IET Cyber-Physical Systems: Theory & Applications

WILEY

The Institution of Engineering and Technology

ORIGINAL RESEARCH OPEN ACCESS

Cyber Risk Identification and Classification-Based Load Forecasting Tool for Pandemic Situations

Kuldeep Singh Shivran[1] | Kyle Swire-Thompson[2] | Neetesh Saxena[2] | Sarasij Das[1]

[1]Department of Electrical Engineering, Indian Institute of Science, Bangalore, India | [2]School of Computer Science and Informatics, Cardiff University, Cardiff, UK

Correspondence: Neetesh Saxena (saxenan4@cardiff.ac.uk)

Received: 9 July 2024 | **Revised:** 10 January 2025 | **Accepted:** 31 March 2025

Handling Editor: Rocco Zaccagnino

Funding: This research was supported by the EPSRC IAA research funding through Cardiff University.

Keywords: computer network security | power system planning | power system reliability | power system security

ABSTRACT

Smart grid operators use load forecasting algorithms to predict energy load for the reliable and economical operation of the electricity grid. COVID-19 pandemic-like situations (PLS) can significantly impact energy load demand due to uncertainties in factors such as regulatory orders, pandemic severity and human behavioural patterns. Additionally, in a smart grid, cyberattacks can manipulate forecasted load data, leading to suboptimal decisions, economic losses and potential blackouts. Forecasting load during these situations is challenging for traditional load forecasting tools, as they struggle to identify cyberattacks amidst uncertain load demand, where cyberattacks may mimic pandemic-like load patterns. Traditional forecasting methods do not incorporate factors related to pandemics and cyberattacks. Recent studies have focused on forecasting by considering factors such as COVID-19 cases, social distancing, weather, and temperature but fail to account for the impact of regulatory orders and pandemic severity. They also lack the ability to differentiate between normal and anomalous forecasts and classify the type of attack in anomalous data. This paper presents a tool for short-term load forecasting, anomaly detection and cyberattack classification for pandemic-like situations (PLS). The proposed short-term load forecasting algorithm uses a weighted moving average and an adjustment factor incorporating regulatory orders and pandemic severity, making it computationally efficient and deterministic. Additionally, the proposed anomaly detection and cyberattack classification algorithm provides robust options for detecting anomalies and classifying various types of cyberattacks. The proposed tool has been evaluated using K-Fold cross-validation to improve generalisability and reduce overfitting. The mean squared error (MSE) was used to measure prediction accuracy and detect discrepancies. It has been analysed and tested on real-load data from the State Load Dispatch Centre (SLDC), Delhi, of the Indian National Grid.

1 | Introduction

The electricity grid is evolving into a smart grid, integrating communication, information and computation infrastructure into its operation, control and protection [1]. Accurate load forecasting during both normal and abnormal conditions is essential for reliable grid operation and maintenance [2]. However, load forecasting during pandemic-like situations (PLS), such as COVID-19, can become increasingly difficult due to dramatic changes in electricity demand caused by regulatory orders, the severity of the pandemic and other factors [3]. Different studies have analysed the significant impact of

Abbreviations: CPS, Cyber-physical Systems; PLS, Pandemic-like Situations; SLDC, State Load Dispatch Centre.

COVID-19 on electricity demand, revealing load variations ranging from 13 to 30% [4–6]. The load showed significant variable patterns with a total electricity consumption reduction in both commercial and industrial sectors. Also, the increased attack base makes load forecasting methods vulnerable to cyber incidents [7, 8]. During COVID-19, low demand and uncertain load conditions could camouflage cyberattacks that threaten to compromise power system stability [9]. Anomalous forecasted data can worsen PLS, impact the life span of power system components and result in a negative economic impact [7].

Government orders such as weekend curfew, complete lockdown, severity of the pandemic and public reaction have a high impact on the load profile. Figure 1 illustrates a scenario of the load demand variation with COVID-19 cases and restrictions imposed by the Government of Delhi, India, for the given duration [10] during which a weekend curfew on the movement of individuals was imposed (16 April 2021 to 19 April 2021) to control the increasing COVID-19 cases. These restrictions shut down the nonessential industries and prevented human movement. This further reduced the power consumption in different industrial segments. Further, a complete curfew imposed (on 19 April 2021) extended the complete lockdown's duration, impacting power use similarly. As load demand is uncertain, forecasting the load demand during PLS can be essential in providing reliable supply to hospitals, emergency care and related industries.

During PLS, electricity load demand changes significantly as shown in 1 for Delhi SLDC. This change is not normal and has unseen patterns which can align with various cyberattack patterns. A similar pattern attack can be misunderstood with a normal event during PLS. Although forecasting algorithms have seen significant enhancements recently, a successful coordinated attack remains capable of altering the forecasted load.

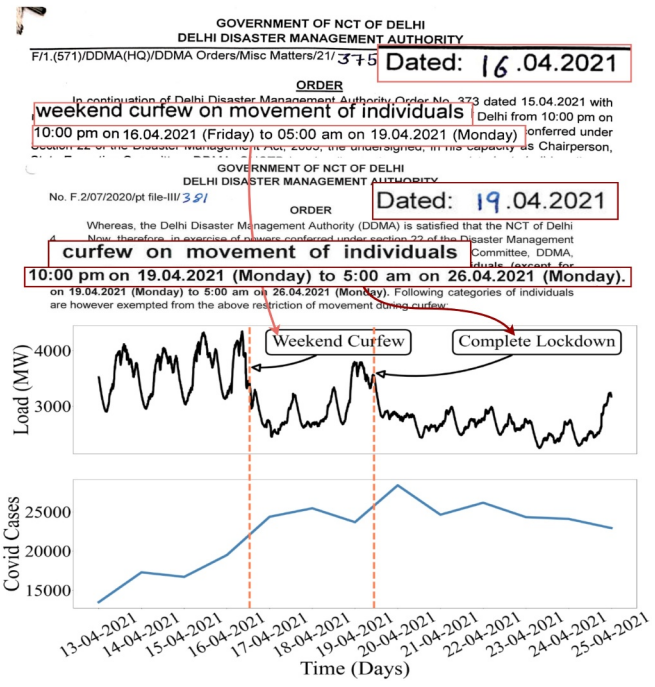

FIGURE 1 | Relation between lockdown orders, COVID cases and actual load demand of SLDC, Delhi.

This paper presents a novel anomaly detection method that utilises function-based, rule-based and tree-based predictors as base models to classify loads as anomalous or normal based on predefined thresholds and further categorises anomalous data into various types of cyberattacks. The proposed load forecasting method incorporates the impact of regulatory orders and pandemic severity, ensuring accurate predictions under dynamic conditions. Additionally, the developed tool offers quick evaluation capabilities and in-depth visualisations of data and results, enhancing its usability and effectiveness in various situations. The main contributions of this paper can be summarised below.

- Proposed a machine learning classifier-based anomaly detection method to distinguish load volatility from cyberattacks during PLS, ensuring reliable detection.
- Developed a rotation forest-based method to classify anomalous loads into six cyberattack types, enhancing diversity and accuracy through subset creation and PCA, effective even for small datasets.
- Developed a short-term energy load forecasting method for PLS that integrates regulatory orders and pandemic severity factors to account for the pandemic's impact, significantly enhancing accuracy.

The rest of the paper is organised as follows. In Section 2, a review of relevant literature is presented. In Section 3, the details of the cyber risk identification and classification-based load forecasting tool for pandemic situations are discussed. Section 4 presents a load forecasting algorithm that considers government orders and pandemic severity. Section 5 describes the cyberattack models, detailed methodology of anomaly detection and cyberattack classification. Case studies and result analysis performed on publicly available load data are discussed in Section 6. Concluding remarks are summarised in Section 7.

2 | Related Works

In this section, we have discussed a brief literature review on load forecasting, anomaly detection and cyberattack classification in the view of PLS. In ref. [6], it is shown that COVID-19 had a significant impact on the energy demand curve, leading to a reduction in consumption and changes in the demand pattern throughout the day. As discussed, load forecasting is important for planning and scheduling power to ensure the reliable operation of power systems. The existing short-term load forecasting models are based on time series analysis (ARIMA, seasonal decomposition, exponential smoothing etc.) [11], regression analysis [12] and machine learning approaches [13–15]. These methods do not account for factors related to the impact of COVID-19-like situations, resulting in inaccurate forecasts that affect power system reliability. Recently explored approaches for load forecasting during pandemics and similar situations use either statistical methods or artificial intelligence-based methods [15–17]. These methods require a large amount of data and have higher computational requirements. Given the short-term nature of pandemics and similar situations, acquiring long-duration data may not be feasible. In ref. [8], an online model for probability density load forecasting is proposed, using a regression LSTM network to capture time

dependencies and learn new concepts in the data. However, this model does not include factors such as government orders, pandemic severity and other external influences.

Attacks on load forecasting from an attacker's perspective can be divided into six types, which include random attack, pulse attack, Type 1 ramping attack, Type 2 ramping attack, scaling attack and smooth curve attack [7, 18, 19]. These attacks can cause the forecast system to over forecast or under forecast which will have negative economic and social impacts. During an under-forecast, the available capacity may not meet the required demands, increasing strain on the energy supply system. As reserves are used to fill the gap during regular operating times, outages to entire geographical areas become even more likely during peak times. While, if over-forecasted, power scheduled to produce this over-forecast can result in the unnecessary start-up of power generation units and excess use of storage devices in the network, resulting in over-frequency events leading to power system instabilities. Load forecasting and detection of cyberattacks in load forecasting during PLS can be challenging due to the regulatory orders, pandemic severity, human reaction patterns, coordinated attack strategy etc [9].

There are various tools available in the market for load forecasting. Sensewaves Adaptix Grid platform [20] is an adaptive machine learning-based forecasting tool which considers individual customer's energy usage data and weather forecasts to calculate separate predictions for residents, commercial market and industrial customers. Flexo Hive Power [21] is a smart grid analytic tool to manage electrical energy and electrical grids using data-driven methods. Amperon AmpGrid [22] enables short-term load forecasting at the level of individual buildings. Nectaware E4SIGHT [23] is a cloud-based platform which makes use of both consumption data from smart grids and predictive algorithms. These tools have limited capability of cyberattack detection during PLS due to the uncertainty of load demand, impact of imposed rules etc.

A short-term load forecasting algorithm with limited previous days' data is proposed in ref. [16]. However, the impact of government orders and pandemic severity was not considered. Arjomandi et al. in ref. [24] uses a machine learning ensemble method to forecast the load, considering lockdown temporal policies as a feature. The inclusion of pandemic-related policies has been mentioned. Still, the contribution of each severity due to policies needed to be clarified, and the contribution due to COVID-19 cases, is not included. In ref. [25], an ensemble anomaly detection method outperformed ARIMA in forecasting electricity demand during the COVID-19 pandemic. However, it does not consider factors like pandemic severity and regulatory orders, which are crucial for understanding the pandemic's impact on energy use. Other limitations can be in terms of the duration of load data available as the PLS can be a short-term situation. Also, previous studies [7, 26] focus on anomaly detection, but less attention is given to attack classification. In ref. [27], an ARIMA and autoencoder-based load forecasting method are presented, demonstrating that the autoencoder outperforms ARIMA for load forecasting. The paper also proposes a method to detect cyberattacks in load forecasting, utilising the Gaussian 3-sigma rule as a threshold to determine anomalies in the forecasted load. However, autoencoder-based methods require a large amount of historical data, which pose a challenge for short-term events such as COVID-19. Although this paper successfully detects anomalies, it does not identify the specific types of attacks performed on the load data.

Considering the limitations of existing studies, this research performs short-term load forecasting using actual load data from Delhi SLDC during the COVID-19 pandemic, taking into account pandemic severity and regulatory orders. It employs ML classification methods to detect anomalies and predict the type of cyberattack among six categories, as detailed in Section 3.

3 | Overview of the Proposed Cyber Risk Assessment Methodology

As discussed, a *cyber risk identification and classification-based load forecasting method for pandemic situations* can be utilised to detect and classify cyberattack incidents and anomalies in forecasted energy load during normal and PLS. This section provides an overview of the functionalities of the developed tool, as illustrated in Figure 2.

Data analysis involves performing mathematical analysis on the data we are using for cyberattack identification and classification-based load forecasting tools for PLS.

In order to perform data analysis, three different types of analysis are developed in this tool. First, some attributes provide more significant information than others when predicting a scenario. Measurements used in this are known as information gained, and the analysis performed is known as *information gained analysis*. The second data analysis functionality is the *correlation attribute analysis*. This analysis evaluates the worth of an attribute by measuring its Pearson's correlation coefficient between it and the identified class. Last, *data summary analysis* includes the name of each attribute, its type, how many were missing, the uniqueness of the data found within and the distinct values found within the data.

The *load forecasting* feature of the tool allows system operators to forecast the load during COVID-19-like pandemic situations based on previous loads, pandemic situations and government orders. The *attack detection and classification* function enable

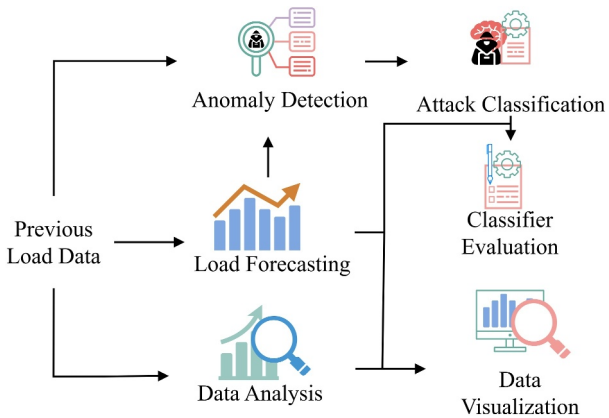

FIGURE 2 | Functionality overview of the cyber risk assessment tool.

system operators to detect the anomalies and cyberattacks in forecasted load and classify the cyberattack types using an attack classifier trained based on the attack classifier feature of the tool. A total of six types of classifiers were identified as successful with the data we used and provided to the tool users. A feature to train the *attack classifier* is provided with different attack classifiers, which can be selected based on user choice. This tool enhances user flexibility by presenting a range of prediction models as a base for precise anomaly detection.

Quick-fold and ten-fold *classifier evaluation* methods are utilised to evaluate the selected classifier. Also, the tool adds the *data visualisation* feature of load input data, attack classification and forecasted load for the operator's convenience. The tool provides a data analysis feature which is essential to understand the data quality, correlation between data and information gained from each data point. The tool is developed using Python and is available as open source on GitHub [28].

4 | Proposed Load Forecasting Approach

As discussed in 3, load forecasting during PLS is critical for a reliable and secure operation of power systems. In this work, a load forecasting model during PLS is developed using the weighted factor for immediate previous loads, considering the availability of limited previous load data due to the short duration of the situation and COVID-19-related orders from the regulatory authorities. Equation (1) gives the forecasted load value of day of interest d at time block t $\left(Y_t^d\right)$.

$$Y_t^d = \begin{cases} Y_t'^d\left(1 - \dfrac{\beta}{b^{\left(\alpha_{t,\max}^d - \alpha_t^d\right)}}\right), & \text{if } \alpha_t^d > 0 \\ Y_t'^d\left(1 + \dfrac{\beta}{b^{\left(\alpha_{t,\max}^d + \alpha_t^d\right)}}\right), & \text{if } \alpha_t^d < 0 \\ Y_t'^d, & \text{Otherwise} \end{cases} \tag{1}$$

In Equation (1), α_t^d, $\alpha_{t,\max}^d$, b and β are the model parameters. A country may have different states which are governed by different governments. During emergencies, different actions are taken by the governments of different states, which leads to different form of lockdowns with varying infections spread across state borders. α_t^d is the relative severity factor assigned to each time interval within a day. This score is derived by analysing both government-issued COVID-19 prevention orders and guidelines, along with historical patterns of COVID-19 cases. $\alpha_{t,\max}^d$ is the severity factor corresponding to maximum severity, that is, complete lockdown. b is the exponential base which provides the relation between different lockdowns. We can determine the relative impact on load demand changes by analysing previous orders and their relationships. β is the max load deviation due to complete lockdown in the given geometry.

$Y_t'^d$ is the contribution of the previous days load on the forecasted load. $Y_t'^d$ is determined from Equation (2) which is also presented in ref. [16].

$$Y_t'^d = \widetilde{\eta^{\eta-1}} L_t^{d-1} + \widetilde{\eta^{\eta-2}} L_t^{d-2} + \cdots + \widetilde{\eta^{\eta-\eta}} L_t^{d-\eta} \tag{2}$$

$$\widetilde{\eta^{\eta-i}} = \frac{\eta^{\eta-i}}{\sum_{j=1}^{\eta} \eta^{\eta-j}} \tag{3}$$

$L_t^{d-1}, L_t^{d-2} \ldots L_t^{d-\eta}$ are the load values of the immediate previous η days. Where η is the number of days contributing to load forecasting. $\widetilde{\eta^{\eta-2}}, \widetilde{\eta^{\eta-3}}, \ldots, \widetilde{\eta^{\eta-\eta}}$ are normalised weights such that

$$\sum_{i=1}^{\eta} \widetilde{\eta^{\eta-i}} = 1 \tag{4}$$

Weights $\widetilde{\eta^{\eta-i}}$ decrease exponentially with the highest weight to immediate previous day load, that is, L_t^{d-1} as represented in Figure 3.

The optimal η value for the load forecasting is selected using the averaged mean absolute percentage error (MAPE), where the averaged MAPE is the average of MAPE of given days and MAPE of each day is the average of MAPE for training intervals in that day. The η with least average MAPE is considered as the optimal η. The weight contributions from previous days, shown in Figure 3, are modelled by Equations (2–4), where the weights $\left(\widetilde{\eta}^{\eta-i}\right)$ decrease exponentially, giving higher importance to more recent loads. This ensures the model effectively captures temporal dependencies, making it robust to abrupt load pattern changes during emergencies. If any previous day is missing while determining the optimal η, the number of days can be extended to ensure the total number of days matches η.

The forecasted load $\left(Y_t^d\right)$ is determined using the previous loads $\left(Y_t'^d\right)$ and adjusts exponentially based on the severity factor α_t^d. Figure 4 illustrates the impact of the relative severity factor $\left(\alpha_t^d\right)$ on the forecasted load $\left(Y_t^d\right)$ by capturing the relative difference in pandemic-related restrictions and COVID-19 cases between the current and previous days with different exponential bases ($b = 1.5, 2.0, e, 10$). The impact of the severity factor α_t^d can be summarised as follows:

1. *Negative relative severity* $\left(\alpha_t^d > 0\right)$: When restrictions are eased, the forecasted load increases relative to the previous day. This increase is more pronounced for smaller values of the base b, indicating higher sensitivity to changes in severity.
2. *Positive relative severity* $\left(\alpha_t^d < 0\right)$: When restrictions are tightened, the forecasted load decreases relative to the

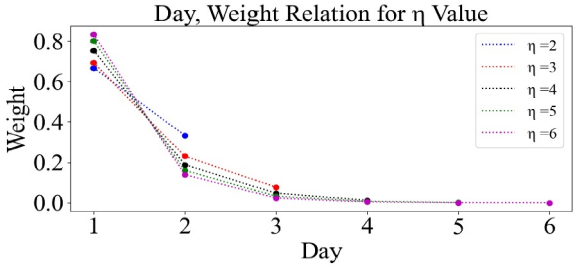

FIGURE 3 | Day and weight relation for the $\eta = 2, 3, 4, 5, 6$.

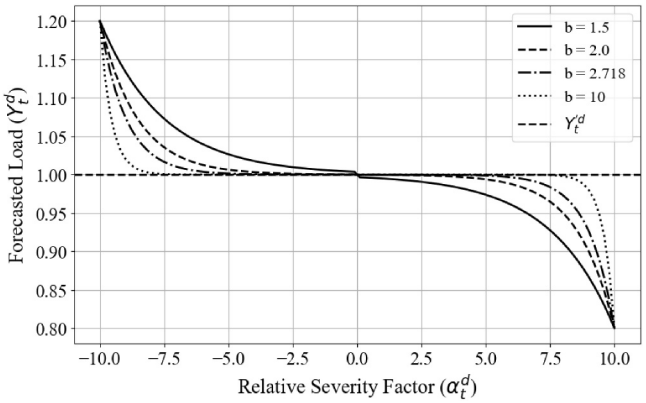

FIGURE 4 | Impact of the severity factor (α_t^d) on forecasted load.

previous day. The rate of decrease is more significant for smaller bases, reflecting sharper load responses.

3. *Neutral severity* $(\alpha_t^d \approx 0)$: In this scenario, the forecasted load remains nearly constant, matching the contribution from the immediate previous day's load, as described in Equation (2).

The proposed method emphasises the exponential relationship between the severity factor α_t^d and load deviations. By incorporating historical load data and regulatory information, the model provides accurate and dynamic load forecasting, which is critical for a reliable power system operation during disruptive events.

A detailed proposed algorithm is presented in Algorithm 1. This method depends on the lockdown orders, COVID-19 cases and load demand of previous days, which makes it accurate. Also, it is easy to decide the model parameter by analysing the regulatory orders and information from other trusted broadcast sources.

5 | Proposed Cyberattack Detection and Classification Approach

As discussed in 1, load forecasting during COVID-19 PLS is crucial and may be vulnerable to cyberattacks due to a large attack base. This section presents a detailed explanation of cyberattack models, anomaly detection and attack classification methodology.

5.1 | Attack Models

Accurate modelling of cyberattacks is essential for the effective detection of data integrity attacks in load forecasting. All six types of cyberattacks on load forecasting based on the attacker's perspective as discussed in Section 1 are modelled as follows:

5.1.1 | Random Attacks

Random attacks involve adding positive values from a uniform random function to the initially forecasted loads. These attacks are challenging to detect due to their unpredictable nature,

ALGORITHM 1 | Algorithm for load forecasting considering government orders and pandemic severity.

Input: Previous load data for η duration, α_t^d for each load of η duration
Output: Forecasted load value for days of interest
1: *Initialisation*:
2: Determine optimal η as based on averaged MAPE
3: Determine normalised weights as per (3) for optimal η
4: *Forecast load*
5: **for** $t = 0$ to $t_{\max}$ **do**
6: Calculate $Y_t^{'d}$ for time block t using (2)
7: Calculate Y_t^d for time block t using (1)
8: **end for**
9: **return** Y_t^d for $0 \le t \le t_{\max}$

especially during lockdown periods when energy consumption levels are already inconsistent.

$$Af(t) = Of(t) + SF * rand(t) \tag{5}$$

Where, for t_s (start time of attack) $< t < t_e$ (end time of attack), $rand()$ is a uniformly distributed random number generator, $Af(t)$ is the altered forecast, $Of(t)$ is the original forecast and SF is a scale factor defined as half of the maximum load forecast value, that is, $SF = \max(Of(t))/2$.

5.1.2 | Pulse Attacks

A pulse attack involves manipulating the load forecasting data to higher or lower values at specific points throughout an attack. These attacks are often mistaken as normal loads with anomalies during lockdown periods due to the unpredictability of energy requirements.

$$Af(t) = (1 + PulseAttackParameter) * Of(t) \tag{6}$$

For $t = t_P$, where t_P is the occurrence time of a single pulse attack. During a lockdown situation, pulse attack may remain undetectable because of the uncertainty of the load irrespective of previous loads.

5.1.3 | Type 1 Ramping Attacks

Type 1 ramping attack involves only the ramping up of values in a specific range by a ramping function. These attacks are particularly effective during transition periods of lockdowns, exploiting rapid changes in energy demand.

$$Af(t) = RF * (t - t_s) * Of(t); t_s < t < t_e \tag{7}$$

When this time increases, the value of $t - t_s$ ensuring that an up-ramping anomaly is performed.

5.1.4 | Type 2 Ramping Attacks

Type 2 ramping attacks involve both ramping up and ramping down of the forecasted loads, making them harder to detect due to their similarity to natural load variations during volatile periods.

$$Af(t) = [1 + RF * (t - t_s)] * Of(t); t_s < t < (t_s + t_e)/2 \quad (8)$$

$$Af(t) = [1 + RF * (t_e - t)] * Of(t); (t_s + t_e)/2 < t < t_e \quad (9)$$

Equations (8) and (9) are up and down ramping anomalies for Type 2 ramping attacks, respectively. Ramping attacks particularly during lockdown's initial and end stages are difficult to detect.

5.1.5 | Scaling Attack

A scaling attack involves modifying the values produced during a specific time frame. These sets of values are multiplied by the scaling attack parameter.

$$Af(t) = (1 + ScaleAttackParameter) * Of(t); t_s < t < t_e \quad (10)$$

Scaling attacks are more effective during low consumption periods, resembling the pandemic's recovery conditions.

5.1.6 | Smooth Curve Attack

A smooth curve attack is produced by replacing a set of continuous start and endpoints within the original load forecasting data. These attacks are less detectable initially but cause significant deviations as time progresses.

As discussed, each attack can have one of two effects on the system. They can either cause the forecast system to over-forecast or under-forecast. Both over and under forecast can cause a negative impact on the grid in terms of stability, load loss or equipment failure or life-span reduction. A Python programme was developed to simulate all six attack events on real-load data. This simulated event data were then used to train a nominal classifier to distinguish between all seven possible events.

5.2 | Anomaly Detection and Attack Classification Method

A numeric machine learning model is employed to identify various anomalies in load forecasts by training on historical load data. Training a numerical machine learning model on past load data can predict the expected load. If the disparity between the forecasted load and the predicted load exceeds a predefined threshold, the forecasted load will be categorised as anomalous. After identifying an anomalous load, an attack classifier is employed to categorise the cyberattack present within the load. The attack classifier is trained on simulated attack and historical load data to classify the anomalous forecasted load. Figure 5 represents the anomaly detection and attack classification method based on historical load data.

5.2.1 | Load Anomaly Detection

In the proposed attack detection method, a load prediction classifier utilises a numeric machine learning classification algorithm trained on previous load data to detect altered load. This load data would have been used to produce the forecast so the classifier will understand where each forecasted load should be in relation to each other. This means that when there is a sudden increase or decrease in the forecasted load, a classifier will identify this as anomalous and not identify regular changes in load forecasting as anomalous data. Load prediction models for anomaly detection are specified in Table 1. This load prediction classifier uses changing the classifiers class attribute to be the same as the load currently being tested. This takes into context the rest of the data points within the instance (day) due to the volatility experienced during a lockdown. Previous load data produce the forecast (P^{Pred}) by the selected load prediction model. It helps the load prediction classifier to understand where each forecasted load should be in relation to each other. When there is a sudden increase or decrease in the forecasted

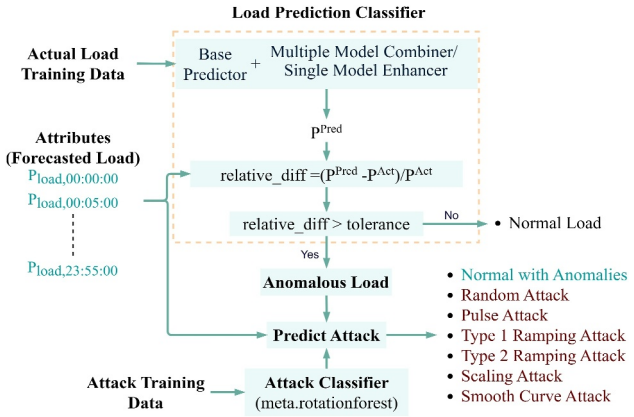

FIGURE 5 | A representation of the proposed method with attack type classification and anomaly detection.

TABLE 1 | Load prediction models details.

Prediction model group	Model name	Description
Function	PLSFilter and classifier [29]	Partial least square method
	SLR [30]	Simple linear regression method
	SMOreg [14]	Support vector machine for regression
Rule	M5Rules [31]	Builds a model tree using M5 and makes the 'best' leaf into a rule
Tree	M5P [32]	Tree-based multivariate linear model

load, the load prediction classifier will identify this as anomalous, not affecting regular changes. The tolerance Equation (11) is represented as a fraction of the actual load (P^{Act}) in the given time period to classify the load as anomalous. The details of each anomaly detection classifier are summarised below.

$$tolerance = \alpha P^{Act}; 0 < \alpha < 1 \tag{11}$$

The load is identified as anomalous if the difference between the predicted (P^{Pred}) and actual (P^{Act}) loads exceeds relative tolerance.

5.2.1.1 | PLSFilter and Classifier. Partial least squares (PLS) regression is a dimensionality reduction technique that constructs latent variables to maximise the covariance between the predictor (P^{Act}) and target (P^{Pred}) variables. The PLSFilter and classifier directly calculate PLS components as linear combinations of the original attributes, bypassing the need for iterative data deflation as in traditional algorithms like NIPALS [29]. For P^{Pred}, the PLSFilter identifies the most relevant components from P^{Act} by maximising their covariance while maintaining orthogonality and normalisation. Each latent variable is a weighted sum of original attributes, ensuring the model remains robust against multicollinearity. The classifier uses these components to classify data points and detect anomalies in load forecasting, particularly under pandemic-like scenarios.

5.2.1.2 | Simple Linear Regression. Simple linear regression (SLR) is a numeric prediction technique that models the relationship between the predicted load (P^{Pred}) and previous load attributes within an instance (e.g., a day) [30]. SLR establishes this relationship by representing the dependent variable (P^{Pred}) as a linear combination of attribute values $(x_1, x_2, ..., x_k)$ from the load data, weighted by coefficients $(w_1, w_2, ..., w_k)$ and expressed mathematically as follows:

$$P^{Pred} = w_0 + w_1x_1 + w_2x_2 + \cdots + w_kx_k, \tag{12}$$

where w_0 is the intercept. The model's coefficients (w_j) are optimised by minimising the sum of squared residuals between the actual load (P^{Act}) and the predicted load (P^{Pred}):

$$\min \sum_{i=1}^{n} \left(P_i^{Act} - \sum_{j=0}^{k} w_j x_j^{(i)} \right)^2. \tag{13}$$

SLR is particularly suited for identifying load anomalies during pandemic-like situations (PLS), where sudden shifts in demand caused by factors such as regulatory actions or human behaviour changes can significantly affect load patterns. By being trained on previous load data, SLR captures the inherent relationships between load attributes, allowing it to contextualise forecasted values within the broader temporal structure of the data (e.g., daily or hourly patterns).

5.2.1.3 | SMOreg. SMOreg uses support vector regression (SVR) to model the relationship between actual load (P^{Act}) and predicted load (P^{Pred}) for anomaly detection [14]. The regression function is defined as follows:

$$P^{Pred} = b + \sum_{i \in \text{support vectors}} \alpha_i P_i^{Act} K(x, x_i), \tag{14}$$

where b is the bias term, α_i are Lagrange multipliers and $K(x, x_i)$ is the kernel function.

An ϵ-insensitive loss function is employed to ignore deviations within a margin of tolerance (ϵ), focusing only on significant anomalies. The optimisation problem minimises the model complexity while penalising deviations beyond ϵ:

$$|P^{Act} - P^{Pred}| > \alpha P^{Act}, \tag{15}$$

where α is the tolerance parameter.

SMOreg efficiently handles large datasets by iteratively optimising pairs of Lagrange multipliers, ensuring computational scalability. This makes it well-suited for detecting sudden spikes or drops in load while filtering out regular fluctuations, particularly in volatile conditions like lockdowns.

5.2.1.4 | M5Rules. M5Rules generates regression rules from model trees to predict P^{Pred} based on P^{Act} and other attributes of the instance. It constructs a model tree with linear regression models at its leaf nodes and extracts rules from the tree. Each rule is derived from a tree branch, with a linear model representing the relationship between P^{Pred} and its prediction attributes [31]. For anomaly detection, M5Rules identifies deviations between P^{Pred} and P^{Act} beyond a predefined tolerance, leveraging its ability to model nonlinear interactions while maintaining interpretability. The rules are pruned to ensure generalisation, and smoothing is applied to reduce discontinuities between adjacent models, enhancing robustness in detecting anomalous patterns in load forecasting.

5.2.1.5 | M5P. M5P is a model tree algorithm that integrates decision tree construction with multivariate linear regression at the leaf nodes, enabling piecewise linear predictions for P^{Pred} using P^{Act} and related attributes [32]. It recursively splits data to minimise variance, with variance reduction calculated as follows:

$$\Delta\text{Error} = \text{sd}(T) - \sum_i \frac{|T_i|}{|T|} \text{sd}(T_i),$$

where T represents the dataset at the current node, T_i are the subsets formed by the split, $|T|$ and $|T_i|$ are the number of instances in T and T_i, respectively, and $\text{sd}(T)$ is the standard deviation of the target variable P^{Act} in T. Linear models at leaf nodes predict P^{Pred} as in Equation (12). These linear models use only attributes relevant to the subtree, ensuring computational efficiency. Pruning replaces subtrees with linear models when they improve accuracy, and smoothing adjusts predictions along the path from the root to leaf, reducing discontinuities and enhancing robustness. Unlike M5Rules, which extracts regression rules from branches for simplicity, M5P retains the full tree structure, offering global consistency and superior handling of complex and dynamic datasets. In anomaly detection for load forecasting, M5P effectively identifies deviations between P^{Pred}

and P^{Act} when these exceed the predefined tolerance, making it a robust tool for identifying irregularities in volatile conditions.

5.2.2 | Prediction Model Enhancers

Single or multiple model enhancers can be applied to the selected regression models to improve load prediction for anomaly detection. Three types of model enhancers are considered:

5.2.2.1 | Bagging. Bagging establishes a regressor/classifier for each uniquely created multiple samples based on the base learner [33]. A bootstrap aggregate predictor combines the prediction of each base learner to produce overall predictions. This single-learner enhancer also significantly increases the creation time of the prediction model. It can improve accuracy if uniquely created learning sets can cause the changes in the predictor.

5.2.2.2 | Boosting. Ada Boost EM1 is a statistical classification meta-algorithm that gives a weighted sum of weak learner algorithms output to an overall boosted predictor. This algorithm enhances the accuracy, albeit at the cost of increased time requirements.

5.2.2.3 | Vote. The vote is a single predictor enhancer that allows the combination of predictors to produce an overall predictor that is, in theory, more accurate than an individual predictor. This can be applied to one predictor by merging the adapted boosting predictor and the bootstrap aggregating predictor. Subsequently, this single predictor enhancer can also be applied to all the individual predictor algorithms, making it a multiple predictor enhancer.

These predictors and predictor enhancers are incorporated into the tool using a Python wrapper of WEKA API [34].

5.2.3 | Attack Classification

The load anomaly detection method as detailed in Section 5.2.1 determines whether the load is classified as anomalous or not depending on its adherence to a specified tolerance threshold. When a group of anomalous loads is detected, the challenge lies in accurately determining the specific type of attack within the load forecasting data. This includes categorising the attack among the seven possibilities of no event or attack events as defined in Section 5.1 from the perspective of the attacker. A rotation forest [35] classifier is employed to classify the cyberattack on forecasted loads marked as anomalous by the initial classifier. This classifier is called the cyberattack classifier as shown in Figure 6. This classifier is trained using N days historical load and attack data generated using attack models specified in 5.1. These data are used to train the attack classifier where load data (X_t^i) of each time interval t of given day i are considered the features/input for the classifier. The attack class is determined by the numeric value predicted by the attack classifier.

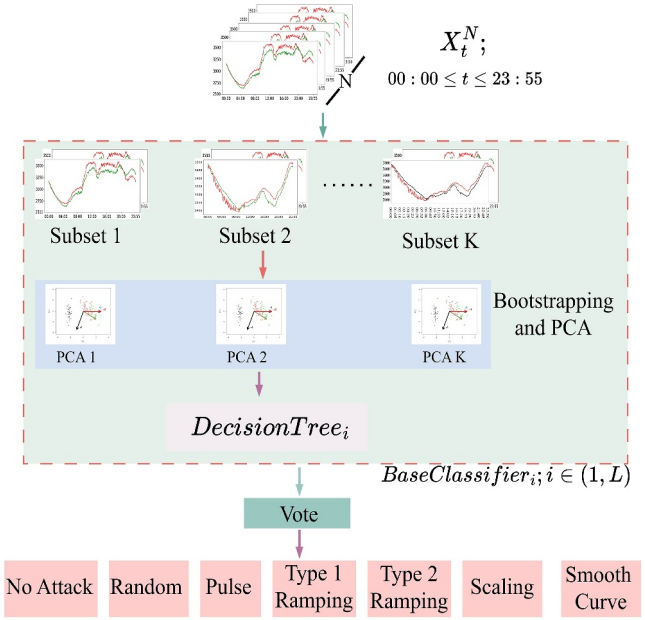

FIGURE 6 | Attack classification model using rotation forest.

5.2.4 | Evaluation Methods

This section discusses the evaluation methods of the classifiers considered for cyberattack classification. This tool uses WEKA library-based evaluation models to evaluate the classifiers.

5.2.4.1 | K Fold Analysis. In k-fold cross-validation, the dataset is split into k number of subsets, and then training is performed on all the subsets, but one (k-1) subset is used to evaluate the trained model [36].

5.2.4.2 | Quick Analysis. The default evaluation method for the *weka.classifier.evaluation* class in WEKA is typically '10-fold cross-validation'. In this default evaluation method, the dataset is divided into 10 subsets (folds), and the classifier is trained and tested on these subsets in a rotating fashion.

These analysis methods reduce overfitting, improve generalisation to unseen data and ensure fair performance assessment, even with limited or imbalanced datasets. Specifically, 10-fold cross-validation, as the WEKA default, balances computational efficiency and reliability, enabling consistent evaluation across varied data splits.

6 | Case Study and Results

India is a largely populated country, so the impact of COVID-19-like situations can be huge. The load demand of Delhi (the capital city of India) has been considered for the analysis purpose. The Delhi State Load Dispatch Centre (SLDC) provides the load demand data for 5 min for all distribution companies in their territory [10]. All the cases analysed use the data available at the Delhi SLDC during the first and second waves of the COVID-19 pandemic. Cyberattack incidents are generated by imposing attacks specified in 5.1 on Delhi SLDC load data.

6.1 | Load Forecasting

As shown in Figure 3, the model first determines the optimal η, which is then used for load forecasting for the given time period. The model parameters are determined based on Delhi Government Orders and pandemic cases and are described below.

- The value of α_t^d has a maximum value of 10, which corresponds to maximum severity, that is, complete lockdown and has lower values when severity decreases. As per load data from Delhi SLDC, we have a range of α_t^d in between 1 and 10.
- By observing the load demand, it is determined that the impact of each relaxed level on load demand decreases by a factor of 2, so the value of b is considered as 2.
- Additionally the maximum impact on the load due to the complete lockdown is considered to have a 0.2 times decrease from the nominal value.

The determination of the severity factor is done based on the government orders and COVID-19 related cases as shown in Figure 7.

Complete lockdown and increasing COVID-19 cases can be assigned with the highest severity. No restrictions imposed by the government and very low COVID-19 cases can define zero severity, which means the pandemic's impact is minimal. The considered cases include single-day and multi-day forecasts, as described below.

6.1.1 | One Day Forecast

The initial η range can be considered based on the available previous load data where more than 1 week of data is sufficient. Here, the considered η range is between 2 and 9. As discussed in Section 4, the optimal η value is determined based on the average MAPE. Load data for April 2–10 April 2021 have been considered to forecast the load for April 11, 2021. The optimal η obtained from the model is 9. Based on this, Figure 8 shows the results of the forecasted and actual load demand of April 11, 2021, which shows high accuracy for the forecasted load.

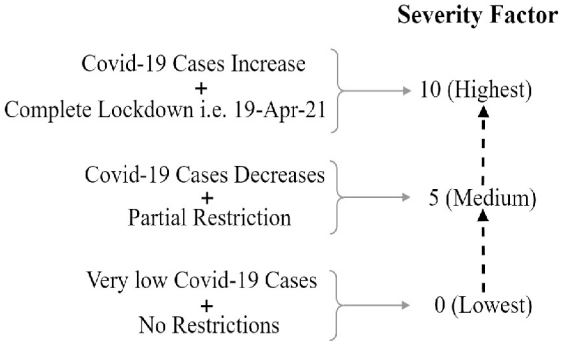

FIGURE 7 | Determination of the COVID-19 severity factor with government orders and COVID-19 confirmed cases.

6.1.2 | Multi-Day Forecast

For the multi-day forecast, optimal η is determined using the previous load data present and remains unchanged for the duration of the forecast. In this case, the load forecast duration is May 14–May 17, 2021. The optimal η determined for this case is 3. The forecasted load is shown in Figure 8. The average MAPE obtained for this case is 4.3%. The proposed method in this paper achieves higher accuracy with a low-averaged MAPE (mean absolute percentage error) for both single-day and multi-day forecasts.

6.1.3 | One Day Forecast With Missing Data

Load data for April 2–10, 2021, are considered to forecast the load for April 11, 2021. However, load data for April 7–9 are missing. The model compensates for the missing data by utilising the available load data from earlier days during forecasting. Figure 9 shows the load profiles of the actual load, forecasted load without missing data and forecasted load with missing data for April 11, 2021. The MAPE for the actual load compared to the forecasted load without the missing data is 2.81%, whereas the MAPE for the actual load compared to the forecasted load with the missing data increases to 3.19%. This demonstrates a slight reduction in accuracy due to the missing

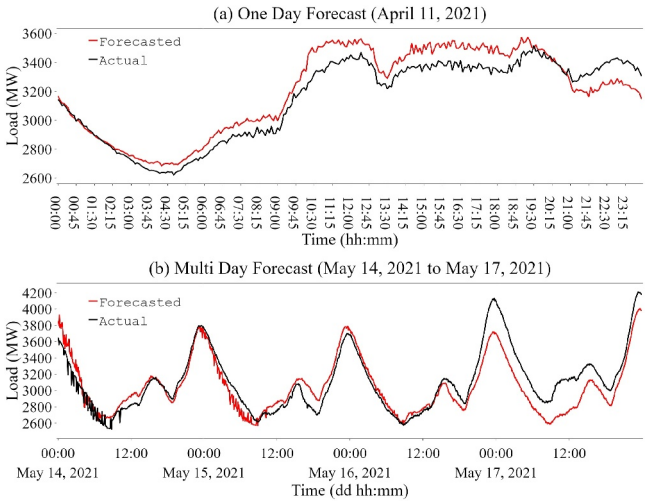

FIGURE 8 | Plots of actual and forecasted load for (a) one day (April 14, 2021) (b) multi-day (May 14–May 17, 2021).

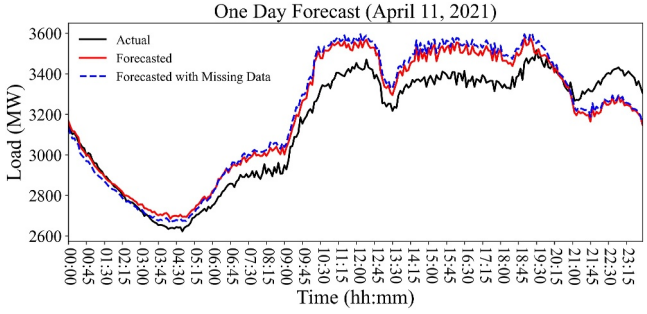

FIGURE 9 | Plot of actual and forecasted load for one day (April 14, 2021) with and without missing data.

data, emphasising the importance of continuous data availability for reliable load forecasting.

6.2 | Cyberattack Detection and Classification

Due to the unavailability of real attack data from Delhi SLDC, attack scenarios were simulated using forecasted load data from the SLDC and the attack models defined in Section 5.1. This simulated attack data were then used to perform anomaly detection and attack classification. A total of 511 cases were developed, including all six types of attack models as shown in Table 2.

6.2.1 | Anomaly Detection

The attack and forecasted data are inputted to the anomaly classifier. This classifier will predict the load and compare it with the input forecasted load. As defined in Section 5.2.1, a total of five types of classifiers are used to determine whether the forecasted load is anomalous or not. To evaluate the models, k-fold cross-validation is performed for all the classifiers. In k-fold cross-validation, the dataset is split into k subsets; training is performed on $k - 1$ subsets, and the remaining subset is used to evaluate the trained model [36]. A '10-fold cross-validation' is used for a quick evaluation of classifier performance. A relative performance comparison for all classifiers is shown in Table 3. SMOreg performed relatively better, detecting 241 anomalies out of a total of 287 tested instances, with a maximum relative difference of 0.4322 and an average difference of 478.46.

6.2.2 | Attack Classification

After detecting anomalies in the load, an attack classification was performed. In the initial testing of the meta-type classifier, the rotation forest achieved a 92.37% correct classification rate without any single-classifier enhancers compared to other classifiers. A confusion matrix is presented in Figure 10 for meta-type classifier rotation forest for specified data types and classes.

The rotation forest confusion matrix shows high levels of correct classification amongst all event types but the worst performance for the pulse attack. A small minority of the pulse attack instances are classified as no events due to the nature of the pulse attack only affecting the forecasted load for a single instance, making it difficult to detect with these algorithms. The best attack types classified were the ramping and smooth curve attacks. These three attack types were correctly classified as 71 out of 73 times. This is due to the significant increase in the difference between the forecasted load and the affected load by the end of the day, thus making it relatively apparent when these types of attacks have occurred.

6.2.3 | Data Analysis

Individual attributes used in training data affect the model's prediction ability. Some attributes provide a more significant amount of information predicting the scenario than others. This measurement is known as information gained. In this tool, the WEKA code library produces a ranked list of attributes based on how much information they provide to a classifier. In Figure 11, using information gained analysis, it can be seen that each of the different time attributes is useful in predicting the forecasted load for 12:00 o'clock. Two distinct groups have formed within the graph. Most of the time attributes belong to the first group, providing a Pearson's correlation coefficient of over 6.15. The attributes day, month and year are also shown in Figure 11, where the day attribute is the best performing out of the three, still managing to score a respectable 4.90, but apart from this exception, both month and year perform very poorly, with month scoring only 1.48 and year scoring 0. This is because the PLS are short-term events, and the impact of temporal attributes diminishes with increasing granularity. Specifically, the day attribute exhibits some influence on the attack classification, whereas the month and year attributes have progressively decreasing impacts.

The second data analysis performed is the correlation attribute evaluation. This evaluates the worth of an attribute by measuring its Pearson's correlation coefficient using Equation (16) between it and the identified class in the given instance, whether the instance is malicious or not, where each

TABLE 2 | Training data distribution for all attack types and normal forecasted load.

No event	Pulse	Random	Smooth curve	Scaling	Type 1 ramping	Type 2 ramping
73	73	73	73	73	73	73

TABLE 3 | An analysis of different classifiers in anomaly detection in the forecasted load.

Prediction model	Anomalous	Anomalies detected	Total tested	Maximum relative difference	Average difference
Linear regression method	Yes	243	287	0.5262	559.19
PLSFilter and classifier	Yes	240	287	0.4787	490.24
SMOreg	Yes	241	287	0.4322	478.46
M5Rules	Yes	241	287	0.740	516.45
M5P	Yes	241	287	0.740	514.14

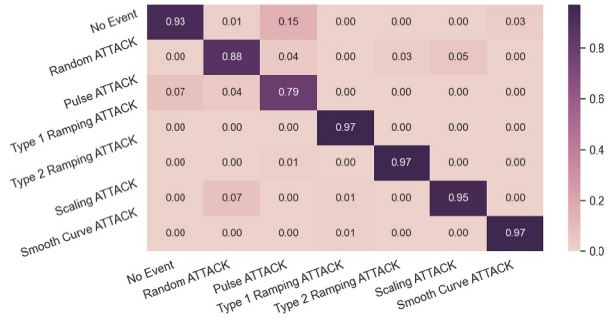

FIGURE 10 | Confusion matrix.

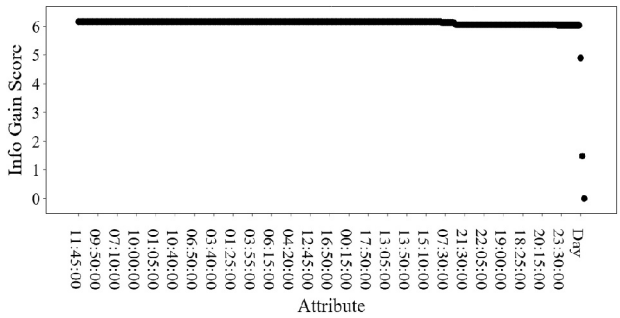

FIGURE 11 | Information gain analysis.

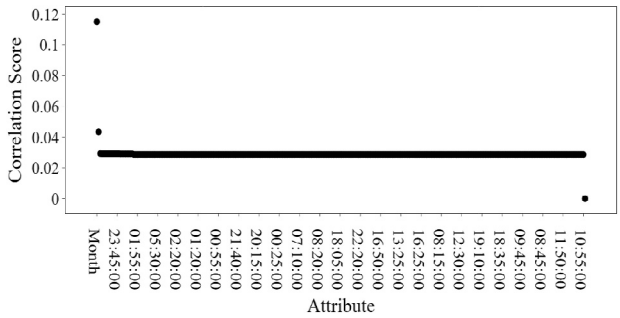

FIGURE 12 | Correlation analysis.

instance in our data is defined as load data for the day of interest d at time block t, day, month and year.

$$\rho_{X,Y} = \frac{\sum_{t=1}^{T}(x_t - \bar{x})(y_t - \bar{y})}{\sqrt{\sum_{t=1}^{T}(x_t - \bar{x})^2}\sqrt{\sum_{t=1}^{T}(y_t - \bar{y})^2}} \tag{16}$$

In Figure 12, different attributes are ranked based on correlation analysis. The vast majority of the attributes performed poorly, with only a few surrounding the targeted load achieving higher than the rest correlation, although this increase is only 0.0005. Month and day had a higher Pearson's correlation coefficient than the best-performing time attribute. Month attribute managed to score 0.115, and day managed to score 0.043, with the highest time attribute coming in at 0.0294. Year attribute is still the worst-performing attribute, scoring 0 for this analysis. Different days of week affect the forecasted load as more energy might be required. Month attribute is affected by weather conditions, which can introduce difficulty in classification.

6.2.4 | Applications

The proposed tool provides various functionalities for load forecasting during emergency energy situations, as well as anomaly detection and cyberattack classification. It can assist system operators in predicting load patterns during emergencies, such as the COVID-19 pandemic, and ensure reliable operations. By detecting anomalies and classifying attacks, the tool can enable operators to correct load forecasts and prevent using false data in system operations. Additionally, it can help analyse the impact on the grid caused by the severity of PLS, such as COVID-19, along with government-imposed regulations. Load forecasting can be crucial for optimising power usage, particularly in scenarios involving human-in-the-loop systems, contributing to the societal and economic benefits of a country. The tool can also benefit utilities and service providers by maximising their contributions during such situations while maintaining optimal costs and ensuring system reliability.

7 | Conclusion

This paper introduces a novel energy load forecasting tool for the detection and classification of cyberattacks in the context of pandemic scenarios. This tool takes into account regulatory orders and incorporates COVID-19 information for short-term load forecasting, ensuring its relevance during uncertain conditions. The proposed load forecasting approach is computationally efficient and effectively utilises limited historical load data, making it suitable for rapidly changing scenarios like pandemics. The inclusion of regulatory orders has a positive impact on the accuracy of load predictions. The anomaly detection and cyberattack classification are enabled through multiple base prediction models, with the developed anomaly detection method effectively handling the volatility commonly observed during lockdowns. By incorporating the context of lockdowns, the method accurately classifies regular pandemic-induced load patterns, distinguishing them from anomalous or malicious data. Notably, rotation forest demonstrates robust performance across various types of attacks. In future, the load forecasting model can be enhanced by incorporating more precise values of forecasting parameters derived using machine learning-based methods.

Author Contributions

Kuldeep Singh Shivran: conceptualisation, methodology, formal analysis, software, visualisation, writing – original draft. **Kyle Swire-Thompson:** conceptualisation, methodology, formal analysis, software, visualisation. **Neetesh Saxena:** conceptualisation, supervision, funding acquisition, formal analysis, project administration, writing – review and editing. **Sarasij Das:** conceptualisation, supervision, funding acquisition, formal analysis, writing – review and editing.

Acknowledgments

This work was supported by the Engineering & Physical Sciences Research Council (EPSRC) - Impact Acceleration Accounts (IAAs).

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data that support the findings of this study are openly available in GitHub repository at https://github.com/CyCISlab/Anomalous-Load-Forecast-Identifier-in-Pandemic [28]. This repository includes all datasets and scripts used for load forecasting, anomaly detection and attack classification as described in the study.

References

1. G. N. Ericsson, "Cyber Security and Power System Communication—Essential Parts of a Smart Grid Infrastructure," *IEEE Transactions on Power Delivery* 25, no. 3 (2010): 1501–1507, https://doi.org/10.1109/TPWRD.2010.2046654.

2. N. Amjady, "Short-Term Bus Load Forecasting of Power Systems by a New Hybrid Method," *IEEE Transactions on Power Systems* 22, no. 1 (2007): 333–341, https://doi.org/10.1109/tpwrs.2006.889130.

3. D. Agdas and P. Barooah, "Impact of the Covid-19 Pandemic on the Us Electricity Demand and Supply: An Early View From Data," *IEEE Access* 8 (2020): 151523–151534, https://doi.org/10.1109/access.2020.3016912.

4. Z. Li, H. Ye, N. Liao, R. Wang, Y. Qiu, and Y. Wang, "Impact of Covid-19 on Electricity Energy Consumption: A Quantitative Analysis on Electricity," *International Journal of Electrical Power & Energy Systems* 140, no. 108 (2022): 084, https://doi.org/10.1016/j.ijepes.2022.108084.

5. J. V. Andrade, R. S. Salles, M. N. Silva, and B. D. Bonatto, "Falling Consumption and Demand for Electricity in South Africa-a Blessing and a Curse," in *2020 IEEE PES/IAS PowerAfrica* (IEEE, 2020), 1–5.

6. A. Navon, R. Machlev, D. Carmon, A. E. Onile, J. Belikov, and Y. Levron, "Effects of the Covid-19 Pandemic on Energy Systems and Electric Power Grids—A Review of the Challenges Ahead," *Energies* 14, no. 4 (2021): 1056, https://doi.org/10.3390/en14041056.

7. M. Cui, J. Wang, and M. Yue, "Machine Learning-Based Anomaly Detection for Load Forecasting Under Cyberattacks," *IEEE Transactions on Smart Grid* 10, no. 5 (2019): 5724–5734, https://doi.org/10.1109/tsg.2018.2890809.

8. C. Cao and Y. He, "An Online Probability Density Load Forecasting Against Concept Drift Under Anomalous Events," *IEEE Transactions on Industrial Informatics* 20, no. 4 (2024): 5241–5252, https://doi.org/10.1109/TII.2023.3331076.

9. S. Lakshminarayana, J. Ospina, and C. Konstantinou, "Load-Altering Attacks Against Power Grids Under Covid-19 Low-Inertia Conditions," *IEEE Open Access Journal of Power and Energy* 9 (2022): 226–240, https://doi.org/10.1109/oajpe.2022.3155973.

10. State Load Despatch Center, Delhi, (2021), https://delhisldc.org/HomeSldc.aspx.

11. G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control* (John Wiley & Sons, 2015).

12. A. Lahouar and J. B. H. Slama, "Day-Ahead Load Forecast Using Random Forest and Expert Input Selection," *Energy Conversion and Management* 103 (2015): 1040–1051, https://doi.org/10.1016/j.enconman.2015.07.041.

13. M. Barman and N. B. D. Choudhury, "Season Specific Approach for Short-Term Load Forecasting Based on Hybrid Fa-Svm and Similarity Concept," *Energy* 174 (2019): 886–896, https://doi.org/10.1016/j.energy.2019.03.010.

14. S. K. Shevade, S. S. Keerthi, C. Bhattacharyya, and K. R. K. Murthy, "Improvements to the Smo Algorithm for Svm Regression," *IEEE Transactions on Neural Networks* 11, no. 5 (2000): 1188–1193, https://doi.org/10.1109/72.870050.

15. M. Ghalib, Z. Bouida, and M. Ibnkahla, "Pandemic-Aware Electric Load Forecasting: A Multitask Bidirectional Lstm/cnn Model," in *ICC 2022-IEEE International Conference on Communications* (IEEE, 2022), 5511–5515.

16. S. Lokhande, S. Soman, and N. Hiremath, "Quick Learn Approach for Load Forecasting During Covid 19 Lockdown," in *2020 21st National Power Systems Conference (NPSC)* (IEEE, 2020), 1–6.

17. R. Bareth, A. Yadav, S. Gupta, and M. Pazoki, "Daily Average Load Demand Forecasting Using Lstm Model Based on Historical Load Trends," *IET Generation, Transmission & Distribution* 18, no. 5 (2024): 952–962, https://doi.org/10.1049/gtd2.13132.

18. A. Ahmadi, M. Nabipour, S. Taheri, B. Mohammadi-Ivatloo, and V. Vahidinasab, "A New False Data Injection Attack Detection Model for Cyberattack Resilient Energy Forecasting," *IEEE Transactions on Industrial Informatics* 19, no. 1 (2022): 371–381, https://doi.org/10.1109/tii.2022.3151748.

19. S. Sridhar and M. Govindarasu, "Model-Based Attack Detection and Mitigation for Automatic Generation Control," *IEEE Transactions on Smart Grid* 5, no. 2 (2014): 580–591, https://doi.org/10.1109/tsg.2014.2298195.

20. Adapti Grid, Sensewaves, (2024), https://www.sensewaves.io/adaptix-grid/.

21. Flexo, Hive Power, (2024), https://www.hivepower.tech/flexo.

22. Amperon Platform, Operational Demand Forecasting for the Energy Transition, (2024), https://www.amperon.co/platform.

23. e4sight Nectaware, (2024), https://nectaware.com/en/soluzioni.php.

24. A. Arjomandi-Nezhad, A. Ahmadi, S. Taheri, M. Fotuhi-Firuzabad, M. Moeini-Aghtaie, and M. Lehtonen, "Pandemic-Aware Day-Ahead Demand Forecasting Using Ensemble Learning," *IEEE Access* 10 (2022): 7098–7106, https://doi.org/10.1109/ACCESS.2022.3142351.

25. M. R. Baker, K. H. Jihad, H. Al-Bayaty, et al., "Uncertainty Management in Electricity Demand Forecasting With Machine Learning and Ensemble Learning: Case Studies of Covid-19 in the Us Metropolitans," *Engineering Applications of Artificial Intelligence* 123, no. 106 (2023): 350, https://www.sciencedirect.com/science/article/pii/S0952197623005341.

26. M. Yue, T. Hong, and J. Wang, "Descriptive Analytics-Based Anomaly Detection for Cybersecure Load Forecasting," *IEEE Transactions on Smart Grid* 10, no. 6 (2019): 5964–5974, https://doi.org/10.1109/tsg.2019.2894334.

27. I. Arun Abhishek and T. Dafinny, "Anomaly Detection in Load Forecasting Using Arima and Autoencoder," in *2023 IEEE Fifth International Conference on Advances in Electronics, Computers and Communications (ICAECC)*, 2023), 01–06, https://doi.org/10.1109/ICAECC59324.2023.10560105.

28. [DataSet] K. S. Shivran, K. Swire-Thompson, N. Saxena, and S. Das, Anomalous Load Forecast Identifier in Pandemic (2021), https://github.com/CyCISlab/Anomalous-Load-Forecast-Identifier-in-Pandemic.

29. S. De Jong, "Simpls: An Alternative Approach to Partial Least Squares Regression," *Chemometrics and Intelligent Laboratory Systems* 18, no. 3 (1993): 251–263, https://doi.org/10.1016/0169-7439(93)85002-x.

30. G. A. Seber and A. J. Lee, *Linear Regression Analysis* (John Wiley & Sons, 2012).

31. G. Holmes, M. Hall, and E. Prank, "Generating Rule Sets From Model Trees," in *Advanced Topics in Artificial Intelligence: 12th Australian Joint Conference on Artificial Intelligence, AI'99 Sydney, Australia, December 6–10, 1999 Proceedings*, Vol. 12 (Springer, 1999), 1–12, https://doi.org/10.1007/3-540-46695-9_1.

32. J. Quinlan, "Learning With Continuous Classes," in *5th Australian Joint Conference on Artificial Intelligence* (1992).

33. L. Breiman, "Bagging Predictors," *Machine Learning* 24, no. 2 (1996): 123–140, https://doi.org/10.1007/bf00058655.

34. E. Frank, M. A. Hall, and I. H. Witten, *The WEKA Workbench* (Morgan Kaufmann, 2016).

35. J. J. Rodriguez, L. I. Kuncheva, and C. J. Alonso, "Rotation Forest: A New Classifier Ensemble Method," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28, no. 10 (2006): 1619–1630, https://doi.org/10.1109/tpami.2006.211.

36. R. Kohavi, "A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection," in *Ijcai*, Vol. 14, 1995), 1137–1145.