour periods, grouped by each hour. Before and after this treatment process, the comprehensive details of the outliers are delineated in Table 6.

TABLE 6. Detail description of outliers.

Date	**Outliers**	**Handled**
2021-01-09 23:00:00	0.0 MW	9885.54 MW
2021-01-10 00:00:00	50.4 MW	9225.72 MW
2021-01-10 01:00:00	44.1 MW	8699.84 MW
2021-01-10 02:00:00	196.35 MW	8509.91 MW
2021-01-10 03:00:00	553.35 MW	8334.55 MW
2021-01-10 04:00:00	256.2 MW	8561.45 MW
2021-01-10 05:00:00	851.55 MW	9661.33 MW
2023-01-23 07:00:00	0.0 MW	12706.38 MW
2023-01-23 08:00:00	148.05 MW	13314.66 MW
2023-01-23 09:00:00	238.35 WM	13267.94 MW
2023-01-23 10:00:00	134.4 MW	13046.51 MW
2023-01-23 11:00:00	321.3 MW	13168.62 MW
2023-01-23 12:00:00	281.4 MW	12943.17 MW
2023-01-23 13:00:00	554.4 MW	12766.35 MW
2023-01-23 14:00:00	549.15 MW	13106.28 WM
2023-01-23 15:00:00	486.15 MW	13397.85 MW
2023-01-23 16:00:00	664.65 MW	13093.40 MW

Various factors influencing the load consumption pattern were incorporated for feature extraction. These factors included the hour of the day, the month, and the weekdays, with their effects illustrated in Figures 16 and 17, respectively. Additionally, Pakistani national holidays were encoded as binary signals (0 for no holiday, 1 for a holiday), and weekends were included in the analysis.

The data was prepared for the deep learning network to accommodate single-step and multistep inputs. Algorithms for configuring single-step and multistep data sequences were delineated in our previous work. Notably, 17 input features

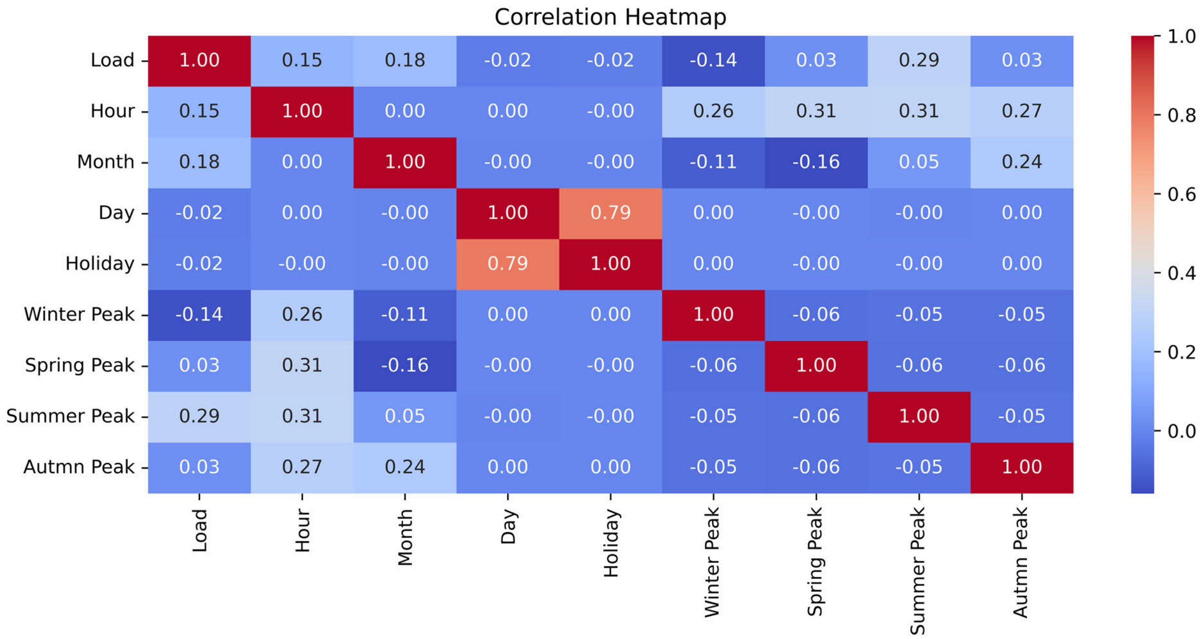


FIGURE 15. Correlation of Feature with Load.

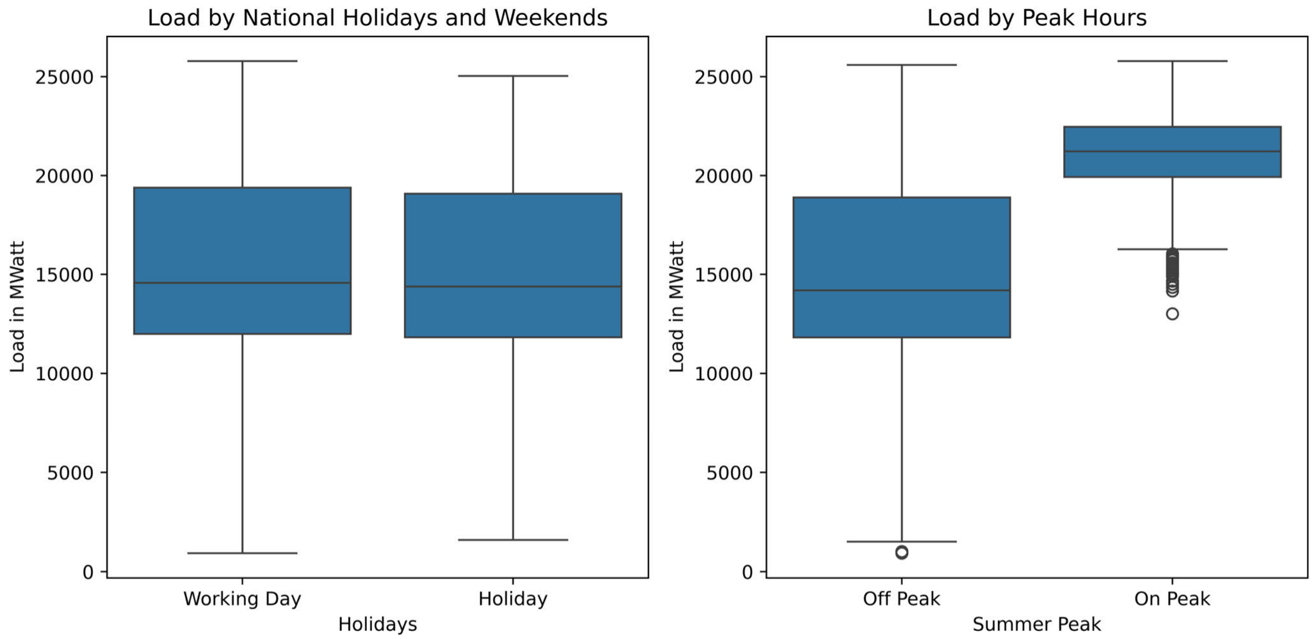


FIGURE 16. Categorical features.

were incorporated to train the CNN and LSTM models, as illustrated in Table 7. Accordingly, the performance evaluation metrics are presented in Tables 8 and 9, respectively.

C. RESULTS AND DISCUSSION

Hourly load forecasting is an essential component of energy management systems, ensuring the stability and efficiency of power networks. Accurate forecasts enable utilities to allocate resources and balance loads effectively, enhancing overall system performance. Recently, there has been an increasing interest in developing sophisticated forecasting models to improve the precision of hour-ahead load predictions. The actual power consumption and the predicted power from the proposed model are illustrated in Figure 18 for hour-ahead forecasting. This graph juxtaposes the actual load (in megawatts) with the predicted load across various dates. The x-axis denotes the dates from December 2020 to April 2021,

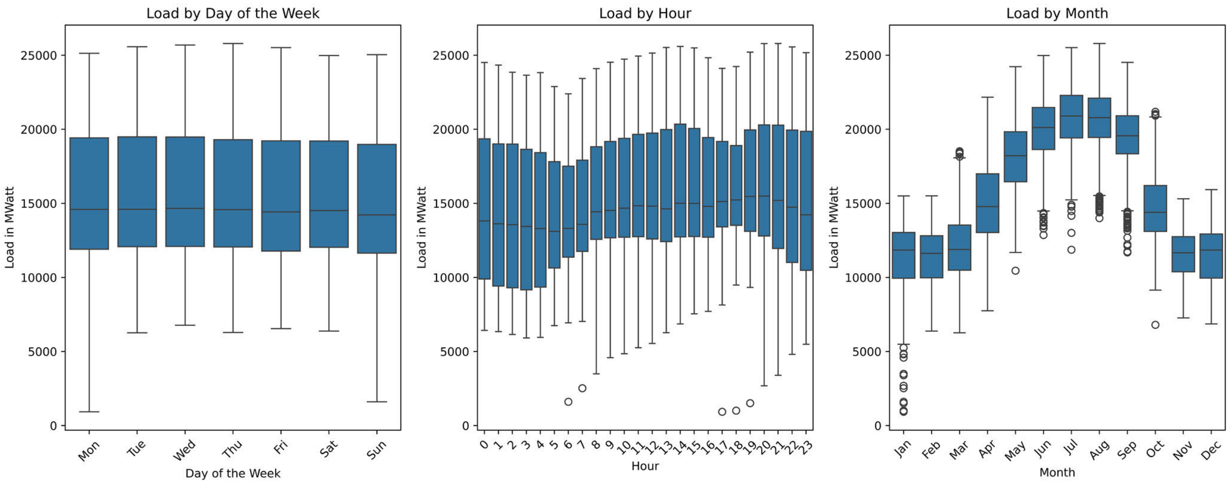


FIGURE 17. Cyclic features.

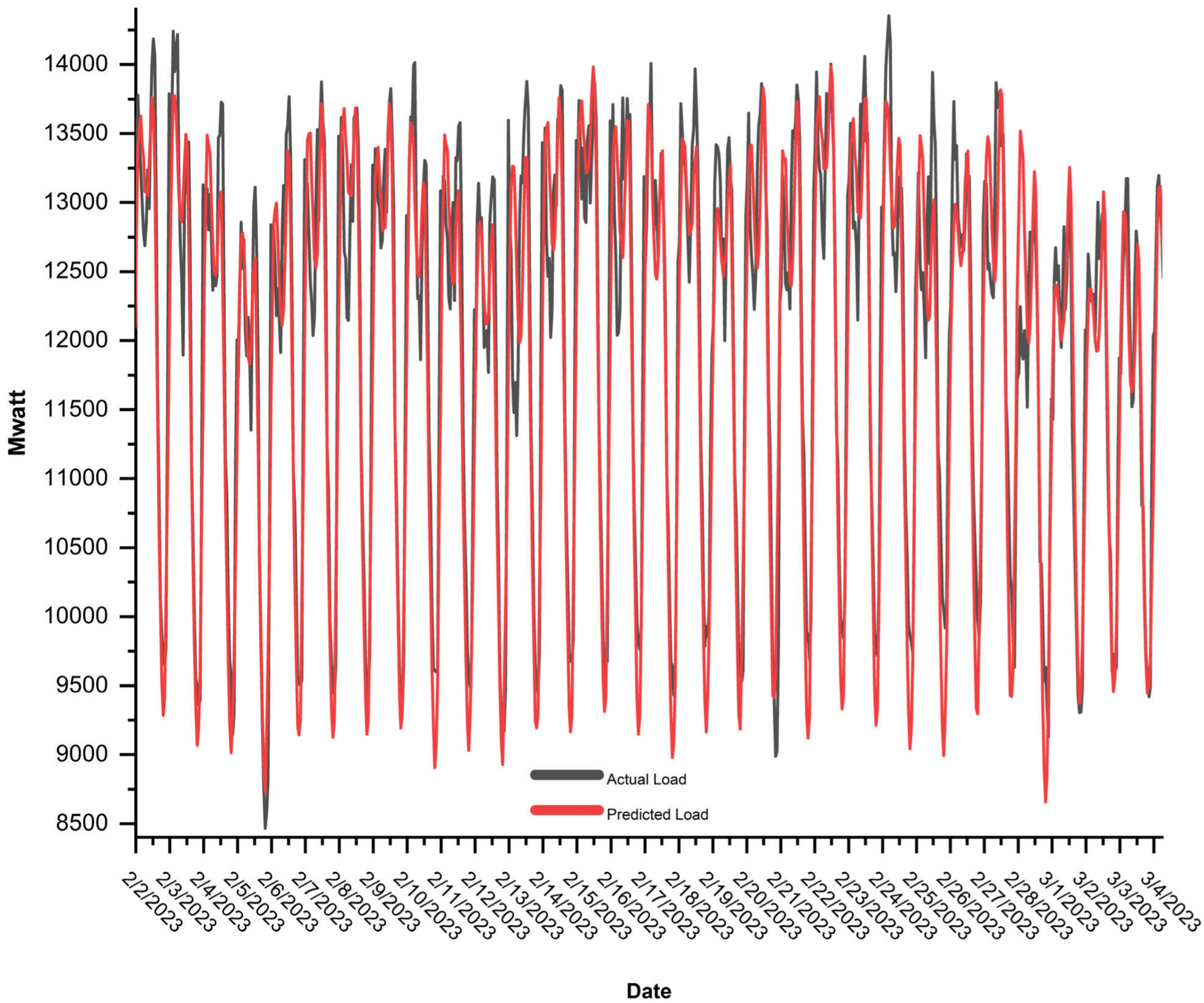


FIGURE 18. Actual and predicted load consumption.

while the y-axis measures the electrical load in megawatts (MW). The black line represents the actual load, and the predicted load is depicted in red. The graph indicates that the predicted load closely mirrors the actual load, suggesting that the prediction model has effectively captured the overall trend and most fluctuations in the load data. Although the prediction is conducted for the entire test dataset, only a few days from the beginning are shown for better visibility.

TABLE 7. Input features.

Features	Type	Encoding Approach	Shape
Load	Numerical	Min Max Normalization	1
Hour	Cyclic	Trigonometric Transform	2
Day			2
Month			2
Holiday	Non-Cyclic	One-Hot Encoding	2
Peak hours of Summer			2
Peak hours of Autumn			2
Peak hours of Winter			2
Peak hours of Spring			2

The input features are presented in Table 7. In contrast, the performance evaluation for the single step and 24 steps are presented in Comparing the performance of the forecasting models presented in [55] and [79], and the proposed model, it is possible to note the increase in the performance indicators of the proposed model for both single-step and 24-hour forecasting. In the case of single-step forecasting, the dataset of NTDC has demonstrated that the proposed model has a value of RMSE equal to 538. 71, MAE of 371. 97, and MAPE of 2. 72, outperforming [79]. Likewise, for the AEP dataset, the RMSE of the proposed model is 126. 35, MAE of 94. 27, and MAPE of 0. 64 are substantially better than [79] (RMSE: 372. 77, MAE: 295. 92, MAPE: 2. 05 and [55] RMSE: 472. 13, MAE: 358. 6, MAPE: 2. 41. In 24-hour forecasting, the proposed model also outperforms the other models with NTDC values of RMSE 951. 94, MAE 656. 35, and MAPE 4. 72, compared to [79] (RMSE: And in this case, the proposed model has a lower RMSE (1038. 04) than [55] (1166. 79), a lower MAE (773. 18) than [55] (843. 76), and a lower MAPE (5. 35) than [55] (6. 16). In the case of the AEP dataset, the RMSE of the proposed model is 702. 97, MAE of 507. 97, and MAPE of 3. 39 also outperform [79]. In general, the proposed model outperforms the benchmark models in terms of accuracy and reliability in one-step and n-step ahead forecasting.

TABLE 8. Performance evaluation matrix for single-step.

S. No	NTDC			AEP		
Model	RMSE	MAE	MAPE	RMSE	MAE	MAPE
Ref [79]	899.04	657.07	4.77	372.77	295.92	2.05
Ref [55]	1004.54	733.21	5.33	472.13	358.6	2.41
Proposed	538.71	371.97	2.72	126.35	94.27	0.64

TABLE 9. Performance evaluation matrix for 24 hours.

S. No	NTDC			AEP		
Model	RMSE	MAE	MAPE	RMSE	MAE	MAPE
Ref [79]	1038.04	773.18	5.35	819.86	632.36	4.25
Ref [55]	1166.79	843.76	6.16	870.33	654	4.36
Proposed	951.94	656.35	4.72	702.97	507.97	3.39

Likewise, how they compare is shown in Figure 19. It can be confirmed that the values of the proposed model's predictions (red line) are closer to the actual load values (black

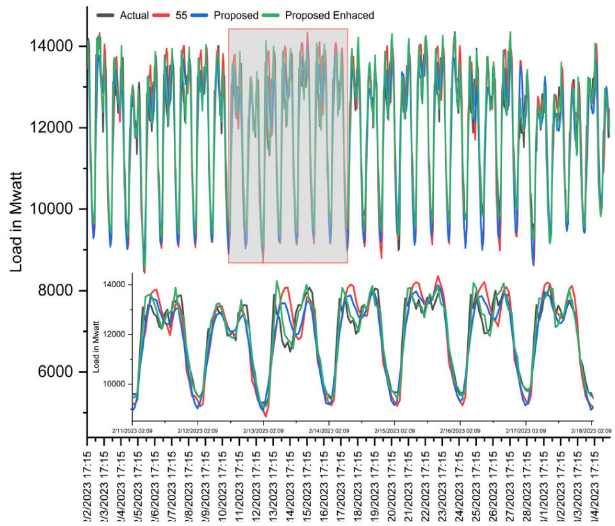


FIGURE 19. Comparison with the reference model.

line) than the values of predictions defined by the reference method (blue line). The data utilized in this case was NTDC, which serves as the study's baseline. This shows that the adopted model has given a better, more accurate load forecast in MWatt over the stipulated period. The reference method has high variability to the actual values, as evidenced by Fig. 19, which explains this method's fairly poor predictive power.

V. CONCLUSION

Short-term load forecasting (STLF) is crucial in large-scale power grids, enabling grid operators to accurately forecast load and efficiently utilize operating reserves. This paper explains the latest STLF techniques, including statistical models such as autoregressive integrated moving averages (ARIMA), seasonal ARIMA (SARIMA), exponential smoothing, and generalized linear models. It also includes smart techniques such as Support vector machine (SVM), decision tree, random forest, gradient boosting, MLP, ENSEMBLE, and hybrid models. These methods are then assessed in the paper through mathematical and graphical analysis, and their advantages and drawbacks are discussed to assist with their proper implementation in present day power systems. Further, the paper has proposed a hybrid CNN-LSTM model for STLF, which combines Convolutional Neural Networks (CNNs) for feature extraction and Long Short-Term Memory networks (LSTMs) for sequence forecasting and uses textual data from the NTDC Pakistan national power grid. The results of the comparative analysis with the previous models using the AEP dataset prove that the proposed hybrid model has higher accuracy in single-step and 24-hour forecasts, thus improving the forecast quality and generalizing the model's performance for various forecast horizons.

In the future, this study can be extended to probabilistic load forecasting to improve the stability and accuracy of the forecast. Probabilistic forecasting measures the degree of uncertainty by providing the range of possible and forecasted values, which is crucial in risk management and decision-making in power systems. This includes incorporating techniques like quantile regression, Monte Carlo simulation, and Bayesian analysis. These approaches would give more context-aware information about expected deviations or risks in load forecasts, enhancing the accuracy of the forecasts and the decision-making process in power grid management.

REFERENCES

[1] L. Cheng and T. Yu, "A new generation of AI: A review and perspective on machine learning technologies applied to smart energy and electric power systems," *Int. J. Energy Res.*, vol. 43, no. 6, pp. 1928–1973, May 2019.

[2] Y. Ding, Y. Zhu, J. Feng, P. Zhang, and Z. Cheng, "Interpretable spatio-temporal attention LSTM model for flood forecasting," *Neurocomputing*, vol. 403, pp. 348–359, Aug. 2020.

[3] H. Fu, J.-C. Baltazar, and D. E. Claridge, "Review of developments in whole-building statistical energy consumption models for commercial buildings," *Renew. Sustain. Energy Rev.*, vol. 147, Sep. 2021, Art. no. 111248.

[4] S. Lu, Q. Li, L. Bai, and R. Wang, "Performance predictions of ground source heat pump system based on random forest and back propagation neural network models," *Energy Convers. Manage.*, vol. 197, Oct. 2019, Art. no. 111864.

[5] M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, and P. N. Suganthan, "Ensemble deep learning: A review," *Eng. Appl. Artif. Intell.*, vol. 115, Oct. 2022, Art. no. 105151.

[6] A. Altan, S. Karasu, and E. Zio, "A new hybrid model for wind speed forecasting combining long short-term memory neural network, decomposition methods and grey wolf optimizer," *Appl. Soft Comput.*, vol. 100, Mar. 2021, Art. no. 106996.

[7] M. Alexander and H. Beushausen, "Durability, service life prediction, and modelling for reinforced concrete structures—Review and critique," *Cement Concrete Res.*, vol. 122, pp. 17–29, Aug. 2019.

[8] A. Kurani, P. Doshi, A. Vakharia, and M. Shah, "A comprehensive comparative study of artificial neural network (ANN) and support vector machines (SVM) on stock forecasting," *Ann. Data Sci.*, vol. 10, no. 1, pp. 183–208, Feb. 2023.

[9] A. S. Dagoumas and N. E. Koltsaklis, "Review of models for integrating renewable energy in the generation expansion planning," *Appl. Energy*, vol. 242, pp. 1573–1587, May 2019.

[10] L.-L. Li, X. Zhao, M.-L. Tseng, and R. R. Tan, "Short-term wind power forecasting based on support vector machine with improved dragonfly algorithm," *J. Cleaner Prod.*, vol. 242, Jan. 2020, Art. no. 118447.

[11] J. Chen and X. Ran, "Deep learning with edge computing: A review," *Proc. IEEE*, vol. 107, no. 8, pp. 1655–1674, Aug. 2019.

[12] A. Ruano, A. Hernandez, J. Ureña, M. Ruano, and J. Garcia, "NILM techniques for intelligent home energy management and ambient assisted living: A review," *Energies*, vol. 12, no. 11, p. 2203, Jun. 2019.

[13] S. M. Hasanat, R. Younis, S. Alahmari, M. T. Ejaz, M. Haris, H. Yousaf, S. Watara, K. Ullah, and Z. Ullah, "Enhancing load forecasting accuracy in smart grids: A novel parallel multichannel network approach using 1D CNN and Bi-LSTM models," *Int. J. Energy Res.*, vol. 2024, no. 1, Jan. 2024, Art. no. 2403847.

[14] M. Sharifzadeh, A. Sikinioti-Lock, and N. Shah, "Machine-learning methods for integrated renewable power generation: A comparative study of artificial neural networks, support vector regression, and Gaussian process regression," *Renew. Sustain. Energy Rev.*, vol. 108, pp. 513–538, Jul. 2019.

[15] H. Liu, C. Chen, X. Lv, X. Wu, and M. Liu, "Deterministic wind energy forecasting: A review of intelligent predictors and auxiliary methods," *Energy Convers. Manage.*, vol. 195, pp. 328–345, Sep. 2019.

[16] A. Nespoli, E. Ogliari, S. Leva, A. Massi Pavan, A. Mellit, V. Lughi, and A. Dolara, "Day-ahead photovoltaic forecasting: A comparison of the most effective techniques," *Energies*, vol. 12, no. 9, p. 1621, Apr. 2019.

[17] M. Mahzarnia, M. P. Moghaddam, P. T. Baboli, and P. Siano, "A review of the measures to enhance power systems resilience," *IEEE Syst. J.*, vol. 14, no. 3, pp. 4059–4070, Sep. 2020.

[18] Z. Liu, P. Jiang, L. Zhang, and X. Niu, "A combined forecasting model for time series: Application to short-term wind speed forecasting," *Appl. Energy*, vol. 259, Feb. 2020, Art. no. 114137.

[19] C. Fan, J. Wang, W. Gang, and S. Li, "Assessment of deep recurrent neural network-based strategies for short-term building energy predictions," *Appl. Energy*, vol. 236, pp. 700–710, Feb. 2019.

[20] E. Hossain, I. Khan, F. Un-Noor, S. S. Sikander, and M. S. H. Sunny, "Application of big data and machine learning in smart grid, and associated security concerns: A review," *IEEE Access*, vol. 7, pp. 13960–13988, 2019.

[21] Y. Eren and I. Küçükdemiral, "A comprehensive review on deep learning approaches for short-term load forecasting," *Renew. Sustain. Energy Rev.*, vol. 189, Jan. 2024, Art. no. 114031.

[22] H. Hassani and E. S. Silva, "Predictions from generative artificial intelligence models: Towards a new benchmark in forecasting practice," *Information*, vol. 15, no. 6, p. 291, May 2024.

[23] M. H. Sulaiman and Z. Mustaffa, "Chiller energy prediction in commercial building: A metaheuristic-enhanced deep learning approach," *Energy*, vol. 297, Jun. 2024, Art. no. 131159.

[24] D. Hou and R. Evins, "A protocol for developing and evaluating neural network-based surrogate models and its application to building energy prediction," *Renew. Sustain. Energy Rev.*, vol. 193, Apr. 2024, Art. no. 114283.

[25] K. Wang, X. Qi, and H. Liu, "Photovoltaic power forecasting based LSTM-convolutional network," *Energy*, vol. 189, Dec. 2019, Art. no. 116225.

[26] S. Du, T. Li, Y. Yang, and S.-J. Horng, "Multivariate time series forecasting via attention-based encoder–decoder framework," *Neurocomputing*, vol. 388, pp. 269–279, May 2020.

[27] A. Mosavi, M. Salimi, S. F. Ardabili, T. Rabczuk, S. Shamshirband, and A. R. Varkonyi-Koczy, "State of the art of machine learning models in energy systems, a systematic review," *Energies*, vol. 12, no. 7, p. 1301, Apr. 2019.

[28] M. Cai, M. Pipattanasomporn, and S. Rahman, "Day-ahead building-level load forecasts using deep learning vs. traditional time-series techniques," *Appl. Energy*, vol. 236, pp. 1078–1088, Feb. 2019.

[29] S. Sobri, S. Koohi-Kamali, and N. A. Rahim, "Solar photovoltaic generation forecasting methods: A review," *Energy Convers. Manage.*, vol. 156, pp. 459–497, Jan. 2018.

[30] R. Ahmed, V. Sreeram, Y. Mishra, and M. D. Arif, "A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization," *Renew. Sustain. Energy Rev.*, vol. 124, May 2020, Art. no. 109792.

[31] C. Tarmanini, N. Sarma, C. Gezegin, and O. Ozgonenel, "Short term load forecasting based on ARIMA and ANN approaches," *Energy Rep.*, vol. 9, pp. 550–557, May 2023.

[32] N. Kalchbrenner, I. Danihelka, and A. Graves, "Grid long short-term memory," 2015, *arXiv:1507.01526*.

[33] K. T. Ponds, A. Arefi, A. Sayigh, and G. Ledwich, "Aggregator of demand response for renewable integration and customer engagement: Strengths, weaknesses, opportunities, and threats," *Energies*, vol. 11, no. 9, p. 2391, Sep. 2018.

[34] G. Fan, Y. Guo, J. Zheng, and W. Hong, "A generalized regression model based on hybrid empirical mode decomposition and support vector regression with back-propagation neural network for mid-short-term load forecasting," *J. Forecasting*, vol. 39, no. 5, pp. 737–756, Aug. 2020.

[35] A. S. Nair, P. Ranganathan, C. Finley, and N. Kaabouch, "Short-term forecast analysis on wind power generation data," in *Proc. IEEE Kansas Power Energy Conf. (KPEC)*, Apr. 2021, pp. 1–6.

[36] L. Shang, K. Chen, G. Wang, Y. Liu, R. Hu, and Y. Shang, "Short-term distribution network peak load forecasting based on generalized linear model," in *Proc. 4th Int. Conf. Power Energy Technol. (ICPET)*, Jul. 2022, pp. 584–589.

[37] A. Pierrot and Y. Goude, "Short-term electricity load forecasting with generalized additive models," *Proc. ISAP Power*, 2011, pp. 1–6.

[38] Y. Liang, D. Niu, and W.-C. Hong, "Short term load forecasting based on feature extraction and improved general regression neural network model," *Energy*, vol. 166, pp. 653–663, Jan. 2019.

[39] G. Hafeez, K. S. Alimgeer, and I. Khan, "Electric load forecasting based on deep learning and optimized by heuristic algorithm in smart grid," *Appl. Energy*, vol. 269, Jul. 2020, Art. no. 114915.

[40] Y. Wang, D. Gan, M. Sun, N. Zhang, Z. Lu, and C. Kang, "Probabilistic individual load forecasting using pinball loss guided LSTM," *Appl. Energy*, vol. 235, pp. 10–20, Feb. 2019.

[41] A. Ahmed and M. Khalid, "A review on the selected applications of forecasting models in renewable power systems," *Renew. Sustain. Energy Rev.*, vol. 100, pp. 9–21, Feb. 2019.
[42] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 841–851, Jan. 2019.
[43] M. Barman and N. B. Dev Choudhury, "Season specific approach for short-term load forecasting based on hybrid FA-SVM and similarity concept," *Energy*, vol. 174, pp. 886–896, May 2019.
[44] S. Aslam, H. Herodotou, S. M. Mohsin, N. Javaid, N. Ashraf, and S. Aslam, "A survey on deep learning methods for power load and renewable energy forecasting in smart microgrids," *Renew. Sustain. Energy Rev.*, vol. 144, Jul. 2021, Art. no. 110992.
[45] R. Wazirali, E. Yaghoubi, M. S. S. Abujazar, R. Ahmad, and A. H. Vakili, "State-of-the-art review on energy and load forecasting in microgrids using artificial neural networks, machine learning, and deep learning techniques," *Electric Power Syst. Res.*, vol. 225, Dec. 2023, Art. no. 109792.
[46] N. Ghadimi, A. Akbarimajd, H. Shayeghi, and O. Abedinia, "Two stage forecast engine with feature selection technique and improved metaheuristic algorithm for electricity load forecasting," *Energy*, vol. 161, pp. 130–142, Oct. 2018.
[47] F. He, J. Zhou, Z.-K. Feng, G. Liu, and Y. Yang, "A hybrid short-term load forecasting model based on variational mode decomposition and long short-term memory networks considering relevant factors with Bayesian optimization algorithm," *Appl. Energy*, vol. 237, pp. 103–116, Mar. 2019.
[48] G. Dudek, "A comprehensive study of random forest for short-term load forecasting," *Energies*, vol. 15, no. 20, p. 7547, Oct. 2022.
[49] S. H. Rafi and M. M. Mahdi, "A short-term load forecasting technique using extreme gradient boosting algorithm," in *Proc. IEEE PES Innov. Smart Grid Technol.*, Dec. 2021, pp. 1–5.
[50] K. Zhu, J. Geng, and K. Wang, "A hybrid prediction model based on pattern sequence-based matching method and extreme gradient boosting for holiday load forecasting," *Electric Power Syst. Res.*, vol. 190, Jan. 2021, Art. no. 106841.
[51] J. Kim, J. Moon, E. Hwang, and P. Kang, "Recurrent inception convolution neural network for multi short-term load forecasting," *Energy Buildings*, vol. 194, pp. 328–341, Jul. 2019.
[52] A. Yang, W. Li, and X. Yang, "Short-term electricity load forecasting based on feature selection and least squares support vector machines," *Knowl.-Based Syst.*, vol. 163, pp. 159–173, Jan. 2019.
[53] P. Du, J. Wang, W. Yang, and T. Niu, "A novel hybrid model for short-term wind power forecasting," *Appl. Soft Comput.*, vol. 80, pp. 93–106, Jul. 2019.
[54] A.-D. Pham, N.-T. Ngo, T. T. Ha Truong, N.-T. Huynh, and N.-S. Truong, "Predicting energy consumption in multiple buildings using machine learning for improving energy efficiency and sustainability," *J. Cleaner Prod.*, vol. 260, Jul. 2020, Art. no. 121082.
[55] S. H. Rafi, S. R. Deeba, and E. Hossain, "A short-term load forecasting method using integrated CNN and LSTM network," *IEEE Access*, vol. 9, pp. 32436–32448, 2021.
[56] L. Zhang, J. Wen, Y. Li, J. Chen, Y. Ye, Y. Fu, and W. Livingood, "A review of machine learning in building load prediction," *Appl. Energy*, vol. 285, Mar. 2021, Art. no. 116452.
[57] M. Tan, S. Yuan, S. Li, Y. Su, H. Li, and F. He, "Ultra-short-term industrial power demand forecasting using LSTM based hybrid ensemble learning," *IEEE Trans. Power Syst.*, vol. 35, no. 4, pp. 2937–2948, Jul. 2020.
[58] M. Bourdeau, X. Q. Zhai, E. Nefzaoui, X. Guo, and P. Chatellier, "Modeling and forecasting building energy consumption: A review of data-driven techniques," *Sustain. Cities Soc.*, vol. 48, Jul. 2019, Art. no. 101533.
[59] K. Yan, W. Li, Z. Ji, M. Qi, and Y. Du, "A hybrid LSTM neural network for energy consumption forecasting of individual households," *IEEE Access*, vol. 7, pp. 157633–157642, 2019.
[60] M. Massaoudi, S. S. Refaat, I. Chihi, M. Trabelsi, F. S. Oueslati, and H. Abu-Rub, "A novel stacked generalization ensemble-based hybrid LGBM-XGB-MLP model for short-term load forecasting," *Energy*, vol. 214, Jan. 2021, Art. no. 118874.
[61] J. F. Bermejo, J. F. G. Fernández, F. O. Polo, and A. C. Márquez, "A review of the use of artificial neural network models for energy and reliability prediction. A study of the solar PV, hydraulic and wind energy sources," *Appl. Sci.*, vol. 9, no. 9, p. 1844, May 2019.
[62] S. Akhtar, S. Shahzad, A. Zaheer, H. S. Ullah, H. Kilic, R. Gono, M. Jasinski, and Z. Leonowicz, "Short-term load forecasting models: A review of challenges, progress, and the road ahead," *Energies*, vol. 16, no. 10, p. 4060, May 2023.
[63] P. Li, K. Zhou, X. Lu, and S. Yang, "A hybrid deep learning model for short-term PV power forecasting," *Appl. Energy*, vol. 259, Feb. 2020, Art. no. 114216.
[64] S. O. Abter, S. M. Jameel, H. M. Majeed, and A. H. Sabry, "Intelligent forecasting temperature measurements of solar PV cells using modified recurrent neural network," *EUREKA, Phys. Eng.*, vol. 3, no. 3, pp. 169–177, 2024, doi: 10.21303/2461-4262.2024.003354.
[65] J. F. Torres, D. Hadjout, A. Sebaa, F. Martínez-Álvarez, and A. Troncoso, "Deep learning for time series forecasting: A survey," *Big Data*, vol. 9, no. 1, pp. 3–21, Feb. 2021.
[66] X. Wang, Z. Yao, and M. Papaefthymiou, "A real-time electrical load forecasting and unsupervised anomaly detection framework," *Appl. Energy*, vol. 330, Jan. 2023, Art. no. 120279.
[67] N. Huang, S. Wang, R. Wang, G. Cai, Y. Liu, and Q. Dai, "Gated spatial–temporal graph neural network based short-term load forecasting for wide-area multiple buses," *Int. J. Electr. Power Energy Syst.*, vol. 145, Feb. 2023, Art. no. 108651.
[68] Y. K. Semero, J. Zhang, and D. Zheng, "EMD–PSO–ANFIS-based hybrid approach for short-term load forecasting in microgrids," *IET Gener., Transmiss. Distrib.*, vol. 14, no. 3, pp. 470–475, Feb. 2020.
[69] M. Luy, V. Ates, N. Barisci, H. Polat, and E. Cam, "Short-term fuzzy load forecasting model using genetic–fuzzy and ant colony–fuzzy knowledge base optimization," *Appl. Sci.*, vol. 8, no. 6, p. 864, May 2018.
[70] D. Yang, J.-E. Guo, Y. Li, S. Sun, and S. Wang, "Short-term load forecasting with an improved dynamic decomposition-reconstruction-ensemble approach," *Energy*, vol. 263, Jan. 2023, Art. no. 125609.
[71] Z. Wang, T. Hong, and M. A. Piette, "Building thermal load prediction through shallow machine learning and deep learning," *Appl. Energy*, vol. 263, Apr. 2020, Art. no. 114683.
[72] S. Bouktif, A. Fiaz, A. Ouni, and M. A. Serhani, "Multi-sequence LSTM-RNN deep learning and metaheuristics for electric load forecasting," *Energies*, vol. 13, no. 2, p. 391, Jan. 2020.
[73] L. Wen, K. Zhou, S. Yang, and X. Lu, "Optimal load dispatch of community microgrid with deep learning based solar power and load forecasting," *Energy*, vol. 171, pp. 1053–1065, Mar. 2019.
[74] M. Sun, T. Zhang, Y. Wang, G. Strbac, and C. Kang, "Using Bayesian deep learning to capture uncertainty for residential net load forecasting," *IEEE Trans. Power Syst.*, vol. 35, no. 1, pp. 188–201, Jan. 2020.
[75] B. Nepal, M. Yamaha, A. Yokoe, and T. Yamaji, "Electricity load forecasting using clustering and ARIMA model for energy management in buildings," *Jpn. Architectural Rev.*, vol. 3, no. 1, pp. 62–76, Jan. 2020.
[76] T. Hong, J. Xie, and J. Black, "Global energy forecasting competition 2017: Hierarchical probabilistic load forecasting," *Int. J. Forecasting*, vol. 35, no. 4, pp. 1389–1399, Oct. 2019.
[77] Z. Zhang and W.-C. Hong, "Electric load forecasting by complete ensemble empirical mode decomposition adaptive noise and support vector regression with quantum-based dragonfly algorithm," *Nonlinear Dyn.*, vol. 98, no. 2, pp. 1107–1136, Oct. 2019.
[78] I. Shah, H. Iftikhar, S. Ali, and D. Wang, "Short-term electricity demand forecasting using components estimation technique," *Energies*, vol. 12, no. 13, p. 2532, Jul. 2019.
[79] M. Alhussein, K. Aurangzeb, and S. I. Haider, "Hybrid CNN-LSTM model for short-term individual household load forecasting," *IEEE Access*, vol. 8, pp. 180544–180557, 2020.

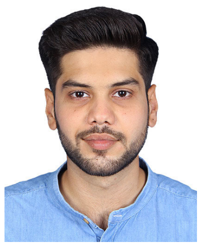
KALEEM ULLAH received the Ph.D. degree in electrical engineering (power) from the University of Engineering and Technology, Peshawar. He is currently an accomplished Electrical Engineer specializing in power systems. During the Ph.D. study, he conducted extensive research on real-time dynamic dispatch strategies for automatic generation control (AGC) and load forecasting mechanisms focusing on integrating large-scale RES power into the grid while maintaining active power balance and enhancing system reliability. USAID Pakistan funded his work through an applied research grant. He secured a significant grant from Pakistan Science Foundation for his research project on load dispatch control for Pakistan Power System.

MUHAMMAD AHSAN received the B.Eng. degree in electrical engineering from the NED University of Engineering and Technology and the master's degree in data science from the University of Alabama at Birmingham. He is equipped with a strong foundation in electrical engineering and data science. His dedication to his field is evident through his commitment to staying updated with the latest advancements in data science and its applications. His research interests include the intersection of data science and electrical engineering, where he aims to develop innovative solutions to complex problems.

SYED MUHAMMAD HASANAT received the B.Sc. degree in electrical engineering from the University of Engineering and Technology, Peshawar. He is currently pursuing the master's degree in electrical energy system engineering with Center for Advanced Studies in Energy, UET Peshawar. He is focusing his research efforts on the critical area of short-term load forecasting. He is dedicated to addressing the challenges and complexities inherent in energy systems management. His scholarly pursuits have garnered recognition, with his work being published in numerous impactful journals in recent years.

MUHAMMAD HARIS received the B.Sc. degree in mathematics and economics from Lahore University of Management Sciences, in 2021. He is currently pursuing the M.S. degree in data science and analytics with Georgia State University. He is also a Graduate Research Assistant at Georgia State University, collaborating with Truist Bank's ESG and Climate Data and Analytics Team. He has significant experience in data science, having worked at Afiniti Inc., where he optimized customer retention and developed predictive models. In this role, he applies advanced NLP techniques and generative AI to enhance climate-related financial disclosures. His research interests include machine learning, LSTM networks, and natural language processing (NLP).

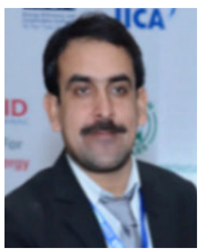
HAMZA YOUSAF received the B.B.A. degree (Hons.) from Lahore School of Economics, Pakistan, in 2016, and the M.S. degree in business analytics from the Stetson-Hatcher School of Business, Mercer University, Macon, GA, USA, in 2024. He was an Analyst in various multinational corporations, leveraging his data science and analytics expertise. His research interests include machine learning, deep learning, predictive modeling, sentiment analysis, and anomaly detection.



SYED FARAZ RAZA received the B.Eng. degree in electronic engineering from the NED University of Engineering and Technology, Karachi. He is currently pursuing the master's degree in data science from the University of Alabama at Birmingham. Demonstrating a commitment to academic excellence and a passion for advancing his knowledge, he is pursuing the master's degree. He is focusing his skill set on machine learning and artificial intelligence to support the research efforts in the critical area of short-term load forecasting.

RITESH TANDON received the master's degree in data science from Indiana University, Bloomington, IN, USA. He is currently an accomplished Lead Data Scientist. With a career spanning over two decades in the information technology industry, he has been at the forefront of the machine learning and artificial intelligence revolution. Since 2019, he has been with PLIC, where his passion for AI has driven him to design and implement cutting-edge data software products tailored for the insurance industry. His expertise in developing machine learning models has been instrumental in optimizing costs and mitigating risks for his company. His work in NLP has led to innovative solutions for document classification and question-answering models. In addition, he has significantly contributed to speech-to-text technology, extracting valuable insights from spoken language. Dedicated to advancing AI, he is committed to architecting and implementing intelligent solutions that push the boundaries of technology.

SAMAIN ABID received the B.Sc. degree in computer science from the Sir Syed University of Engineering and Technology. He is currently pursuing the master's degree in data science with the University of Alabama at Birmingham. His research primarily focuses on machine learning predictive models, where he aims to solve complex problems in data analytics. His dedication to his field is evident through his innovative approach to addressing the challenges in data science.



ZAHID ULLAH (Graduate Student Member, IEEE) received the B.S. degree in electrical engineering from UET Peshawar, in 2014, and the M.S. degree in electrical engineering from COMSATS University Islamabad, Abbottabad Campus, Abbottabad, Pakistan, in 2017. He is currently pursuing the Ph.D. degree in electrical engineering with Politecnico di Milano, Italy. He was a Lecturer with UMT Lahore, Pakistan. His research interests include smart grids, energy management, renewable energy systems, AI for energy, and ICTs for power systems.

Open Access funding provided by 'Politecnico di Milano' within the CRUI CARE Agreement