Research Article

Enhanced Memetic Algorithm-Based Extreme Learning Machine Model for Smart Grid Stability Prediction

Manohar Mishra, Janmenjoy Nayak, Bignaraj Naik, and Bhaskar Patnaik

1Electrical and Electronics Engineering Department, Faculty of Engineering and Technology, Siksha O Anusandhan University, Bhubaneswar, India
2Department of Computer Science, Maharaja Sriram Chandra BhanjaDeo (MSCBD)University, Mayurbhanj, Odisha 757003, India
3Department of Computer Application, Veer Surendra Sai University of Technology, Burla, India
4Department of Electrical Engineering, Biju Patnaik University of Technology, Rourkela, India

Correspondence should be addressed to Bhaskar Patnaik; bhaskar7310@gmail.com

Received 3 January 2022; Revised 8 July 2022; Accepted 14 July 2022; Published 31 August 2022

Academic Editor: Harsh Dhiman

Copyright © 2022 Manohar Mishra et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The smart grid is considered a conventional application domain of cyber-physical system (CPS) tools in the electrical utility industry. The physical system dynamics of SG with the assistance of CPS are generally controlled by connected sensors and controllers via a communication link. These CPSs, which rely heavily on an expansive communication network and intelligent computing algorithms, are susceptible to cyber-physical attacks and are also sensitive to various technical, economical, and social factors compromising their stability. Assessment and prediction of the stability of CPSs are very vital in this context. In this work, a novel optimized (memetic algorithm-based) extreme learning machine model for smart grid-CPS stability prediction has been proposed. Here, the teaching-learning-based optimization and simulated annealing techniques are used to design the memetic algorithm. The experimental result regarding the proposed model is then compared with other contemporary machine learning and deep learning models.

1. Introduction

Use of the state-of-the-art technologies in the field of communication and computational ability has helped evolve most industries as smarter in terms of improved efficiency, productivity, quality of service, etc. However, the power system delivery grid (electric grid) has largely remained nonmodernized as compared to other sectors. The reasons can be attributed to the fact that grids have been built and developed over the years primarily as mechanical systems without attention to the possible future need for refurbishment or technological overhauling. With burgeoning demand for power, there has been a large-scale expansion of grid infrastructure, and manual monitoring of the grid has been increasingly difficult and challenging [1]. Thus, a certain level of computation-based automation in monitoring and controlling has been witnessed over the decade or so. However, it hardly suffices considering the new challenges in the form of changing grid dynamics and the need for enhanced grid performance in terms of efficiency, reliability, and resiliency. Although the conventional electrical power system (EPS) remains saddled with the aforementioned deficiencies, the changing scenarios ushered in due to fast depleting conventional energy sources and resulting technological developments in the field of renewable energy sources (RESs) have brought out newer challenges. Penetration of renewable energy resources at low voltage and distribution level of the EPS convert the grid topology from a centralized system to a distributed generation system. Although DGs help to supplement the energy requirements, they also pose many technological challenges, primarily as the power flow gets bidirectional leading to relay miscoordination. On the other hand, the availability of RESs also has given a fillip to the concept of microgrid technology,

10 pages
https://doi.org/10.1155/2022/8038753
which not only helps cater to the local load demand but also adds much to the system stability. Needless to say, micro-grids being one of the key smart grid components are going to be embedded prominently. However, the implementation of the same, as mentioned earlier, poses many challenges. The issues and challenges associated with implementing these cutting-edge technologies are enumerated in [2]. The complexities involved in the successful realization of a smart grid are very daunting, but the benefits accrued are also of immense importance. In a smart grid scenario, the monitoring and control of the EPS rest heavily on an expansive communication network to gather information generated by numerous sensors placed at various locations in the EPS. A huge amount of data thus collected needs to be processed and subjected to computational algorithms which can provide the appropriate and accurate decision in the shortest possible time. In short, we are veering towards an EPS which is heavily reliant on intelligent computational techniques with fast communication networks serving as the backbone. To be precise, the present-day EPS is smartening up as cyber-physical systems (CPSs). CPS is a strong linkage between cyber technologies and the physical system in order to deal with the operational and controlling complexities in a smart grid that are otherwise manually impossible to execute. NIST cyber-physical system website defines CPS as follows [3]:

“Cyber-physical systems (CPSs) comprise interacting digital, analog, physical, and human components engineered for function through integrated physics and logic.”

A more generalized definition is given in [4] as follows:

“CPS can be characterized as physical and engineered systems whose operations are monitored, controlled, coordinated, and integrated by a computing and communicating core.”

In the industrial smart grid perspective, the concept of CPS embodies physical systems such as the network of power infrastructure strongly interfaced with control, intelligence, processing, and information. The CPS enables the smart grid to deliver its expected functional features, which are as follows [5]:

(i) Minimizing the need for deployment of additional power plants through effective power management
(ii) Adaptive and self-healing protection mechanism in order to ensure high reliability and resiliency of the EPS
(iii) Facilitate plug-in of DGs to the grid to meet the localized power demand
(iv) Exhibit flexibility in the distribution process to supply the right amount of power to the various types of loads connected to the grid.
(v) Predict the short-term and long-term power demand
(vi) Help reduce pollution through environment-friendly renewable energy sources
(vii) Help minimize the price of electricity

The above functionalities of an industrial smart grid that ultimately ensures the stability of the network are possible by using intelligent computing algorithms that comprise the backbone of the cyber technology in a CPS. Applications of these intelligent computational algorithms, such as advanced optimization techniques, expert systems, fuzzy logic, neural network, and deep neural network are used to design schemes meant for control, power management, power system protection, power trading, predisaster, and post-disaster management in order to have enhanced power system resiliency and all other such aspects of the smart grid concept that help to add to its stability.

In this work, an optimized extreme learning machine (ELM) is suggested to predict the stability of the industrial smart grid (SG). The performance of the proposed model is then compared with other contemporary machine learning (ML) and deep learning (DL) models.

The steps involved in this work can be presented as follows:

(1) Initially, the SG data have been retrieved from the UCI-ML repository [6] and processed through a preprocessing step such as normalization and label encoding.
(2) A novel memetic algorithm-based optimized extreme learning machine (ELM) is proposed for the training and testing of the data. Here, the TLBO and SA techniques are used to design the memetic algorithm.
(3) This extracted SG dataset is then processed through the proposed optimized ELM model for training and testing purposes.
(4) The performance of the proposed ML model is then compared with other contemporary ML and DL approaches using a few key indexes, such as accuracy, precision, recall, and F-score.

In short, our key contributions toward the prediction of the SG stability can be summarized as follows:

(a) A novel memetic algorithm-based ELM model has been proposed to predict the stability status of SG.
(b) The collected SG data are then trained and tested through the proposed ML model.
(c) An accuracy of 99.75% is attained through the suggested model. Comparative analysis with other contemporary ML and DL techniques indicates the superiority of the proposed model.

The organization of the remaining parts of this article is presented as follows: Section 2 reviews the current literature associated with the application of intelligent computational techniques in various aspects of SG functionalities. Section 3 presents the proposed memetic algorithm-based ELM architecture in detail. The experimentation and result analysis has been carried out in Section 4. Section 5 highlights the concluding remark with the future direction.
2. Literature Survey

A smart grid vastly relies on the accumulation of a huge amount of data procured from numerous sensors placed at different strategic locations of the EPS, fast and robust internet-based communication channels, and of course a fast-acting intelligent computation algorithm. It is imperative that handling and managing such huge data, protecting the communication channel from cyber infringements, and engaging a fast-acting computational technique for accurate prediction of the set objective are vital for realizing a smart grid. Smart grid deployment thus involves huge complexity. Advanced intelligent systems, with techniques such as ML, DL, reinforcement learning (RL), deep reinforcement learning (DRL), and SG realization are becoming feasible [7].

In the above context, the authors in [8] enumerate how the use of big data analysis in conjunction with intelligent models helps to resolve the issue of processing these huge data in an SG. Various applications of big data in the SG perspective are listed in [9]. Threat to the communication network in the form of covert data integrity assault (CDIA) can be detrimental to the reliability and safety of smart grid functionalities. These smartly designed CDIAs can easily outwit the conventional bad-data detector employed in SG control hubs. The authors in [10] have proposed an unsupervised ML-based model to identify CDIAs in SG communication grids using an unlabeled dataset. The ML algorithm employed is called isolation forest. The security system against CDIAs generally is a three-tier structure. 'Protection,' 'intrusion detection system (IDS),' and 'alleviation,' are the first, second, and third tier, respectively. Through shielded communications and data safeguarding measures, the first tier ensures protection of the communication channel against the majority of CDIAs. In the event of the protection tier getting violated, the IDS as the second tier of defense detects the intrusion and generates precautionary signals for the operators to take up preventive measures against the CDIA. The recent literature enumerates many ML-based IDSs [11–17]. The authors in [11, 12] have depicted the application of ML algorithms in identifying unscrupulous user activities in smart grid communication channels. The authors in [13] have demonstrated the utility of several ML algorithms in the detection of CDIAs at the physical layer of a smart grid. The authors in [14] have employed a support vector machine (SVM) classifier in the above context. The authors in [15] have proposed another ML-based model to detect time synchronization assault (TSA). The authors in [16] proposed the Euclidean-distance-based ML model to predict the CDIAs. The authors in [17] have suggested a genetic algorithm (GA) combined with SVM, for future selection and classification purposes, respectively, in order to detect CDIAs. Most of the CDIA recognition techniques using machine learning as stated in the collected works have considered supervised learning on labeled data only. The third defense tier, known as alleviation, serves as a kind of restoration system, which helps restore the reliable system operations once the CPS-assault recognition message is established at the SG control hub. Many optimizations and intelligent techniques too are engaged in this tier for the intended purpose.

From a smart grid stability perspective, it is very essential that zone-specific accurate load forecasting must be carried out in order to meet the dynamic energy demands. Various ML techniques can be used for such purposes. Based on historical data on weather, load variation pattern, and energy generation, these ML algorithms make an accurate prediction of load demand in specific regions [18, 19]. In [19], a deep neural network (DNN) model is employed for generation and load demand forecasting, where DNN has proven better than the contemporary regression model. In [20], the authors have offered a big data framework for distributed processing to predict energy load demand. Here, the MLib-ML library is used for assessing the performance of different regression models. The stability of SGs, which enables smart cities, is challenged by the dynamic energy consumption due to the household appliances in these smart cities. IoT technologies along with ICTs present many energy management techniques to address the issue. The authors in [21] suggested a consumption prediction technique based on a probabilistic data-driven prognostic technique developed on a Bayesian network (BN) framework. Elsisi and Tran proposed a unified Internet of thing (IoT) architecture to manage the issue of cyberattacks using a DNN model having a rectified linear unit [22]. This stated system can supervise the automated guided vehicles reliably and securely. Elsisi et al. suggested an effective online fault diagnosis system against cyberattacks and data uncertainties using an IoT enabled DL model [23].

The cost of power also plays an important role in ensuring the stability of the distributed power systems. The authors in [24] have proposed a decentralized SG control model to ensure demand-side management in the grid by analyzing the electricity price versus grid frequency deviation. The authors have also implemented an optimized data matching ML technique and the transparent open box learning model to realize dynamic SG stability.

Against the backdrop of the above discussion on various factors of smart grid instability and the possible intelligent methods to mitigate them, it is also essential that there should be an efficient scheme for the estimation and prediction of SG stability in place. The authors in [25] have proposed an ML-based smart grid stability forecasting scheme. The proposed method employs three different genetic algorithms in the future selection stage and four different ML classifiers including the GBM algorithm. The authors in [5] proposed a few DL-based models such as recurrent neural network (RNN), long short-term memory (LSTM), and gated recurrent unit (GRU) for prediction of SG stability.

It can be inferred very well from the above survey that there is no single comprehensive solution to tackle the challenges of stability in an SG. Also, there is a need for further research in the area of stability prediction as evident from the scarcity of the literature in this regard. In the present study, it is envisaged to design the novel memetic algorithm-based ELM model for smart grid stability prediction. The proposed approach shows better results than the
traditional ELM technique and other advanced ML and DL techniques.

3. Proposed TLBO-SA-ELM Approach

The performance of the ML model is generally enhanced by integrating the optimization algorithm [26–29]. In this work, a memetic algorithm based on TLBO and SA is proposed to increase the performance of the ELM classifier. Afterward, the proposed optimized ELM is trained and tested using the SG dataset.

A single metaheuristic algorithm cannot solve all optimization problems and is usually less effective for high-dimensional SG datasets. Therefore, there always remains a possibility to design an improved search approach and develop novel optimization techniques which can be used to solve gene selection problems in the SG dataset. This is our major inspiration behind the scheming of this proposed memetic algorithm. The proposed SG stability predictive algorithm is designed using three basic soft computing models such as TLBO, SA, and ELM.

In this work, an ELM-based model [30] with optimized parameters is developed for efficient prediction of smart grid status from smart grid operational features. The proposed problem can be visualized as an optimization problem where the objective is to select the best \( p_i = \{ f_i, \alpha_i, n_h \} \) in \( P = \{ p_1, p_2, \ldots, p_n \} \) (population with \( n \) number of hyperparameter sets). Here, \( p_i \) is the \( i^{th} \) randomly generated hyperparameter value set which is drawn from an allowed range of values as follows: \( f_i \in \text{list}[1, 2, 3, 4, 5, 6, 7, 8] \), \( n_h \in \text{range}[1, 200] \), and \( \alpha \in \text{range}[0.1, 1.0] \). Here, \( f_i, n_h \), and \( \alpha \) are \( i^{th} \) activation function, a selected number of hidden layer, and learning rate. The activation function \( f_i \) is chosen as '1', '2', '3', '4', '5', '6', '7', and '8' for sine, tanh, tribas, sigmoid, hardlim, softlim, Gaussian, and multi-quadratic, respectively. The performance of ELM on the prediction of smart grid status is dependent on these parameters \( f_i, n_h \), and \( \alpha \). Here, the studied problem can be visualized as an optimization problem to get optimal \( P^*_i = \{ f_i, n_h, \alpha_i \} \) in \( P \), which is the optimal parameter set of ELM for solving the identification of the various states of the smart grid. On the given operational SG properties, \( D = \{ \theta_P, s_t \}_{i=1}^{m} \), \( P = \{ P_i \}_{i=1}^{n} \), and modelELM(D\text{Train}, P), where the objective is to find optimal \( P^*_i \) which optimizes the following objective function:

\[
P^*_i = \arg \max_{P \in P} \{ \text{fit}_i = \text{score}(s_t, s_t = \text{ELM}(D\text{Train}, P_i)) \} = \arg \max_{P \in P} \left\{ \text{fit}_i = \text{score}(s_t, s_t) = \frac{1}{m} \sum_{i=1}^{m} I(s_t, s_t) \right\}.
\]

The dataset \( D = \{ \theta_P, s_t \}_{i=1}^{m} \) is the collection of smart grid operational properties \( \theta_P \) with thirteen features and one class label \( s_t \), representing the status of the grid condition, i.e., either 'stable' or 'unstable.' The proposed ELM model has been trained with these instances. In ELM, the prediction of the class label is made using Equation (2), where Houtput (Equation (3)) is the output matrix, \( \beta \) (Equation (4)) is the weight matrix representing the weights between hidden layer neurons and out neurons, and Equation (5) is the prediction matrix.

\[
\text{Houtput} = \begin{bmatrix}
f(b_1 + op_1 \cdot w_1) & \cdots & f(b_L + op_1 \cdot w_L) \\
\cdots & \cdots & \cdots \\
f(b_1 + op_N \cdot w_1) & \cdots & f(b_L + op_N \cdot w_L)
\end{bmatrix}
\]

\[
\beta = [\beta_1, \beta_2, \ldots, \beta_L]_{L \times 1}^T,
\]

\[
\hat{s}_t = Houtput \times \beta,
\]

In this work, the hyperparameters of ELM are optimized with the memetic version of teaching-learning based optimization (TLBO) [31]. Here, the TLBO is integrated with simulated annealing [32] in order to avoid a locally optimal solution.

In this work, the domains of the considered hyperparameters are as follows: activation function \( f_i \), alpha \( \alpha \), and several hidden neurons \( n_h \). Here, \( f_i \) is the mathematical equation that is responsible to determine whether the neuron input is significant for prediction. \( \alpha \) is the controlling parameter for the adjustment of weights and \( H \) is the number of hidden neurons that highly impact the performance and network stability. This work is focused on the process of finding optimal parameters \( P_i \) ELM by using TLBO with SA. The objective function is used to evaluate the parameter combinations \( P_i \) and output a fitness (accuracy) \( \text{fit}_i = \text{ELM}(D\text{Train}, P) \) which indicates how well the set of hyperparameters performs for the considered problem. For the present problem, we have considered 'accuracy' as the evaluation matrix, and it is the objective to maximize the objective function presented in Equation (2). The architecture of the proposed algorithm is presented graphically in Figure 1.

The proposed scheme for hyperparameter optimization starts with random generation of the population of the hyperparameter set \( P = \{ P_i \}_{i=1}^{n} \), where \( P_i = \{ f_i, n_h, \alpha_i \} \) representing \( i^{th} \) instance of the ELM parameter set. The fitness of each \( P_i \) in \( P \) is evaluated (in Algorithm 1, i.e., \( \text{fit}_P = \text{ELM}(D\text{Train}, P) \)) by setting the \( P_i \) in ELM and testing on the data \( D\text{Train} \). Then, the fittest \( P_i \) in \( P \) is \( P_{\text{Teacher}} \). The teaching factor \( tf = \text{rand}(1 + \text{rand}(1)) \) and the population mean \( P_{\text{Mean}} \) are computed in order to use them for the generation of a new population. For each \( P_i \), the \( \text{Pnew} \) is generated as \( P_{\text{new}} = P_i + r \times (P_{\text{Teacher}} - \text{fit}_P \times P_{\text{Mean}}) \), where \( r \) is a random number. The updating of \( P \) is performed by comparing each of \( P_i \) in \( P \) with the corresponding \( P_{\text{new}} \) in \( P_{\text{new}} \). Then, the resultant population is improvised by choosing two solutions \( P_i \) and \( P_j \) randomly from \( P \) and altering them as mentioned in the following equation:

\[
P_{\text{new}} = \begin{cases}
P_i + \text{rand}(1) \times (P_j - P_i) & \text{if } \text{fit}_{P_i} < \text{fit}_{P_j} \\
P_i + \text{rand}(1) \times (P_i - P_j) & \text{Otherwise}
\end{cases}
\]
After improvisation of \( P \), the best solution \( P_{\text{best}} \) in \( P \) is selected based on the highest fitness. Then, the simulated annealing (Algorithm 2) has been applied on \( P_{\text{best}} \) to generate a new solution from \( P_{\text{best}} \) in order to avoid local optimal solutions. The proposed approach of applying simulated annealing with TLBO not only accepts the best solution but also considers the nearer solution to the best solution with some probability. The major steps of applying simulated annealing on \( P_{\text{best}} \) are as follows: (i) random generation of a solution \( P'_{\text{best}} \) from \( P_{\text{best}} \) and \( N(\mathbf{P}_{\text{best}}) \), where it is the neighborhood operation; (ii) computation of the fitness \( f_{\text{best}} \) of \( P_{\text{best}} \) and \( f'_{\text{best}} \) of \( P'_{\text{best}} \) respectively, by calling the procedure \( \text{ELM}(D_{\text{Train}}, P_{\text{best}}) \) and \( \text{ELM}(D_{\text{Train}}, P'_{\text{best}}) \), respectively; (iii) replacement of \( P_{\text{best}} \) with \( P'_{\text{best}} \) if \( f'_{\text{best}} > f_{\text{best}} \), or if \( r \) and \( (1) < p \), where \( p = e^{-\theta/T} \); (iv) decrement of the temperature: \( T = 0.93 \times T \) and checking the maximum trial. If maximum trial is reached, then extract the \( P_{\text{best}} \).

After getting the update \( P_{\text{best}} \) from Algorithm 2 (simulated annealing process), the population \( P \) has been updated by replacing old \( P_{\text{best}} \) with new \( P_{\text{best}} \) (returned from the procedure SA). If a maximum generation is reached or no further improvement on the performance, then we stop and assign \( P^* = P_{\text{best}} \). Finally, Algorithm 3 returns the best hyperparameter set \( P^* \) from the entire final population \( P \). Then, this is set on ELM as \( f_{\text{fit}*} = \text{ELM}(D_{\text{Train}}, P^*) \) and \( f_{\text{fit}^*} = \text{ELM}(D_{\text{Train}}, P^*) \) to get the generalized performance on training data \( D_{\text{Train}} \) and test data \( D_{\text{Test}} \), respectively. The complete process of getting an optimal hyperparameter \( P^* \) can be realized in Algorithm 3.

4. Results and Discussion

In this section, the proposed SG stability prediction method is evaluated through proper experimental analysis.
4.1. Experimental Platform. The experimental setup for the proposed work comprises an online graphical processing unit (GPU) enabled by Google called “Google Colab,” a supercomputer having a Windows 10 Operating system with a core 17 processor and a Python 3.9 programming tool.

4.2. Dataset Description. The SG data used in this experiment have been extracted from the UCI-ML repository [6]. This dataset comprised 10000 samples with fourteen features. The attributes are divided into 12 primary predictive features and 2 dependent variables. The predictive features provide information about the reaction time (\(\tau[x]\)) of smart grid participants (range: 0.5–10s), nominal power (\(p[x]\)) consumed (negative)/produced (positive), and coefficient related to elasticity price (\(g[x]\)). The dependent attributes can be described as follows: the first attribute indicates: the maximal real part of the characteristic equation root (if positive - the system is linearly unstable) and the second dependent attribute states the stability label (class level) of the system (categorical: stable/unstable).

4.3. Performance Evaluation Matrices. In this work, the classification result corresponding to the proposed TLBO-SA-ELM is interpreted through the following performance evaluation matrices.

Confusion matrix (CM): it represents the actual value with respect to the predicted values corresponding to the class level.

CM is a table that visualizes and compares the result obtained from a classifier by presenting the actual values and predicted values (number of samples) with respect to the class levels in terms of correct and incorrect prediction. A sample of the confusion matrix is presented in Figure 2.

From the above confusion matrix, the following six matrices are calculated as presented from the following equations.

Accuracy (A): it signifies the correctness of the classifier. It is mathematically represented as follows:

\[
A = \frac{TS + TU}{TS + TU + FS + FU}
\]  

Precision (P): 
\[
P = \frac{TS}{TS + FU}
\]

Recall (R): 
\[
R = \frac{TS}{TS + FS}
\]

\(F_\beta\)-Score: 
\[
F_\beta = \left(1 + \beta^2\right) \times \frac{PR}{\beta^2(R + P)}
\]

F1_Score (F1): in the calculation of F1_Score, equal weight is given for both precision and recall. Therefore, in equation (10), a beta (\(\beta\)) value of 1 can be considered for the calculation of F1_measure. So, it can be represented as follows:

\[
F1 = 2 \times \frac{PR}{R + P}
\]

4.4. Performance Evaluation of the Proposed TLBO-SA-ELM Model. The collected SG data are divided into 8:2 ratios for training and testing, respectively. The testing result in terms of the confusion matrix is presented in Figure 3.

Figure 3 shows the confusion matrix for ELM and the proposed model for the classification of the SG stability dataset. The proposed TLBO-SA-ELM model detects 703 data as truly stable and 1292 as truly unstable, whereas traditional ELM detects 669 data as truly stable and 1223 as truly unstable. Only 5 data are misclassified through the proposed model. From these confusion matrices, different measuring indices such as precision, recall, accuracy, and \(F_\beta\)-Score are calculated and tabulated in Table I. It can be seen from the table that the proposed model outperforms traditional ELM with respect to all the measuring indices.

The proposed approach is a hybrid model where the parameter of the ELM is optimized through a memetic algorithm comprised TLBO and SA. In order to check the effectiveness of the proposed memetic algorithm, a few other optimization algorithms such as genetic algorithm (GA), particle swarm optimization (PSO), and TLBO are hybridized with the ELM algorithm, and the corresponding result is interpreted in Figure 4. Figure 4 shows the plot of fitness (accuracy) versus the number of generations. It is seen from the figure that the proposed memetic algorithm-
Figure 3: Confusion matrix for (a) traditional ELM and (b) proposed TLBO-SA-ELM.

Figure 4: Fitness changes in various generations.

Let $D = \{\{p_i, s_i\}\}_{i=1}^m$ be the smart grid operational data, where $p_i = \{p_{i,1}, p_{i,2}, \ldots, p_{i,N}\}$ is the $i^{th}$ instance of the smart grid operational feature value and $s_i$ is the status of the grid.

1. Randomly generate bias $b_i$, $i = 0$ to $L$, and weight $w_i$, $i = 0$ to $L$.
2. Calculate the hidden layer output function $H_{\text{output}}$ by using the selected activation function $f(\cdot)$.

$$H_{\text{output}} = \begin{bmatrix} f(b_1 + o_{p_1} \times w_1) & \cdots & f(b_L + o_{p_L} \times w_L) \\ \vdots & \ddots & \vdots \\ f(b_1 + o_{p_N} \times w_1) & \cdots & f(b_L + o_{p_N} \times w_L) \end{bmatrix} \in \mathbb{R}^{N \times L}.$$

3. Compute the output weight matrix $\hat{\beta} = H_{\text{output}}^T \times s$, which maximizes the objective function $\|H_{\text{output}} \times \hat{\beta} - s\|_{\text{min}} = \|H_{\text{output}} \times \hat{\beta} - s\|_2$. Here, $H_{\text{output}}^T$ is the Moore–Penrose generalized inverse of the Houtput.

$$H_{\text{output}}^T = (H_{\text{output}} \times H_{\text{output}})^{-1} \times H_{\text{output}}^T.$$

Perform the prediction by using $\hat{\beta}$ on the data $\hat{s} = \alpha \times (H_{\text{output}} \times \hat{\beta})$, $\alpha$ is the learning rate. $\hat{s}_i^k = \begin{cases} 1 & \text{ if } (s_i = k) \\ -1 & \text{ if } (s_i \neq k) \end{cases}$, $k = 1, 2, \ldots, c$

Predict the final class label as $\hat{s}_i^c = \arg \max_{k=1,2,\ldots,c} (\hat{s}_i^k)$

Return score of the prediction $fit_{\text{best}} = \text{score}(s, \hat{s})$

Algorithm 1: $fit_{\text{best}} \leftarrow \text{ELM}(X, P_i)$.

1. Randomly generate a solution $P'_{\text{best}}$ from $P_{\text{best}}$ and $N(P_{\text{best}})$ (neighbor structure of $P_{\text{best}}$).
2. Compute the fitness $fit'_{\text{best}}$ of $P_{\text{best}}$ and $P'_{\text{best}}$, respectively, by calling the procedure $\text{ELM}(D_{\text{Train}}, P_{\text{best}})$ and $\text{ELM}(D_{\text{Train}}, P'_{\text{best}})$, respectively.
3. If $fit'_{\text{best}} > fit_{\text{best}}$, then assign $P_{\text{best}} = P'_{\text{best}}$.

Else calculate $p = e^{\frac{\text{draw}}{T}}$. If $r$ and $(1) < p$, then assign $P_{\text{best}} = P'_{\text{best}}$

4. Decrease the temperature, $T = 0.93 \times T$

5. If the maximum trial is reached, then return $P_{\text{best}}$

6. Else go to step 1.

Algorithm 2: $P_{\text{best}} = \text{SA}(P_{\text{best}}, N(P_{\text{best}}), D_{\text{Train}})$. 
Let $D = \{op_i, st_i\}_{i=1}^n$ be the smart grid operational data, where $op_i = \{op_{i1}, op_{i2}, \ldots, op_{iN}\}$ is the $i^{th}$ instance of the smart grid operational feature value and $st_i$ is the status of the grid.

1. Randomly generate the population $P = \{P_i\}_{i=1}^n$, where $P_i = \{f_i, nh_i, \alpha_i\}$ representing $i^{th}$ is an instance of the ELM parameter set.
2. Evaluate the fitness of each $P_i$ in $P$. fit$_P = ELM(D_{\text{Train}}, P_i)$.
3. Select fittest $P_i$ in $P$ as $P_{\text{Teacher}}$.
4. Compute teaching factor $tf = r$ and $(1 + rand(1))$ and population mean $P_{\text{Mean}}$.
5. Generate $P_{\text{new}}$ from $P_i$ in $P$. $P_{\text{new}} = P_i + r \times (P_{\text{Teacher}} - tf \times P_{\text{Mean}})$, where $r$ is a random number.
6. Perform the updating $P$ by comparison of each $P_i$ in $P$ with the corresponding $P_{\text{new}}$ in $P_{\text{new}}$.
7. For each pair of randomly chosen $P_i$ and $P_j$ in $P$, improvise them by using the following equation:
   $$P_{\text{new}} = P_i + r \times (P_{\text{Teacher}} - tf \times P_{\text{Mean}})$$
   otherwise
   $$P_{\text{new}} = P_j + r \times (1 - P_{\text{Teacher}})$$
   if $P_i < P_j$.
8. Find the best solution $P_{\text{best}}$ in $P$ based on the highest fitness.
9. Generate a new solution from $P_{\text{best}}$ by using a simulated annealing process $P_{\text{best}} = SA(P_{\text{best}}, N(P_{\text{best}}), D_{\text{Train}})$.
10. Update the population by replacing old $P_{\text{best}}$ with new $P_{\text{best}}$ (returned from the procedure SA).
11. If a maximum generation is reached, then assign $P^* = P_{\text{best}}$.
12. Return $P^*$.

**Algorithm 3:** $P^* \leftarrow \text{ELM-TLBO-SA}(D_{\text{Train}}, P)$.

### Table 1: Performance metrics of prediction models.

<table>
<thead>
<tr>
<th>Prediction Models</th>
<th>Precision</th>
<th>Recall</th>
<th>F1 Score</th>
<th>F2 Score</th>
<th>ROC-AUC</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>DT</td>
<td>0.7488</td>
<td>1.0</td>
<td>0.8871</td>
<td>0.702315</td>
<td>0.61</td>
<td>74.8</td>
</tr>
<tr>
<td>NB</td>
<td>0.6485</td>
<td>1.0</td>
<td>0.786775</td>
<td>0.585075</td>
<td>0.5</td>
<td>64.85</td>
</tr>
<tr>
<td>LR</td>
<td>0.6485</td>
<td>1.0</td>
<td>0.786775</td>
<td>0.585075</td>
<td>0.5</td>
<td>64.85</td>
</tr>
<tr>
<td>RF</td>
<td>0.946227</td>
<td>0.909020</td>
<td>0.927251</td>
<td>0.5795</td>
<td>0.906857</td>
<td>90.75</td>
</tr>
<tr>
<td>XGBoost</td>
<td>0.910437</td>
<td>0.995373</td>
<td>0.951012</td>
<td>0.5944</td>
<td>0.907359</td>
<td>93.35</td>
</tr>
<tr>
<td>ELM</td>
<td>0.972951</td>
<td>0.942945</td>
<td>0.957713</td>
<td>0.5986</td>
<td>0.947290</td>
<td>94.6</td>
</tr>
<tr>
<td>Proposed model</td>
<td>1.0</td>
<td>0.996144</td>
<td>0.998068</td>
<td>0.6197</td>
<td>0.996086</td>
<td>99.75</td>
</tr>
</tbody>
</table>

### Table 2: Performance comparison with existing approaches.

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recurrent neural network (RNN) [5]</td>
<td>96.60</td>
</tr>
<tr>
<td>Gated recurrent unit (GRU) [5]</td>
<td>97.30</td>
</tr>
<tr>
<td>Multidimensional long short term memory (M-LSTM) [5]</td>
<td>99.07</td>
</tr>
<tr>
<td>Proposed model (ELM-TLBO-SA)</td>
<td>99.75</td>
</tr>
</tbody>
</table>

**4.4.1 Comparative Analysis.** In order to validate the performance of the proposed algorithm applied to the SG stability dataset, the corresponding experimental results are compared with the output of several other contemporary ML models. In this regard, the results corresponding to a decision tree (DT), naïve bias (NB), linear regression (LR), random forest (RF), and extreme gradient boosting (XGBoost) are depicted in Table 1. A similar ratio (8:2) of the training and testing dataset is considered during these experimentations.

It can be analyzed from the table that NB and LR show the worst performance (accuracy: 64.85) compared to XGBoost having an accuracy of 93.35%. It can also be analyzed that the performance of XGBoost is observed to be less than the proposed model with a very high margin.
(accuracy > 6.4%). Finally, it can be concluded that the proposed approach performs better than all other ML models with respect to each performance indices. In addition to this, the performance of the proposed approach is compared with the result of different deep learning models (such as RNN, GRU, LSTM, and MLSTM) applied to a similar dataset [5]. This comparative result analysis is presented in Table 2. It can be analyzed from the table that MLSTM performs better than RNN, GRU, and LSTM; however, the proposed algorithm (TLBO-SA-ELM) outperforms in all aspect. Figure 5 shows the overall comparative results graphically.

5. Conclusion

Smart grids are identified with the cyber-physical system used for intelligent management of power generation and dissipation ensuring quality power supply at the most economical price. However, the threat to CPS and several issues including a threat to CPSs severely affects the stability of a smart grid. Machine learning techniques play a vital role in predicting the stability of the smart grid. In this work, a novel memetic algorithm-based ELM model is introduced to predict the stability of the SG. The proposed optimized ELM algorithm is tested on the SG dataset extracted from the UCI-ML repository. The proposed model achieved 99.75% accuracy and 100% precision in classifying the stability status of the smart grid. To justify the effectiveness of the proposed model, a detailed comparative analysis is carried out with other contemporary machine learning models (such as decision tree, linear regression, random forest, naïve bias, XGBoost, and ELM) and previously published work based on deep learning (such as LSTM, GRU, RNN, and MLSTM). 99.8072% of ROC is achieved through the proposed algorithm which outperformed the other comparative models. The effectiveness of the proposed memetic algorithm based on TLBO and SA is additionally compared with several other optimization techniques (such as particle swarm optimization, genetic algorithm, and TLBO) hybridized with ELM. The future scope of the work may deal with the analysis of dynamic power requirements, reliability, and resiliency of SG.

Abbreviations

SG: Smart grid
CPS: Cyber-physical system
EUI: Electrical utility industry
ELM: Extreme learning machine
TLBO: Teaching-learning-based optimization
SA: Simulated annealing
EPS: Electrical power system
RES: Renewable energy resources
DG: Distributed generation
ML: Machine learning
DL: Deep learning
CDIA: Covert data integrity assault
IDS: Intrusion detection system
SVM: Support vector machine
TSA: Time synchronization assault
GA: Genetic algorithm
DNN: Deep neural network
IoT: Internet of thing
BN: Bayesian network
RNN: Recurrent neural network
GRU: Gated recurrent unit
LSTM: Long short-term memory
CM: Confusion matrix
f: Activation function
α: Alpha
nh: Number of hidden neuron
P*: Hyperparameter
DTrain: Training data
DTest: Test data.

Data Availability

The data used in this study are openly available in (UCI-ML Repository) at (https://archive.ics.uci.edu/ml/datasets/Electrical+Grid+Stability+Simulated+Data+) [6].

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References


