Received: Added at production | Revised: Added at production | Accepted: Added at production
DOI: xxx/xxxx

ARTICLE TYPE

Comparative Analysis of Machine Learning Algorithms for prediction of Smart Grid Stability†

Ali Kashif Bashir[1] | Suleman Khan[2] | Prabadevi B[3] | N Deepa[4] | Waleed S. Alnumay[5] | Thippa Reddy G[6] | Praveen Kumar Reddy M[7]

[1]Department of Computing and Mathematics, Manchester Metropolitan University Manchester, Manchester, UK
[2]Air University Islamabad, Islamabad , 44000, Pakistan
[3]School of Information Technology and Engineering, Vellore Institute of Technology, Tamilnadu, India
[4]School of Information Technology and Engineering, Vellore Institute of Technology, Tamilnadu, India
[5]Computer Science Departement, King Saud University, Riyadh, Saudi Arabia
[6]School of Information Technology and Engineering, Vellore Institute of Technology, Tamilnadu, India
[7]School of Information Technology and Engineering, Vellore Institute of Technology, Tamilnadu, India

Correspondence
*Thippa Reddy G, *Praveen Kumar Reddy M Email: thippareddy.gy@vit.ac.in, praveenkumarreddy@vit.ac.in

Abstract

The global demand for electricity has visualized high growth with the rapid growth in population and economy. It thus becomes necessary to efficiently distribute electricity to households and industries in order to reduce power loss. Smart Grids (SG) have the potential to reduce such power losses during power distribution. Machine learning and artificial intelligence techniques have been successfully implemented on SGs to achieve enhanced accuracy in customer demand prediction. There exists a dire need to analyze and evaluate the various machine learning algorithms, thereby identify the most suitable one to be applied to SGs. In the present work, several state-of-the-art machine learning algorithms, namely Support Vector Machines (SVM), K-Nearest Neighbor (KNN), Logistic Regression, Naive Bayes, Neural Networks, and Decision Tree classifier, have been deployed for predicting the stability of the SG. The SG dataset used in the study is publicly available collected from UC Irvine (UCI) machine learning repository. The experimentation results highlighted the superiority of the Decision Tree classification algorithm, which outperformed the other state of the art algorithms yielding 100% precision, 99.9% recall, 100% F1 score and 99.96% accuracy.

KEYWORDS:
Smart Grid, Stability, Machine Learning Algorithms

1 | INTRODUCTION

The immense growth in the global population and economy, along with the rapid surge in urbanization has high possibilities to increase the demand in energy consumption in the succeeding future years. Electricity being an important source of energy, can be produced from various sources such as water, wind, solar cells, fossil fuels, thermal and nuclear plants. Also, with the advancement and extensive growth of our population, the demand for electricity is ever increasing, which automatically impacts the demand for higher electricity production. Power generation, transmission and its distribution are the most critical issues involved in electricity management. It is a known fact that electric grid is an interconnected network that connects the consumers to the producers of electricity and transfers the energy from producer to consumer. It comprises of the power stations generating electricity, substations for stabilizing the electricity voltage based on use, transmission lines (the carrier of electricity) and the distribution lines which connect the customers. The conventional electrical grids adopt a centralized structure with millions of

†Comparative Analysis of Machine Learning Algorithms for classification of Smart Grid dataset

the aforementioned components. Increasing the load of an electric grid thus create possibilities of generating additional overhead resulting in power quality issues. Therefore, the need for installation of new plants emerge. On the other side, these grids do not have proper prediction system in order to predict intermittent power outages, their causes, response delay, storage requirements and resource utilization [1].

Researchers have identified that the current electrical power system has not experienced any change for the past hundred years. It is obvious that with the increase in population, there exists a high demand for electricity. The challenges of the conventional power system are lack of visibility, usage of mechanical switches which result in reduced response time, deficiency of monitoring, and power control. The additional inducing factors that need a new grid technology are change in climatic conditions, need for energy, failure of components, increase in population, lack of storage for energy, demand for fossil fuels, decrease in electric power generation, unilateral communication, and various other problems. Thus, in order to overcome such challenges, new grid infrastructure is required. The next-generation electric power infrastructure, namely the smart grid (SG) emerges as a prominent technology to fulfill such high prioritized necessities to enhance the quality of modern human life [2].

Smart Grid is a new digital electric power grid technology that allows 2-way communication to increase security [3,4,5], efficiency, reliability of the electric power systems for higher generation of electrical power through contemporary communication technologies. It is a two-way energy delivery and transportation system which allows its consumers to make decisions pertinent to energy. SGs basically helps to reduce the electricity bill paid by customers. It also contributes to an increase in security measures taken during the consequences of natural disasters and other human attacks. On the contrary, it also ensures a significant decrease in risks resulting in loss of human lives and other physical infrastructure relevant to conventional grid-related activities. Considering the implementation aspects, SGs assist in integrating electric vehicles and modernizing the transport division. In the realm of global warming issues and the need for optimal energy utilization, Smart grids help to reduce the energy wastage and also environmental pollution caused by the discharge of gases from the greenhouse [6,7].

From the perspective of comparative analysis, the conventional grid provides one-way communication, which is limited to energy users, whereas SG provides massive two-way communication. Power quality issues are solved very slowly in the conventional grid, whereas a rapid self-healing facility is available in the case of SG. The traditional grid system is more prone to cyber-attacks and natural disasters with a much slower response. The SG, on the contrary, is much durable during natural calamities and cyber attacks. The conventional grid system responds slowly to system disturbances, whereas the SG provides automatic detection and response to the problems and has a much lesser impact on customers. In the conventional grid system, power flow control is quite restricted, whereas much vast in SG.

In SG, various components are integrated with sensor nodes [8] and communication paths to provide inter-operability in business, manufacturing, and residential applications. The objective is to avoid power turbulence caused by component failures, natural disasters, and capacity constraints by providing online intelligent electric power monitoring and control system. The SG offers state-of-the-art services with two-way communication, an intelligent system, automatic monitoring, and self-remedial skills. SG also renders support in demand management by predicting energy usage. The usage of electricity can act as criteria for providing incentives to the consumers by revising their utilization pattern thereby, efficiency can be improved. The aforementioned can be achieved by distributing energy to customers with improved reliability and security features [9].

As a huge amount of data from various applications need to be analyzed and controlled, the communication requirement plays a significant role in SG infrastructure. Therefore it is very critical to identify the best communication infrastructure to provide cost-effective, reliable, and secure service for the entire system. Numerous technologies have been found in the case of SG communication with two media, namely wired and wireless, which can be applied for data communication between smart meters and other electrical components. Wireless communication technologies are low cost, and connection establishment is comfortable even in unreachable and problematic areas. But the signal may get weakened due to the nature of the transmission lane, wherein wireless solutions depend on batteries. On the contrary, wired technologies do not have any intervention problem, as they do not depend on batteries.

In the SG infrastructure, two types of information flows are involved. The first information flow is from electrical applications and sensors to smart meters. The second flow is from smart meters and the SG utility data centers. Wireless communication technologies such as 6LowPAN, Z-wave, ZigBee, etc., can be utilized for the first information flow, and internet or cellular technologies can be applied for the second type of information flow. The same communication technology may not be suited for various other applications. Based on the application domain, the choice of communication technologies should thus be made. [2].

Smart Grid provides smart solutions to almost all spheres and activities concerned with electricity [10,11,12]. It offers the following features: real-time monitoring of electricity consumption as per the type of application, dynamic pricing (on-demand pricing), faster and effective restoration of electricity after the power outage, in-house electrical displays, altering the electricity usage during day time based on the pricing signals (offering an electricity incentive to consumer) and usage levels, transferring of the role of consumer to producer (both consume and produce power), tracking the electricity usage through the use of web apps and mobile apps.

SGs ranging from small sizes to large ones have been deployed in most of the developed countries [13]. As an example, a small-sized grid network was used at Gazi University, Turkey, which connected the wind, solar, battery storage and diesel-powered microgrid systems. Similarly, a large-sized SG network was constructed at Jeju island, South Korea. All these SGs incorporated the features that have been discussed earlier. It is important to mention the fact that cloud computing technologies have extensive applications in SGs. Smart Grid uses Information and Communication Technologies (ICT) to enable communication among the various resources. SG involves enormous data accumulation during electricity generation, transmission and distribution. Cloud computing [14,15] can thus be utilized in energy management, security services and information management in SGs [16]. Zigbee technology can be utilized in an SG for monitoring and controlling information, fault locating and transmission lines monitoring [17].

The Internet of Things [18] play an important role in the deployment of energy meters. In connection to this, SGs can be used in smart home automation, smart building automation, smart city automation, smart substation and feeder automation [9].

Although SG encompasses various technologies to resolve the issues with classical electricity networks, it has associated problems that need to be addressed. There are basically two categories, technical and socio-economic issues [19,20]. The technical challenges include lack of policies, storage concern, cybersecurity vulnerabilities while connecting the grid to cyber-physical systems, inadequacy in grid infrastructure to accommodate the future needs and demands in the storage of intermittent power generation, voluminous data management from different components of the grid, grid stability concerned with power-sharing, system inertia, power oscillation and power reservation. Technical challenges, on the other hand, include energy management in using electric vehicle involving power flow from vehicle to grid, grid to vehicle and vehicle to vehicle. Some of the socio-economic challenges include stakeholder management, lack of awareness, lack of policies and substantial capital investments. This also creates additional issues relevant to electricity charges, new tariff, health issues related to radiofrequency usage, privacy, fear of obsolescence and power theft. Security remains to be one of the primary concerns as the SG can be breached by both wired and wireless communication networks. Artificial intelligence and its subset machine learning algorithms [21,22,23] can be employed in predicting the problems in SG that aids in taking precautionary steps. In this work, the most prominent technical challenge, predicting the stability of an SG, is considered because it determines the reliable power transmission in almost 50% of the SGs [24]. Figure 1 depicts the SG environment incorporated with Artificial Intelligence(AI) technology.

The main contributions of this work include:

- A detailed review of the existing literature on application of ML algorithms on SG is presented.
- A thorough investigation on performance of several ML algorithms for predicting the stability of SG.
- Comparative analysis of the performance of ML algorithms implemented on SG dataset against the recent work on Deep Learning based model. [25]. The results highlighted the superiority of the ML algorithms which outperformed the deep learning based model considering the size of the dataset being very less.

The rest of the paper is organized as follows: A thorough review of the existing works on application of ML algorithms on SG is presented in Section 2. Section 3 presents the proposed model. The performance analysis of the ML algorithms is presented in Section 4. The conclusion and future directions are presented in Section 5.

2 | LITERATURE SURVEY

A survey on various research solutions adopted methodologies, outcomes, and limitations of existing works on smart grids are presented in this section. The SG, which replaces the conventional electricity grid, promises to carry out a 2-way communication. A complex system is used with the help of electric vehicles to distribute the power disseminated from heterogeneous sources [26]. This adds additional overhead to the SG modeling, controlling of its components in order to optimize its performance. Therefore it requires consistent monitoring of stability robustness, efficiency and reliability in different operating conditions. Various researchers have used machine learning algorithms like support vector machine (SVM), linear regression, K-nearest neighbors (KNN), Ridge Regression, Artificial neural network(ANN), Random Forest, stochastic gradient descent, gradient boosting, Extra

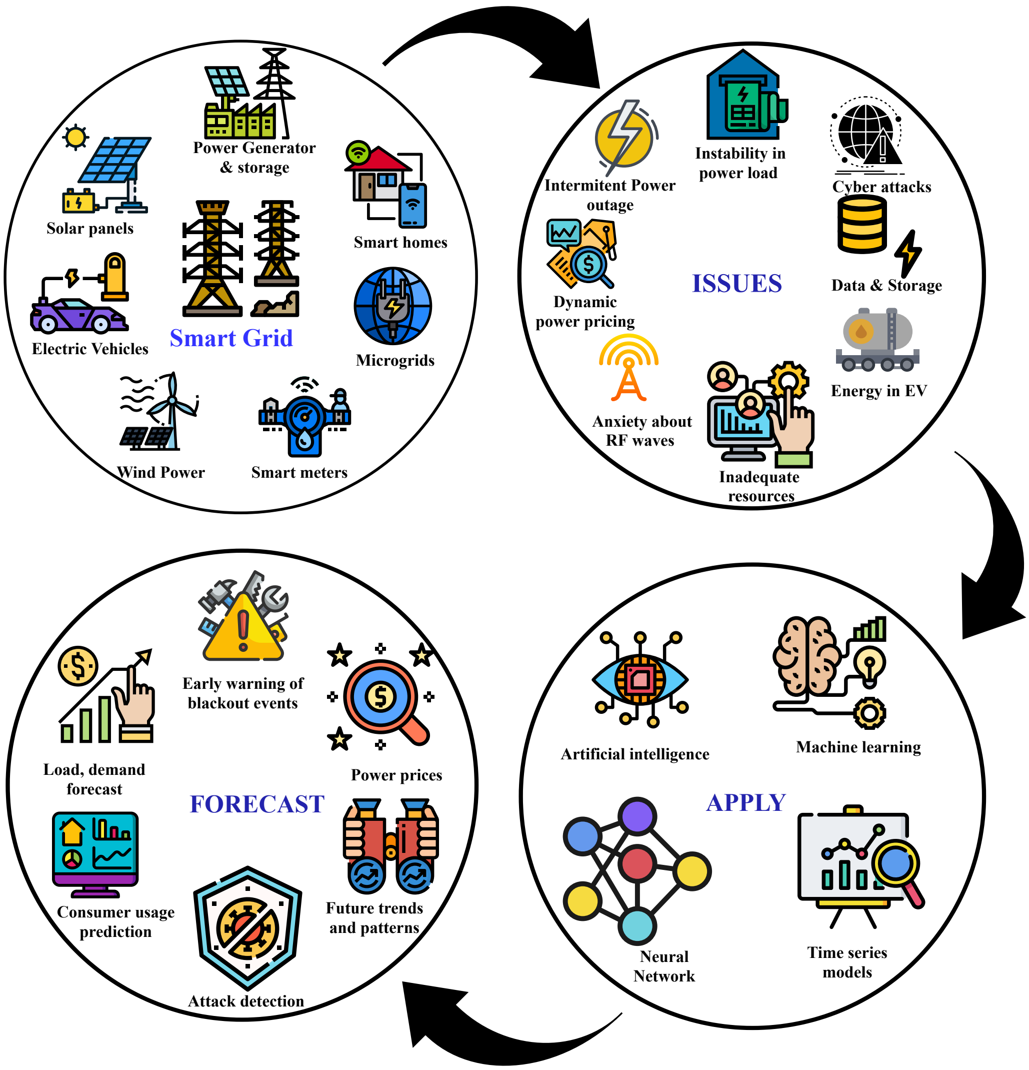


FIGURE 1 SG environment incorporated with Artificial Intelligence(AI) technology.

trees Regressor to predict the load in SG. Deep learning models like Long short term memory (LSTM) neural network, recurrent neural network (RNN), feed-forward neural network (FFNN), Backpropagation neural network, Multi-Layered Perceptron (MLP) are compared with machine learning models in these studies. Feature selection algorithms are applied for selecting the best input values, and hyperparameter tuning is performed to avoid the overfitting of data in machine learning models.

Bouktif et al. have used the LSTM based RNN model for predicting the electric load to handle demand related aspects in SGs[27]. They have applied best-of-breed machine learning algorithms to choose a benchmarking model and a wrapper for feature selection (namely Extra tree regressor and regressive feature elimination) to select the best features as input to the model. The genetic algorithm was used to find the layers and optimal time lags in the LSTM model. They have compared their results with machine learning algorithms and found that LSTM-RNN has less forecasting error than others. But this may not perform well in varying data sets as deep learning selects only good patterns from huge data in different time series and more training data set. Therefore this model has to be evaluated with different datasets.

SGs are transiting towards demand-based power supply services to the consumers. Therefore there is a mandatory need to predict the consumer load. An attempt is made to identify whether the existing short term load forecasting(STLF) model provides better accuracy or the anthropologic-structural data provides better accuracy in forecasting individual consumer household load[28]. To perceive the best forecasting model for individual consumer load, a short term multiple load forecasting (STMLF) model was proposed based on anthropologic structural data of the consumer's house. SLMTF has the potential to forecast multiple loads at different time series using a single model. The study used backpropagation neural network and SVM to compute predictions on the STMTF model. The results highlighted the fact that STMLF is 7% more accurate than SLTF and also reduced 50% of the error. The study justified the enhanced accuracy in the case of anthropologic structural data in comparison to the SLTF model. ANN was trained with three weeks of data in the household, which predicted the power load per hour for the next day.

Normally SLTF is implemented on nation wise or region wise data. Hernandez et al., proposed an ANN-based approach for SLTF within microgrids[29], which consisted of three stages: Pattern recognition using Self Organizing Map (SOM), K-means clustering for partition and MLP for demand forecasting in the individual cluster. The ANN model was validated using real-time data from a Spanish company. The model was trained with periodic values (weekday and months). This model was compared with similar models using radial basis function neural network and generalized regression neural network and proved to be extremely efficient.

The stability of the SG depends on its ability to provide a constant power supply based on demand. Ahmed and Chen have employed three different machine learning models to predict the long-term and medium-term energy demand in the SG[30]. The models used are ANN with non-linear autoregressive exogenous multivariate inputs model (ANN_NAEMI), Ada Boost and multivariate linear regression (MLR). The study classified the load considering three different intervals, namely 1-month ahead, seasonal forecast and 1-year ahead forecast based on the data about aggregated data consumption. The models not only increased the prediction accuracy but also adequately described the Spatio-temporal use of energy inconsistencies, its variations and future perspective of energy prediction. Ada Boost model, through its preponderance in prediction, outperformed the other models. The variations in the prediction results helped to identify the irregularities in prediction operation.

Instability in electricity prices is one of the socio-economic factors determining the usage of electricity by the consumer and often price impacted the electricity load. Therefore, Shayeghi et al. presented a multi-input multi-output(MIMO) model that correlated the relation between electricity price and its load[31]. The model used the wavelet packet, which was transformed for decomposing the price and load signals into numerous subsets at different frequencies, and generalized mutual information with an objective to select the best features. Also, the model made predictions on electricity load and prices simultaneously using the least-squares SVM (LSSVM) based MIMO model. Furthermore, the Quasi Oppositional Ant Bee Colony algorithm was used for parameter optimization. The simulation results showed that the proposed LSSVM model outperformed ANN as it considered the prediction indices for evaluating the forecasting error.

Khan et al. presented a detailed study of dynamic pricing, load prediction in the SG[32]. The study highlighted the correlation among real-time dynamic pricing of electricity, critical peak pricing and time of use. Two different ways of load forecasting, namely Artificial intelligence (AI) models and computational models, were presented. AI models used ANN, Generalized Regression Neural Network(GRNN), RNN, Auto-Regressive Integrated Moving Average (ARIMA)-SVM, SVM, Wavelet Transformation Error Correction(WTEC)-ANN, Wavelet Transformation(WT)-ANN, Probabilistic Neural Network(PNN), Expert System and Fuzzy logic. It was evident from the survey that AI-based forecasting techniques were found to be more accurate than other statistical models.

Muhammad et al. conducted a survey on Photovoltaic (PV) output forecast[33]. Although the majority of the researchers have attempted to forecast PV output using traditional methods, mathematical models and AI methods, this particular study identified ANN to be capable of generating more accurate forecasting when compared to other conventional and statistical models. The study also revealed the fact that the accuracy of any prediction techniques changed based on the day, seasonal variation, input features and other evaluation matrices.

Muhammad and Abbas conducted a survey on AI-based load forecasting models in the SG[34]. The study highlighted that the performance of the forecast model depended on its architecture, input features, activation function, ML algorithms, which was used for training and generating of forecast errors. It was observed that Back-propagation (BP) training algorithm was commonly used to train NN, but it had numerous associated challenges whereas ANN was best suited for STLF yielding better performance than BP. It could be finally concluded that integrated approaches provided better results.

It is a known fact that Blackout events of the traditional power grid lead to many cascading failures. In order to address this issue in SG, early warning of blackout events becomes a mandatory necessity. Gupta et al. applied the time series model, namely SVM, to predict the blackout events earlier and validated using a 30-bus testbed from IEEE[35]. SVM was trained using a historical database constructed by evaluating the system performance in a steady-state and dynamic state. This database recorded the normal cases and abnormal situation(cascading failure condition).

Pan and Lee performed a comparative analysis of SVM and ANN in the midterm load forecast of the SG[36]. ANN was widely used for this type of forecast, whereas SVM was adopted by the researchers in recent times. The factors affecting the load prediction, which was carried out for the daily power load in one year was analyzed. Mitchell et al. used these models for STLF on different load types, namely batch load, continuous load and batch-continuous load[37]. The results showed that SVM produced the global minimum repeatedly. Both the algorithms performed with extreme inferiority with more than 3% deviation on erratic load and 1.2% deviation on a continuous load.

Electricity demand forecast depends on various factors such as climatic changes, seasonal changes, sea level and catastrophic events by nature. Therefore, demand management in the SG determines its reliability and stability for providing consistent power demands of the consumer. Demand scheduling can be effectively derived by the effective forecast of the consumers' electricity usage pattern. Ali and Azad have used machine learning algorithms, namely Linear regression, SVM and MLP for demand management and load prediction[38] wherein it outperformed the other models. Support vector regression (SVR) employed the constrained quadratic optimization problem, which mapped the input features into high dimensional space using a kernel. SVR outperformed the NN trained with BP algorithm and other linear regression methods. It also produced high-quality outputs with time series missing data. The study thus recommended SVM for load forecasting.

Similar to demand management, SVM performed exceptionally better in other forecasts too. In the prediction of lake water levels, SVM showed compatibility and best results (long-term forecast) when compared with ANN and Statistical model, namely the Seasonal autoregressive model in[39]. SVM was employed in the time series forecast for financial analysis, which helped to overcome two common problems, namely noisy data and non-stationary[40]. The study used fuzzy-based Support vector regression basically to serve the purpose. Li et al. have used PCA and SVM with rough sets for long term electricity load forecast[41]. C-ascending SVM in non-stationary financial time series provided better results with fewer support vectors[42]. Cao and Ju have used dynamic SVM for non-stationary time series forecast[45]. The study modified the traditional SVM with a regularized risk function, which enabled the model to effectively track the structural change in financial time series. Also, SVM with rough sets outperformed the statistical models and GRNN[43]. SVR and chaotic GA simulated annealing algorithm which was integrated to improve the prediction accuracy of Chaotic load[44]. Also, SVM showed better accuracy in the case of MLP for wind speed forecasting[46]. Alazab et al., developed a multi-directional LSTM (MLSTM) model to predict the stability of the SG and the results shown that MLSTM outperformed the traditional LSTM, Deep learning models gated recurrent units and RNN[25]. An overview of the survey describing the problems was addressed. The models used and inferences on their methodology is presented in Table 1.

From the literature survey, it is evident that most of the research on SG was done using deep learning algorithms irrespective of the dataset size. But ML algorithms performed better than the deep learning models being subjected to the SG dataset due to its size. The present work thus emphasizes on the use of ML algorithms on the SG dataset.

3 | PROPOSED MODEL

The work-flow of the proposed model is depicted in Figure 2. The steps involved in the proposed methodology can be summarized as follows:

- SG dataset from UCI ML repository is loaded.
- Preprocessing of the dataset is performed by using min–max method for normalization and label encoding for data transformation.
- The dataset is then split into training and testing data.
- The dataset is trained by various machine learning algorithms.
- The performance of the ML algorithms is then evaluated with several metrics.

TABLE 1 Survey of the literature on SGs.

Problem Definition	**Models used**	**Inferences**
Prediction of electric load in SG[27]	↣ Deep Learning LSTM based RNN model for prediction. ↣Recursive Feature Elimination for feature selection. ↣Extra Trees Regressor as benchmark machine learning model. · ↣Genetic Algorithm for identifying the time lages and layers in LSTM	LSTM RNN may not perform well on different dataset as train split was more.
Demand forecast on individual consumer's household load[28]	↣ANN 3 layered (60-20-1 neuron) ↣ SVM ↣ Regression model	The STMLF was not compared with other model, impact of data during prediction and the price demand relationship was not discussed.
STLF within microgrids[29]	↣Self-Organizing Map ↣K-Means Clustering ↣ Multilayer Perceptron	System was not evaluated with microgrid sized environment. Also number of patterns considered for evaluation against other models was lesser
Predicting long-term and medium-term energy demand in SG at district level[30]	↣ANN-NAEMI ↣ MLR ↣ AdaBoost	Prediction results have enormous errors with large data amd input parameter sets
Correlating the non-linear pattern in the electricity load and price signals[31]	↣ LSSVM-MIMO ↣ ANN ↣ QOABC optimization algorithm ↣ GMI	The model can predict the load and power signals without considering the prior data on particular forecast day but it might not be more accurate
Comprehensive review on load prediction and dynamic pricing of electricity[32]	↣ ANN, WT-ANN,ARIMA-SVM, Regression,WTEC-ANN, Fuzzy logic ↣ Statistical models	AI based forecasting techniques are more accurate than statistical models
Review on AI based load forecasting models[34]	↣ ANN ↣ BP, gradient descent	Integrated approaches can be used for training the NN to attain better forecast results. ANN performs better than BP
Early warning system for prediction of blackout events in SG to avoid cascading failure[35]	↣ SVM	Grid resilience feature of the SG can be determined
Midterm load forecasting[36]	↣ SVM ↣ ANN	NA
Short term Load forecasting[37]	↣ SVM ↣ ANN	SVM attain global minimum repeatedly. But both models shows deviations in forecast on continuous load and erratic loads
Demand and Load forecasting in SG[38]	↣ SVM, SVR ↣ Linear Regression ↣ NN with BP algorithm for training ↣ MLP	SVR proved to be the best choice for the context (even in case of huge data set, with the given data set it was compatible with the other algorithms) whereas MLP have higher computational complexity
Forecasting Lake water Levels[39]	↣ SVM ↣ ANN ↣ Seasonal Autoregressive model	SVM showed yet compatible and better results than other model in long-term
Financial tie series forecast[40]	↣ Fuzzy based SVR	The model provided best results in noise and non -stationarity data.
Short term load forecast[41]	↣ SVM with polynomial basis function and Radial basis function ↣ PCA ↣ Rough set theory	Single kernel SVM function performed weakly
Non-stationary financial time series forecast[42]	↣ C-ascending SVM with regularized risk function	Model utilized less support vectors than the traditional one. Model should be explored with sophisticated weight function.
Exchange Rate Prediction[43]	↣ Linear and non-linear SVM ↣ GA for feature selection	SVM outperforms conventional NN and applied structural risk minimization principle to reduce the generalization error.
Cyclic Electric Load forecasting[44]	↣SVR ↣ Chaotic Genetic Algorithm (Simulated Annealing algorithm)	Outperformed ARIMA and Tensor Flow SVR simulated annealing model

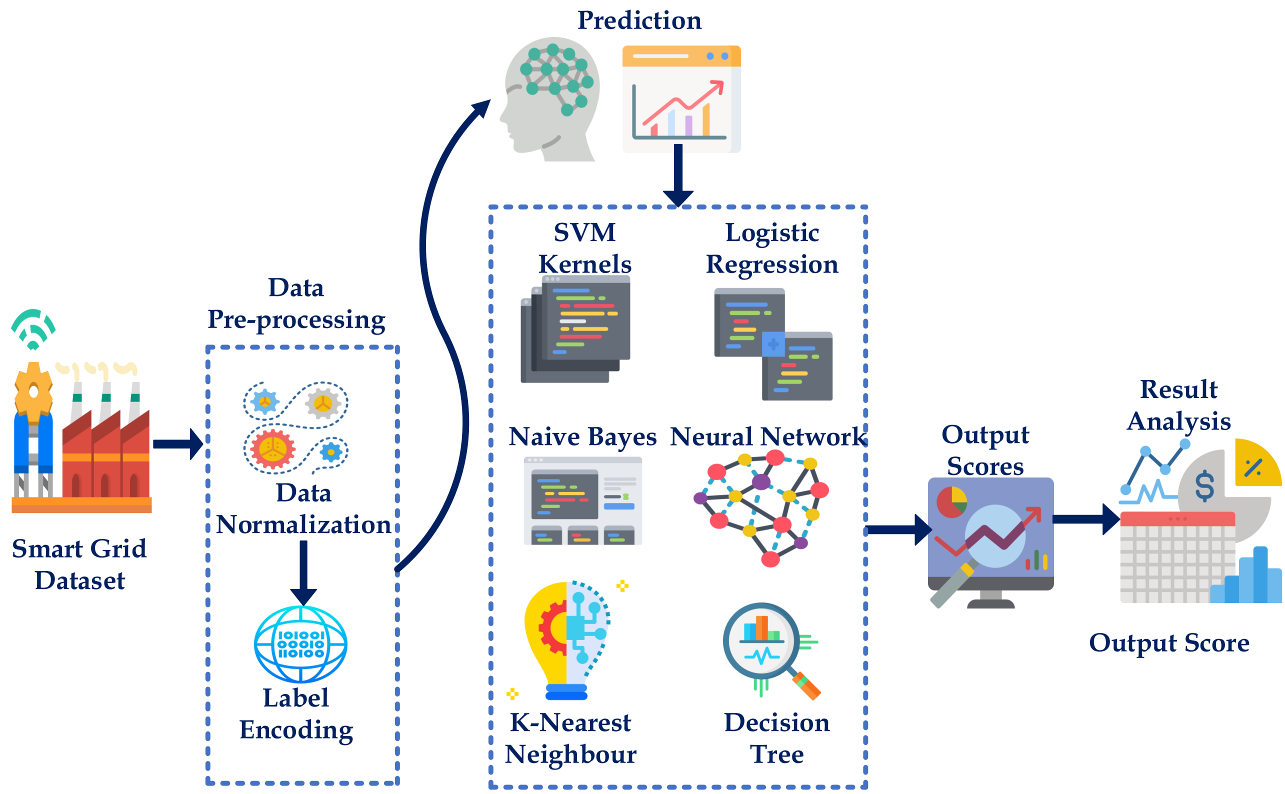


FIGURE 2 Proposed Methodology.

3.1 | Preprocessing

Pre-processing plays an essential role in improving data quality and also the performance of the ML algorithms[47,48]. The two common pre-processing techniques used in any machine learning model are normalization and data transformation. The data in an SG dataset are scattered with different ranges which often lead to bias towards values having higher weights, thereby degrading the performance of the proposed model. In order to avoid this, min-max normalization is used in the present study to normalize the SG dataset. The min-max normalization fits the data into a common scale, which improves the performance of the classifiers. redMachines use mathematical formulae to process the data and hence requires data to be in numeric format. Since most of the dataset contains both numerical and categorical values, data encoding is done during data pre-processing[49], which converts the non-numeric values to numeric ones before being fed to the ML models.

The pre-processed data is then split into training and testing datasets. The ML algorithms are trained by the training dataset and then the trained algorithms are tested with new data set to evaluate its performance. In this work 70% of the dataset is used to train the ML algorithms and the remaining 30% of the dataset is used for evaluating the performance of the trained ML algorithms.

Due to the minimal size of the dataset, several ML algorithms are used for the purpose of classification instead of Deep Learning based algorithms. Some of the popular ML algorithms, namely SVM, Logistic Regression, Naive Bayes, Neural Networks and Decision Tree algorithms, are used in this work to classify the SG dataset. The performance of the ML algorithms are then evaluated using metrics - precision, recall, F1-score, Receiver Operating Characteristic and accuracy. The results obtained are then compared with the recent works on the SG datasets. The ML algorithms used in this work are discussed in the below subsections.

3.2 | Support vector machine

SVM aims to solve classification and regression problems and most researchers prefer to use SVM due to its capability to with high accuracy using minimal computing power. The versatility of SVM lies in the kernelization process n which it uses the kernel trick to model nonlinear decision boundaries.

SVM aims to classify points using hyperplanes and ensures that after developing hyperplanes, two margin lines are produced creating classification points that are linearly separable[50]. These margin lines are created to move one of the margin lines to the nearest positive point and another margin line to the nearest negative point as shown in Figure 3. The distance between the two parallel margins is referred to as the marginal distance. The primary aim of this approach is to maximize the marginal distance by selecting the best hyperplane.

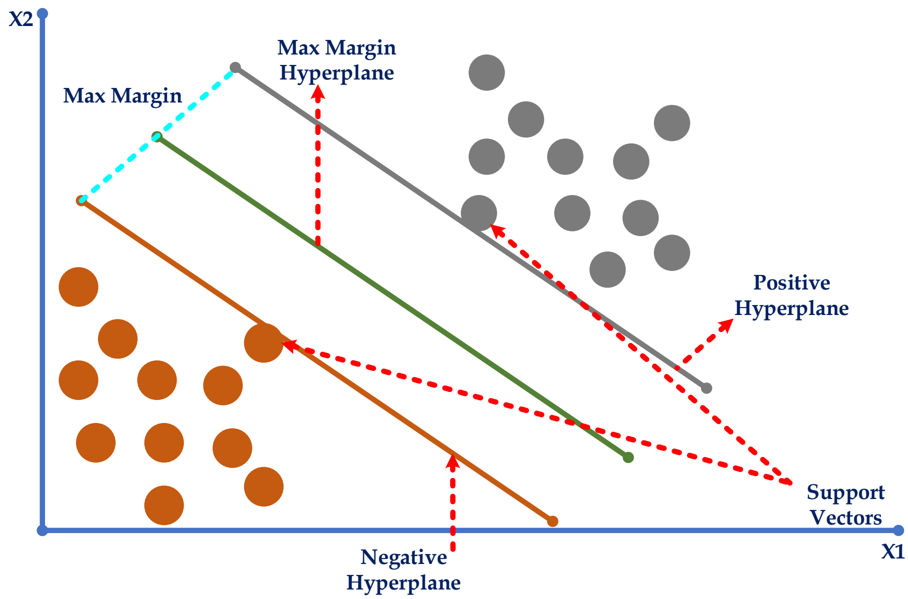


FIGURE 3 Maximum marginal hyperplane with support vectors.

Support Vectors: The vector points extremely closer to the hyperplane are classified as support vector points, and these two data points specifically contribute to the results of the algorithm whereas other data points have no significant impact. Also, it is important to highlight that the removal of support vectors alters the hyperplane's position.

Good Margin: The support vector points of the positive class and the negative class maintain the maximum distance to the hyperplane.

Bad Margin: This refers to the hyperplane that is close to either positive support class vectors or negative support class vectors.

Hard Margin: This refers to data points that can be separated from the positive class and the negative class by maintaining the maximum marginal distance between the parallel hyperplane.

Soft Margin These are data points that can not be separated from the positive class and the negative class by drawing a hyperplane. Soft margins are very difficult to manage as positive and negative data points are intermixed wherein the accuracy efficiency is diminished. The line equation is given in Equation 1.

$$b = xa + y \tag{1}$$

The two vectors are v, a.

$$v\begin{pmatrix}-y\\-x\\1\end{pmatrix} \text{ and } a\begin{pmatrix}1\\a\\b\end{pmatrix}$$

$$\begin{gathered} v^T a = -y \times (1) + (-x) \times a + 1 \times b \\ v^T a = b - xa - y \end{gathered} \tag{2}$$

The hyperplane equation is given in Equation 3.

$$v^T a + y = 0 \tag{3}$$

Where, y is bias and v, a is vector.

Consider two support vectors, a^+ (Positive Support Vector) and a^- (Negative Support Vector). The distance between the two margin lines should be the maximum for scattering the data points linearly. Maximum margin helps to minimize loss. The equation for the approximation of the margin is shown in Equation 4.

$$\left(a^+ - a^-\right) \cdot v = \left(a^+ - a^-\right) \cdot \frac{v}{\|v\|} = a^+ \cdot \frac{v}{\|v\|} - a^- \cdot \frac{v}{\|v\|} \tag{4}$$

Cost Function and Gradient Updates SVM's aim is to maintain maximum margin distance between data points and hyperplane. Hinge loss function allows maximum margin between data points and hyperplane which is given in Equation 5.

$$M(v) = \sum_{i=1} \max\left(0, 1 - b_i\left[v^T a_i + y\right]\right) + \lambda\|v\|_2^2 \tag{5}$$

$\sum_{i=1} \max\left(0, 1 - b_i\left[v^T a_i + y\right]\right)$ is used to minimize misclassification,

$\lambda\|v\|_2^2$ is regularization used to avoid over-fitting. λ is known as the regularization factor to maximize the marginal difference and holds a_i on the appropriate margin side. The difference between positive and negative hyperplane is measured using Hinge loss as given in Equation 6.

$$\begin{gathered} \max\left(0, 1 - b_i\left[v^T a_i + y\right]\right) = 0 \\ \Rightarrow b_i\left[v^T a_i + y\right] = 1 \end{gathered} \tag{6}$$

$$b_i \begin{cases} +1 & va + y \geq 1 \\ -1 & va + y \leq -1 \end{cases}$$

Equation 6 is further derived as in Equation 7.

$$a^+ \cdot \frac{v}{\|v\|} - a^- \cdot \frac{v}{\|v\|} = \frac{1-y}{\|v\|} - \frac{-y-1}{\|v\|} = \frac{2}{\|v\|} \tag{7}$$

Finally, the objective function is given in Equation 8.

$$\max \frac{2}{\|v\|} \rightarrow \max \frac{1}{\|v\|} \rightarrow \min \|v\| \rightarrow \min \frac{1}{2}\|v\|^2 \tag{8}$$

Polynomial kernel: The polynomial kernel is a kernel functions used along with SVM to represent similar data points in a dataset[51]. The polynomial kernel for degree-d is delineated as in Equation 9.

$$F(a, b) = \left(a^T b + z\right)^s \tag{9}$$

where a and b are vectors of the data set, $z \geq 0$ is a weight vector comparing higher-order and lower-order polynomial values. As a kernel, F correlates to an internal data points in a higher dimensional space that is based on a certain mapping φ as shown in Equation 10.

$$F(a, b) = \langle \varphi(a), \varphi(b) \rangle \tag{10}$$

The influence of φ is shown below in Equation 11, where $s = 2$.

$$\begin{gathered} F(a, b) = \left(\sum_{i=1}^{m} a_i b_i - z\right)^2 = \sum_{i=1}^{m} \left(a_i^2\right)\left(b_i^2\right) + \sum_{i=2}^{m}\sum_{j=1}^{i-1}\left(\sqrt{2}a_i a_j\right)\left(\sqrt{2}b_i b_j\right); \\ + \sum_{i=1}^{m}\left(\sqrt{2z}a_i\right)\left(\sqrt{2z}b_i\right) + z^2 \end{gathered} \tag{11}$$

$$\varphi(a) = \left\langle a_m^2, \ldots, a_1^2, \sqrt{2}a_m a_{m-1}, \ldots, \sqrt{2}a_m a_1, \sqrt{2}a_{m-1}a_{m-2}\right.$$

$$, \ldots, \sqrt{2}a_{m-1}a_1, \ldots, \sqrt{2}a_2a_1, \sqrt{2z}a_m, \ldots, \sqrt{2z}a_1, z\Big\rangle \tag{12}$$

Radial basis function kernel(RBF): RBF Kernel is a common kernel function that is predominantly used in numerous kernelized machine learning algorithms. RBF kernel is widely used for SVM classification tasks. The RBF kernel for two separate sample vectors a and $a^{a'}$, in decision boundary, is shown in Equation 13.

$$F\left(a, a'\right) = \exp\left(-\frac{\|a - a'\|^2}{2\rho^2}\right) \tag{13}$$

$\|a - a'\|^2$ is considered to be the Euclidean distance between two vectors and ρ is considered as a variable where $\xi = \frac{1}{2\rho^2}$.

$$F\left(a, a'\right) = \exp\left(-\xi \left\|a - a'\right\|^2\right) \tag{14}$$

The RBF kernel value decreases with respect to the euclidean distance ranging from 0 to ∞. When $\rho = 1$ the equation is expressed as in Equation **??**:

$$\exp\left(-\frac{1}{2}\left\|a - a'\right\|^2\right) = \sum_{j=0}^{\infty} \frac{\left(a^{\top}a'\right)^j}{j!} \exp\left(-\frac{1}{2}\|a\|^2\right) \exp\left(-\frac{1}{2}\left\|a'\right\|^2\right);$$

$$= \sum_{j=0}^{\infty} \sum_{\sum m_i = j} \exp\left(-\frac{1}{2}\|a\|^2\right) \frac{a_1^{m_1} \cdots a_f^{m_f}}{\sqrt{m_1! \cdots m_f!}} \exp\left(-\frac{1}{2}\left\|a'\right\|^2\right) \frac{a_1'^{m_1} \cdots a_f'^{m_f}}{\sqrt{m_1! \cdots m_f!}} \tag{15}$$

Sigmoid Kernel Sigmoid Kernel is often referred to as Hyperbolic Tangent Kernel which is originated from the neural network research area. In most cases, sigmoid function has been used as an activation function for neural networks. The mathematical representation of the sigmoid function is shown in the Equation 16.

$$F(a, b) = \tanh\left(\alpha a^T b + z\right) \tag{16}$$

where α is chosen to be a slope, z is constant, α is considered as $\frac{1}{M}$ where M is data dimension.

3.3 | Logistic Regression

Logistic Regression is one of the most popular machine learning algorithms[52]. The goal of the algorithm is to seek a similarity between both the likeliness of desired outcome and attributes. Logistic Regression Equation is as follows:

$$\log\left(\frac{p(a)}{1 - p(a)}\right) = \zeta_l + A_{1a} \tag{17}$$

$$p(a) = \frac{e^{X_0 + \zeta_{1a}}}{1 + k^{\zeta_0 + \zeta_{1a}}} \tag{18}$$

Equation 18 is considered as a sigmoid function, generating an S-like curve. The probability value is in the range of 0<p<1.

3.4 | Naive Bayes

Naive Bayes (NB)[53] is a classification technique that assumes independence between predictors. NB consists of two parts namely Naive and Bayes. The NB classifier assumes that the presence of a particular feature in a class is independent of the presence of any other feature. All the features independently contribute to the probability of a variable belonging to the specific target class or otherwise. NB is quite trivial to develop and is particularly useful for very large dataset. In probability theory and statistics, this is alternatively known as Bayes Law. The conditional probability of NB is shown in the Equation 19.

$$p\left(Z_f|a\right) = \frac{p\left(Z_f\right) p\left(a|Z_f\right)}{p(a)} \tag{19}$$

The Bayes classifier of the probability model is shown in the Equation 20.

$$\hat{b} = \underset{f \in \{1, \ldots, F\}}{\operatorname{argmax}}\, p\left(Z_f\right) \prod_{i=1}^{m} p\left(a_i|Z_f\right) \tag{20}$$

3.5 | K-Nearest Neighbour(k-NN)

k-NN is one of the simplest ways to classify data and is primarily used for regression and classification. It is used to identify data points using the closest training examples in the feature space. It involves instance-based learning where the function is locally approximated and all performance is deferred until classification. The F value is considered as the number of closest neighbors in a vector classification and the selection of the most appropriate F value is essential for attaining superior quality results. In the proposed approach, we considered F=5, leaf-size=30 and Minkowski metric weights are uniform. Equations for k-NN are as follows:

$$\text{Eculidean equation } = \sqrt{\sum_{i=1}^{F}(Ai - bi)^2} \tag{21}$$

$$\text{Manhattan equation} = \sum_{i=1}^{F} |Ai - Bi| \tag{22}$$

$$\text{Minkowski} = \left(\sum_{i=1}^{F}(|A - Bi|)^r\right)^{1/r} \tag{23}$$

3.6 | Decision Tree (DT)

DT belongs to one of the few classification models where we can understand the exact reasoning behind the classifier, making a particular decision [54]. DT provides a graphical representation of all possible decision solutions based on certain conditions. It begins with a root and then branches off to a number of possible solutions, just like a tree. The root node initially gets added to the tree, which receives the trained data set, and then each node asks a true and false question to one of the features. Henceforth, the dataset gets divided into two different subsets. These subsets then act as an input to the child node. The aim is to produce the purest possible distribution of the labels at each node. This iteration process continues till no further questions are asked, and it finally reaches the leaf. The equation for Entropy is given in Equation 24.

$$\text{E: } I\left(p_1, p_2, \dots p_n\right) = \sum_{i=1}^{n}\left(p_i \log\left(1/p_i\right)\right) \tag{24}$$

$\left(p_1, p_2, \dots p_n\right)$ denotes the probabilities of class labels.

In the proposed model the splitting was performed using Gini-index, max-depth = none, min-samples-split=2, min-samples-leaf=1, class-weights = none, random-state = none, min-impurity-decrease=0.0 and min-impurity-split = none.

3.7 | Neural Networks

The neural network maps the input units to its appropriate output unit after performing some mathematical calculations [55,56]. The neural network consists of input layer where features are given as input, hidden layers are placed between input and output layers. The role of the hidden layer is to multiply weights to the input layer and then pass the resultant values to activation functions. The predictions are performed at the last layer, called as an output layer.

$$H_n = \phi_1 + \left(\sum_{i=1}^{k} W_{mn} + \theta_n\right) \tag{25}$$

Equation 25 shows the working of hidden layer when features f_i are given as input. Weight between nth input and m^{th} hidden layer is calculated by W_{mn} where θ_n is the value of bias factor.

$$\text{output} = \phi_2 + \left(\sum_{i=1}^{k} W_{MO} + \theta_O\right) \tag{26}$$

The mapping inputs to the outputs is an iterative operation, in which weights W_{mn} are changed in each iteration. Back propagation (BP) algorithm is one of the widely used algorithms and the equation for back propagation is given below:

$$W_{nm}(t+1) = W_{nm}(t) - e\frac{\partial Ef}{W_{nm}} \tag{27}$$

The error between the calculated and target output is used for weighing updates.

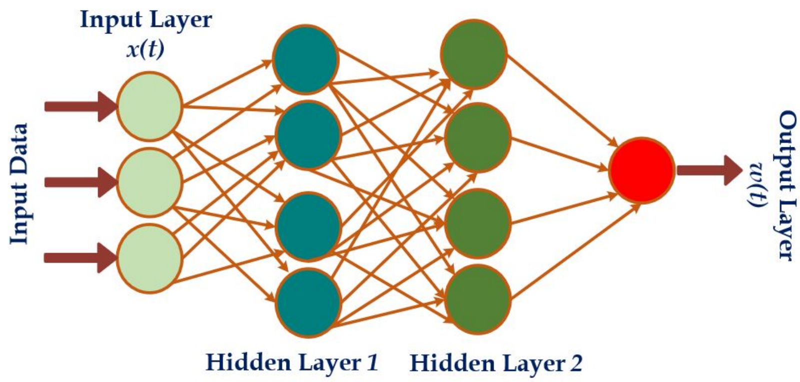


FIGURE 4 Neural Network Architecture.

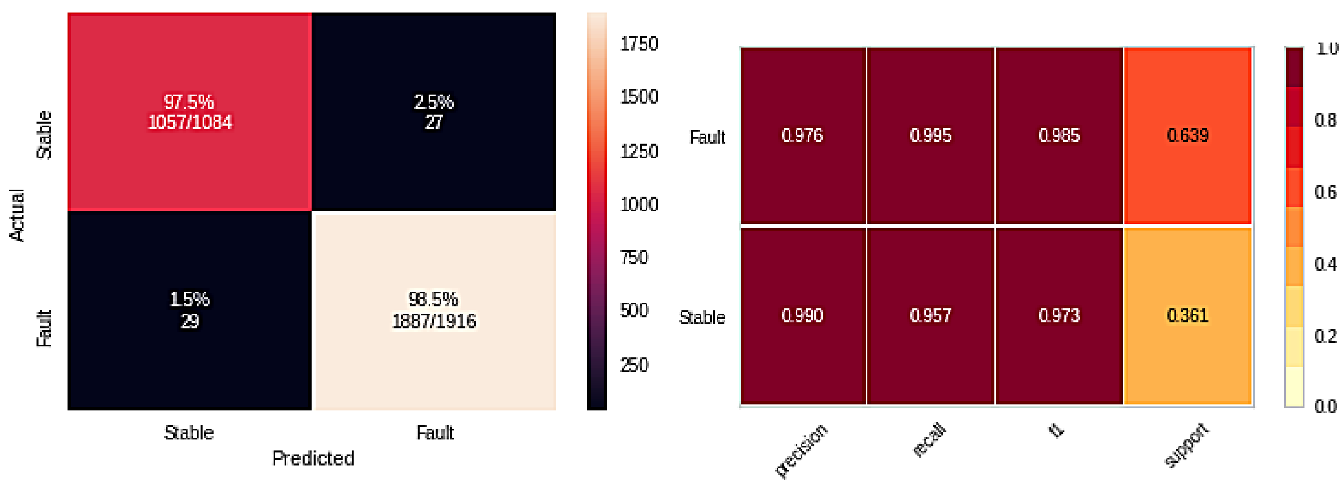
4 | RESULTS AND DISCUSSION

In this work, the experimentation is carried out using "Google Colab", Google's online Graphical Processing Unit (GPU). Python 3.7 is used as the programming language. The dataset used for the experimentation is collected from publicly available UCI machine learning repository [57] which consists of 10000 instances with 14 attributes. The attributes in the dataset are electricity producer values, nominal power consumed/produced, coefficient related to price elasticity, the maximum value of the equation root and the stability of the system (class label, whether the system is stable or not).

The performance metrics considered in the study are accuracy, recall, F1 measurement, and detection rate (DR) which helps to evaluate the proposed solution. The above performance measurements are based on True Positive (TP), False Positive (FP), False Negative (FN) and True Negative (TN).

In the present study, 70% of the SG dataset is used for training and remaining 30% of the SG dataset is used for testing and validation purposes. The machine learning models are evaluated based on the parameters mentioned above in association with the Receiver Operating Characteristic (ROC) curve which helps in justifying the results.

4.1 | Neural Network Results



((a)) Confusion matrix

((b)) Classification Report

FIGURE 5 Neural Network Results

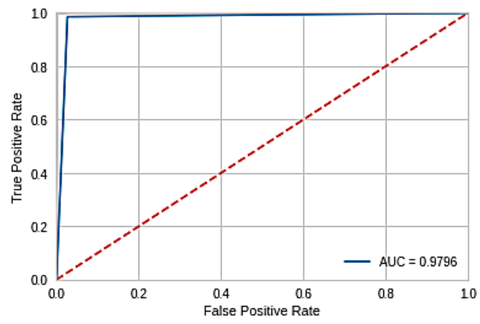
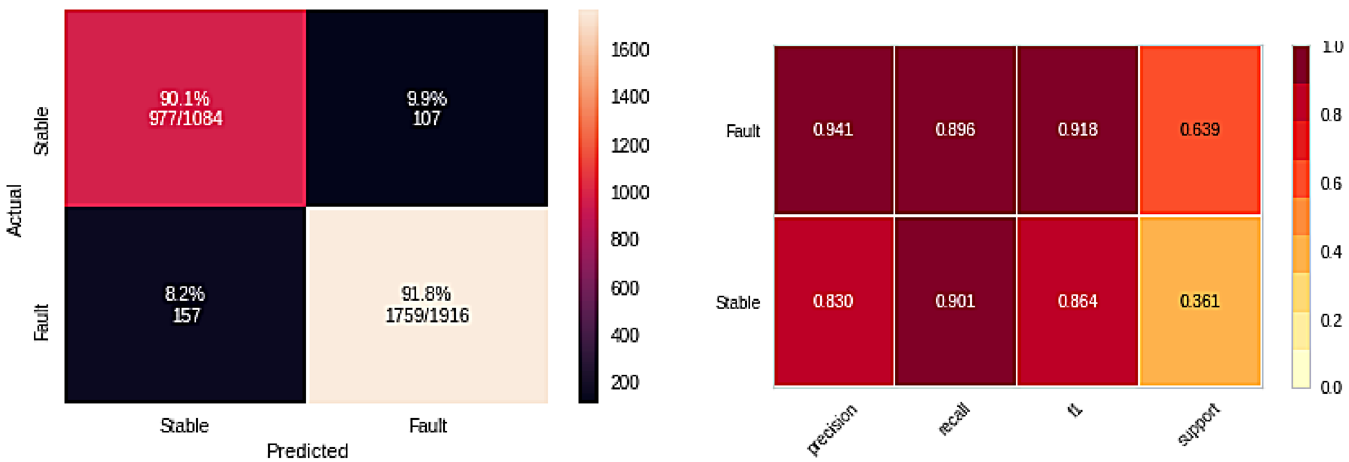


FIGURE 6 NN Roc Curve

Figure 5 represents the confusion metric (CM) and classification report (CR) for neural network classifier. Using neural network classifier we achieved an accuracy of 97.50%, for stable class and 98.50% accuracy for fault class. Also, 2.5% false positive rate (FPR) and 1.5% false negative rate (FNR) is achieved for stable and fault class respectively, depicted in figure 5.a. The result highlight the fact that 97.60% precision, 99.50% recall and 98.50% F1-Measure is achieved for fault class. Similarly for stable class precision, recall and F1-Measure scores are 99.00%, 95.70% and 97.30% respectively as represented in figure 5.b. Figure 6 represents the ROC curve for neural network wherein the area under the curve score is 97.96%.

4.2 | Support vector Machine Results

4.2.1 | SVM Polynomial Kernel



((a)) Confusion matrix ((b)) Classification Report

FIGURE 7 SVM Polynomial kernel Results

Figure 7 presents the SVM Polynomial Kernel results. As shown in the figure, 91.20% average predication accuracy is achieved in case of SVM Polynomial Kernel wherein 977 records are classified accurately as a stable class achieving an accuracy of 90.10% . On the other hand, 1759 out of 1916 records are detected as fault class with the accuracy of 91.80% using SVM with Polynomial kernel as shown in figure 7.a. In figure 7.b SVM polynomial kernel precision score for fault and stable grids are 94.10% and 83.00% respectively. Similarly recall for both fault and stable classes are 89.60% and 90.10% respectively. F1-Measure for fault class is 97.80% and for stable class F1-Measure score is 86.40% respectively. The Area under the curve score is 90.21% as shown in figure 8.

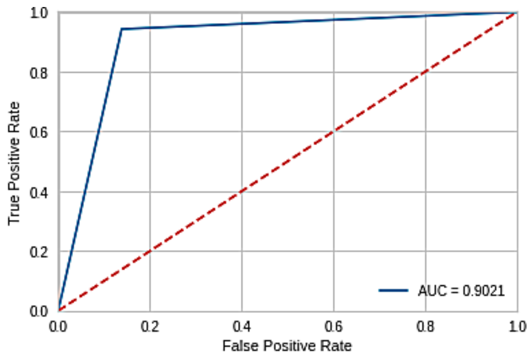
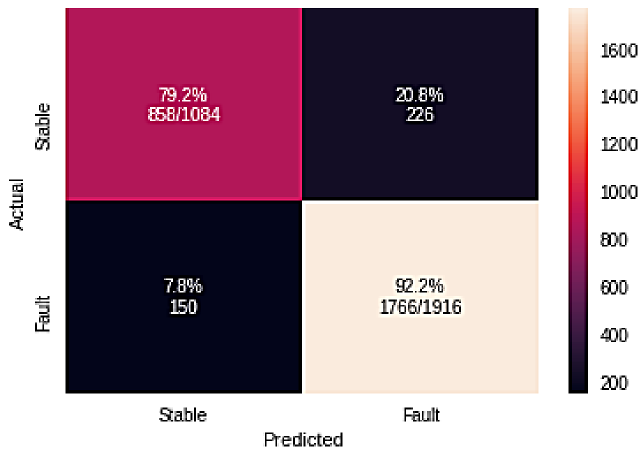
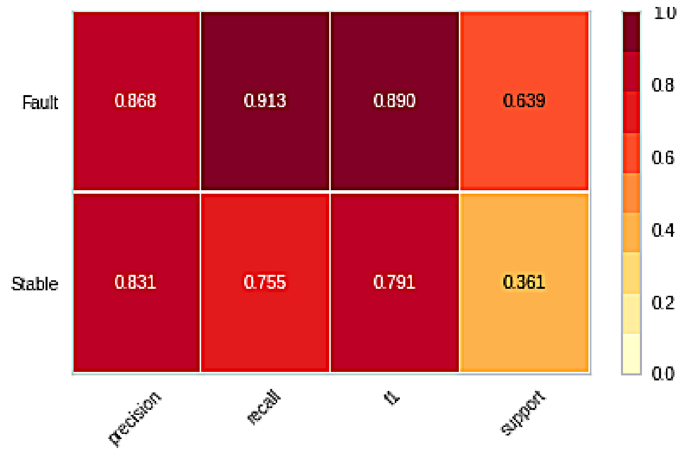


FIGURE 8 SVM Polynomial kernel ROC Curve

4.2.2 | SVM RBF Kernel



((a)) Confusion matrix



((b)) Classification Report

FIGURE 9 SVM RBF kernel Results

Figure 9 presents the SVM RBF Kernel results wherein 87.46% average predication accuracy is achieved in case of the SVM RBF Kernel. A total number of 858 records are classified accurately as stable class generating an accuracy of 79.20% , while 1766 out of 1916 records are detected as fault class yielding an accuracy of 92.20% when SVM with RBF kernel is deployed as shown in figure 9.a. In figure 9.b SVM RBF kernel precision score for fault and stable grids are 86.80% and 83.10% respectively. Similarly recall for both fault and stable classes are 91.30% and 75.50% respectively. F1-Measure for the fault class is 89.00% and for the stable class, the F1-Measure score is 79.10% respectively. The Area under the curve score is 86.89% as shown in figure 10.

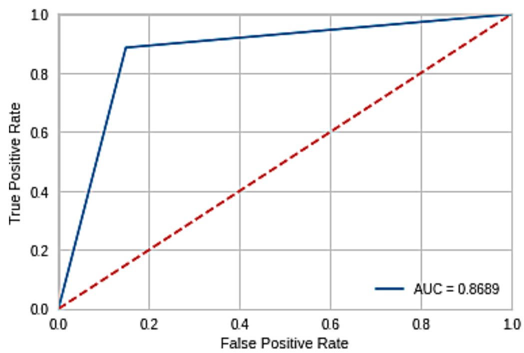
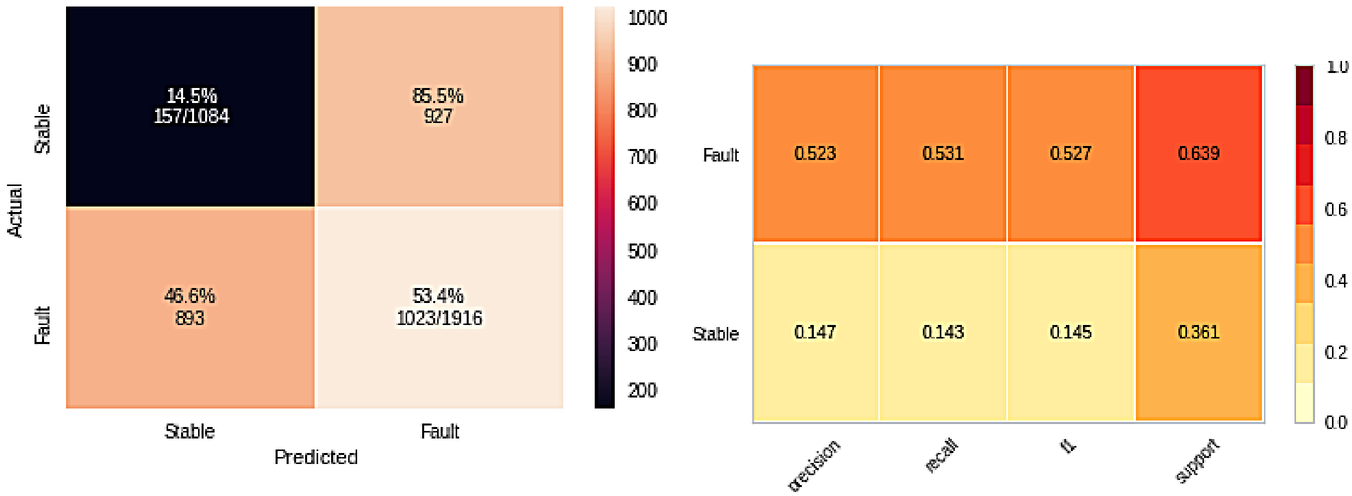


FIGURE 10 SVM RBF kernel ROC Curve

4.2.3 | SVM Sigmoid Kernel



((a)) Confusion matrix **((b))** Classification Report

FIGURE 11 SVM Sigmoid kernel Results

It can be observed from figure 11.a that 157 records are correctly classified as stable and 1023 records are predicated as fault class when SVM sigmoid kernel is implemented. Similarly figure 11.b depicts the classification report for SVM sigmoid kernel. Precision, recall and F1-Measure scores for fault class are 52.30%, 53.10% and 52.70% respectively. Similarly precision for stable class is 14.70%, recall score is 14.30% and F1-Measure score is 14.50% respectively. Figure 12 represents the area under the curve (AUC) for SVM sigmoid kernel and AUC score for SVM sigmoid kernel is 33.71%.

4.3 | Decision Tree

A total of 1084 records are classified correctly as stable, with an accuracy of 100%, while 1915 records are detected as fault with an accuracy of 99.90%, respectively using decision tree as depicted in figure 13.a. The precision, recall and F1-Measure scores are 100%, 99.90%,100% respectively for fault class. Similarly for stable class precision, recall and F1-Measure scores are 99.90%, 100%,100% respectively as depicted in figure 13.b. The Figure 14 presents the decision tree AUC score of 99.95%.

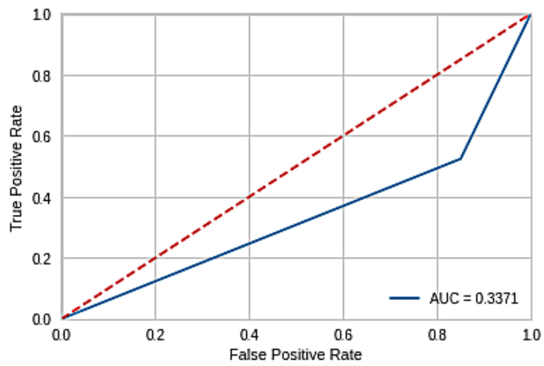
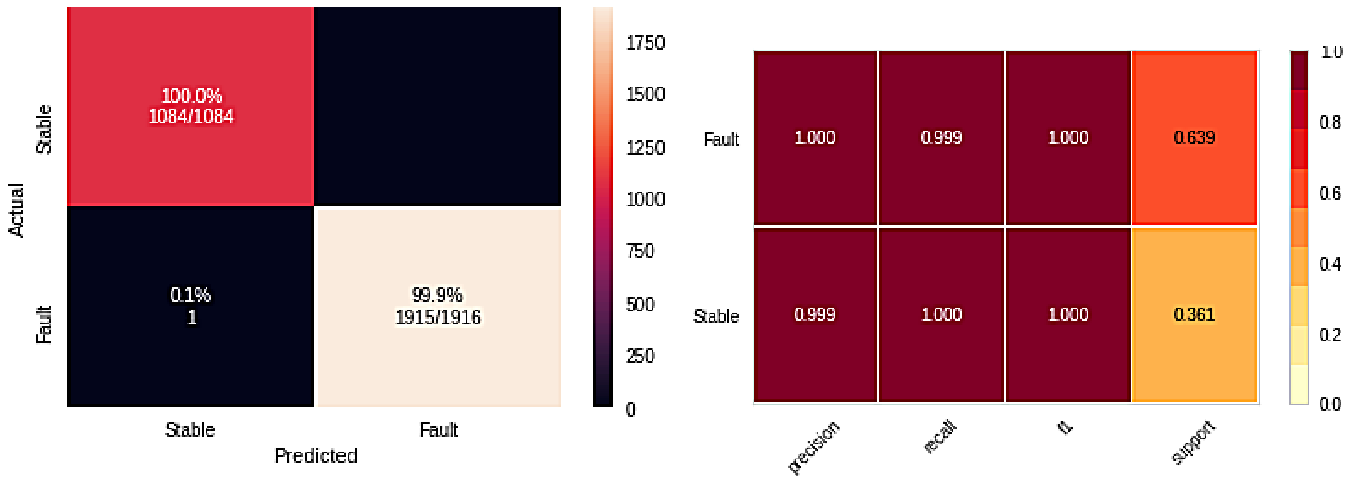


FIGURE 12 SVM Sigmoid kernel ROC Curve



((**a**)) Confusion matrix

((**b**)) Classification Report

FIGURE 13 Decision Tree Results

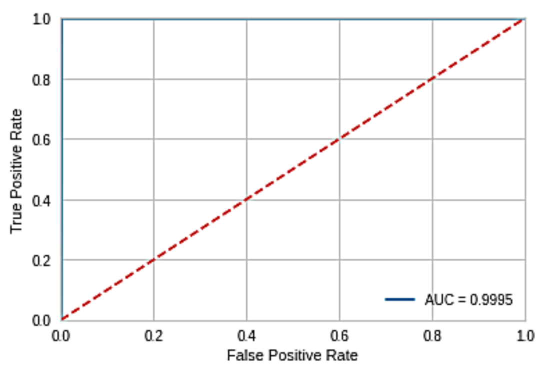


FIGURE 14 Decision Tree ROC Curve

4.4 | K Nearest Neighbour (KNN) Results

Figure 16.a represents the confusion matrix while figure 16.b represents classification report for KNN. KNN classifier detects 701 stable class instances correctly with an accuracy of 64.70%. Similarly, the fault grid prediction accuracy for KNN is observed to be 85.60% which refers 1641 instances to be detected accurately out of 1916. The results show that 81.10% precision, 85.60%

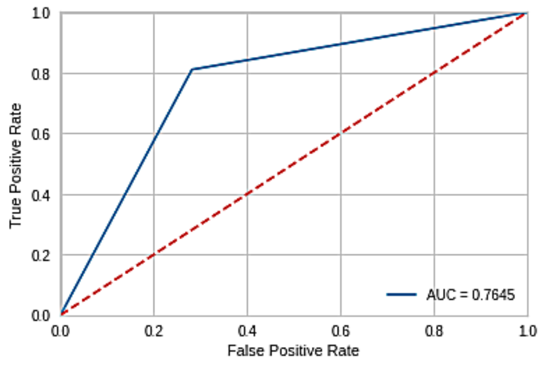
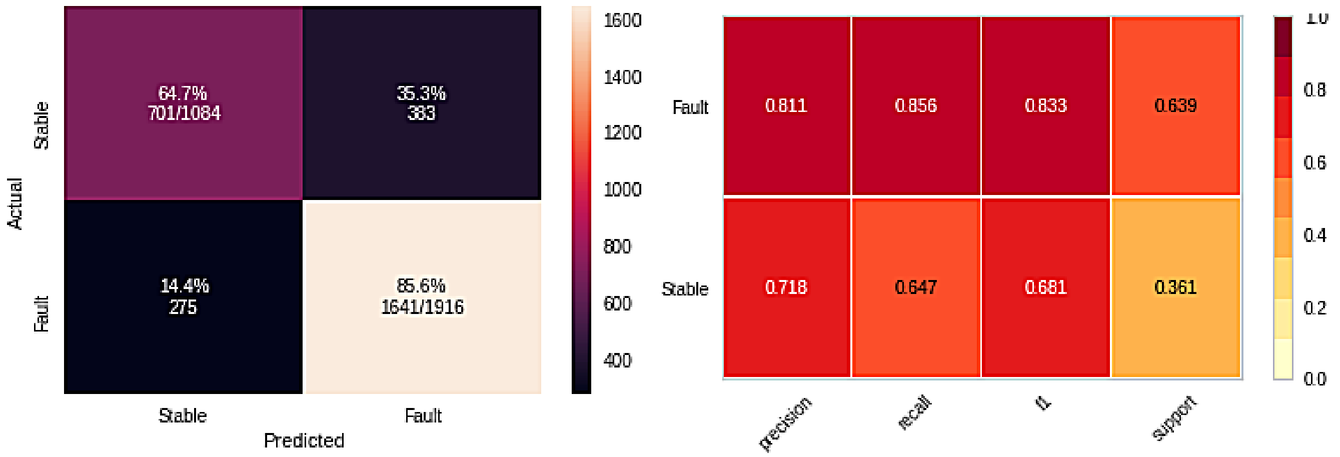


FIGURE 15 KNN ROC Curve



((a)) Confusion matrix

((b)) Classification Report

FIGURE 16 KNN Results

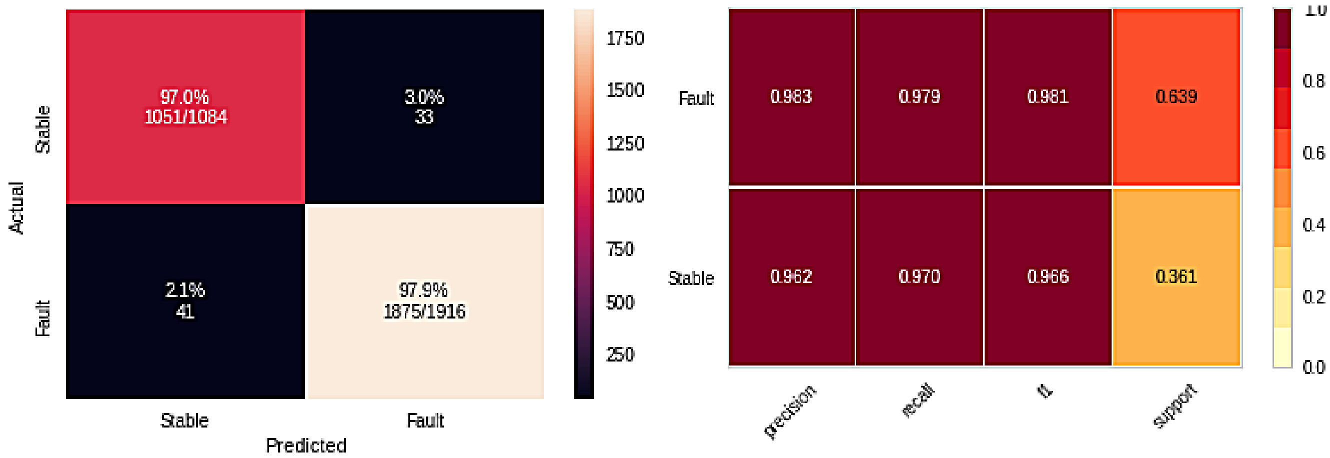
recall and 83.30% F1-Measure score is achieved in case of fault class. For the stable class, precision, recall and F1-Measure scores are 71.80%, 64.74%, 68.10% respectively. Figure 15 represents the ROC curve of KNN classifier and AUC score for KNN classifier which is 76.45%.

4.5 | Naive Bayes Results

A total of 1051 records are classified correctly as stable with an accuracy of 97%, while 1875 records are detected as fault with an accuracy of 97.90%, respectively using naive bayes as depicted in figure 17.a. Precision, recall and F1-Measure scores are 98.30%, 97.90%,98.10% respectively for fault class. Similarly for stable class, precision, recall and F1-Measure scores are 96.20%, 97%,96.60% respectively depicted in figure 17.b. AUC for naive bayes is 97.26% as depicted in figure 18.

4.6 | Logistic Regression Results

It is observed that 877 records are classified correctly as stable with an accuracy of 80.90%, while 1770 records are detected as fault with an accuracy of 92.40%, respectively using logistic regression as depicted in figure 19.a. Precision, recall and F1-Measure scores are 89.50%, 92.40, 90.90% respectively for the fault class. Similarly for stable class, precision, recall and F1-Measure scores are 85.70%, 80.90%,83.20% respectively as depicted in figure 17.b. AUC for logistic regression is 87.63% which is represented in figure 20.



((a)) Confusion matrix ((b)) Classification Report

FIGURE 17 Naive Bayes Results

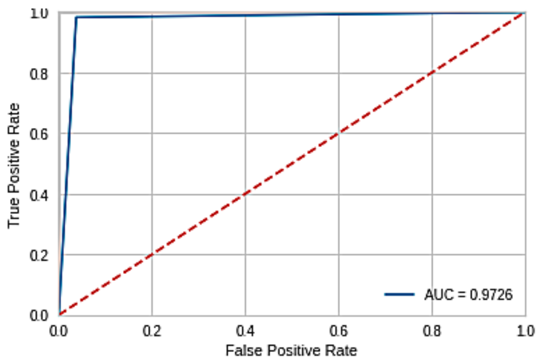
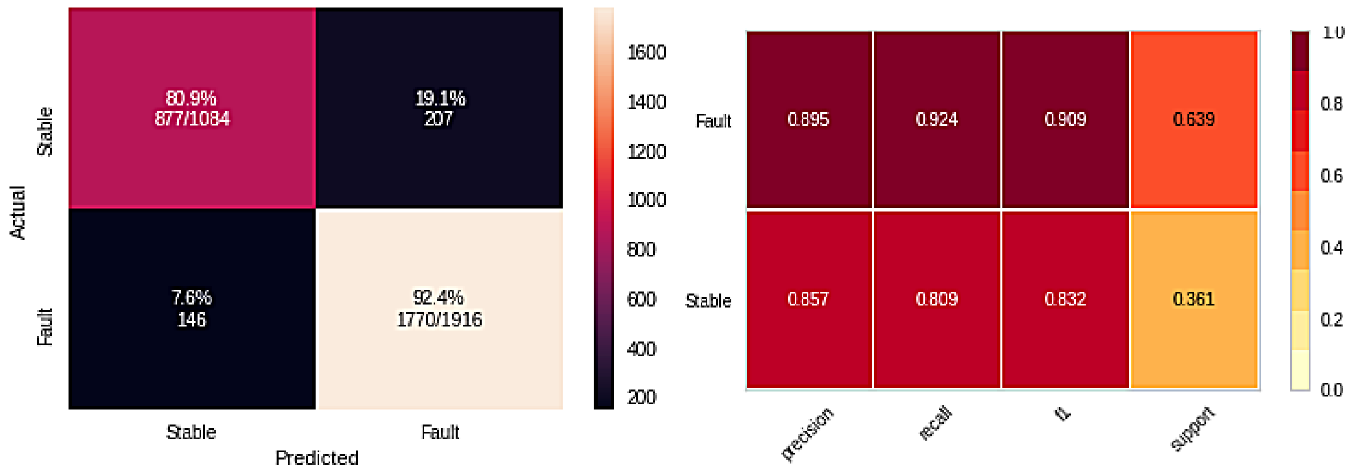


FIGURE 18 Naive Bayes ROC Curve



((a)) Confusion matrix ((b)) Classification Report

FIGURE 19 Naive Bayes Results

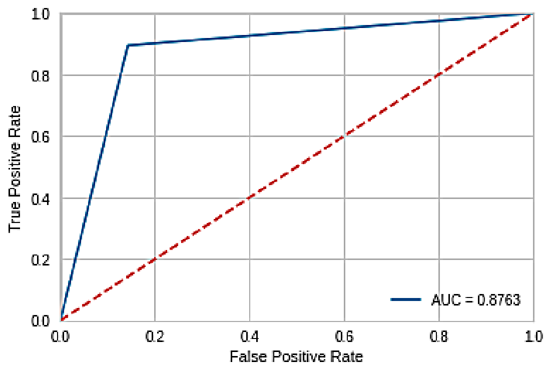


FIGURE 20 Logistic Regression ROC Curve

TABLE 2 Testing Accuracy Comparison of ML Algorithms.

Classifier	**Testing Accuracy**
SVM(Polynomial Kernel)	91.20
SVM (RBF Kernel)	87.46
SVM (Sigmoid Kernel)	39.33
KNN	78.06
Logistic Regression	88.23
Naive Bayes	97.53
Decision Tree	99.96
Neural Network	98.13

TABLE 3 Comparison With Existing Works.

Method	Accuracy
MLSTM [25]	99.07
AdaBoost [58]	97.87
KNN [59]	95.90
Proposed Model	99.96

The predication accuracy for the decision tree model is 99.96% which is highest in comparison to other classifiers used in this work as depicted in Table 2. Since Decision tree is a probability based algorithm, it outperforms other algorithms used in this work in terms of prediction accuracy, precision, recall and F1-Measure respectively.

The authors in [25] achieved 99.07% accuracy, 97% precision, 100% recall and 99.00% F1-Measure for the stable class. Similarly for unstable class they achieved 100% precision, and 99.00% recall and F1-Measure respectively. In this research, decision tree achieved 99.96% predication accuracy, 100% precision and F1-Measure respectively for fault class and 99.90% recall. For stable class, decision tree classifier achieved 99.90% precision, 100% recall and 100% F1-Measure score respectively. In [25] authors also achieved 99.27% AUC using multidimensional long short term memory while decision tree classifier achieved 99.95% AUC. Table 3 depicts the comparison of the current work with recent works.

From the results obtained the following can be concluded:

- Since the size of the dataset is not huge, ML algorithms are apt for classification of the SG dataset when compared to deep learning models.
- DT classifier outperforms the other ML algorithms considered as the number of attributes are relatively less.

5 | CONCLUSION AND FUTURE SCOPE

Machine learning algorithms play a vital role in maintaining the stability of SG due to its ability of predicting the electricity demands of the customers. With the emergence of various machine learning algorithms, the ultimate challenge is to find the most appropriate algorithm to predict the stability of the SG. In order to achieve this, a comprehensive survey on the state-of-the-art machine learning algorithms have been performed in order to predict the stability of SGs. The dataset used in this work is collected from the publicly available UCI machine learning repository. The experimental results proved that the decision tree classification algorithm outperforms SVM, KNN, Naïve Bayes, Logistic Regression and Neural Network. The limitations of the current work indicate that the size of the dataset is not quite huge. But in real-time, SGs generate massive data. In order to address this problem, as part of future work, effective feature engineering-based models could be applied on real-time SG data.

Conflict of interest

The authors declare no potential conflict of interests.

Acknowledgement

Researchers Supporting Project number (RSP-2020/250), King Saud University, Riyadh, Saudi Arabia

References

1. Zhao B, Zeng L, Li B, et al. Collaborative control of thermostatically controlled appliances for balancing renewable generation in smart grid. *IEEJ Transactions on Electrical and Electronic Engineering* 2020; 15(3): 460–468.

2. Gungor VC, Sahin D, Kocak T, et al. Smart grid technologies: Communication technologies and standards. *IEEE transactions on Industrial informatics* 2011; 7(4): 529–539.

3. Alazab M, Huda S, Abawajy J, et al. A hybrid wrapper-filter approach for malware detection. *Journal of networks* 2014; 9(11): 2878–2891.

4. Alazab M, Layton R, Broadhurst R, Bouhours B. Malicious spam emails developments and authorship attribution. In: IEEE. ; 2013: 58–68.

5. Alazab M, Broadhurst R. An analysis of the nature of spam as cybercrime. In: Springer. 2017 (pp. 251–266).

6. Desai SK, Dua A, Kumar N, Das AK, Rodrigues JJ. Demand Response Management Using Lattice-Based Cryptography in Smart Grids. In: IEEE. ; 2018: 1–6.

7. Deepa N, Pham QV, Nguyen DC, et al. A Survey on Blockchain for Big Data: Approaches, Opportunities, and Future Directions. *arXiv preprint arXiv:2009.00858* 2020.

8. Numan M, Subhan F, Khan WZ, et al. A systematic review on clone node detection in static wireless sensor networks. *IEEE Access* 2020; 8: 65450–65461.

9. Dileep G. A survey on smart grid technologies and applications. *Renew. Energy* 2020; 146: 2589–2625.

10. Jindal A, Aujla GS, Kumar N. SURVIVOR: A blockchain based edge-as-a-service framework for secure energy trading in SDN-enabled vehicle-to-grid environment. *Computer Networks* 2019; 153: 36–48.

11. Kaneriya S, Tanwar S, Nayyar A, et al. Data Consumption-Aware Load Forecasting Scheme for Smart Grid Systems. In: IEEE. ; 2018: 1–6.

12. Qureshi NMF, Bashir AK, Siddiqui IF, Abbas A, Choi K, Shin DR. A knowledge-based path optimization technique for cognitive nodes in smart grid. In: IEEE. ; 2018: 1–6.

13. Bayindir R, Colak I, Fulli G, Demirtas K. Smart grid technologies and applications. *Renewable and Sustainable Energy Reviews* 2016; 66: 499–516.

14. Reddy GT, Sudheer K, Rajesh K, Lakshmanna K. Employing data mining on highly secured private clouds for implementing a security-asa-service framework. *J. Theor. Appl. Inf. Technol* 2014; 59(2): 317–326.

15. RM SP, Bhattacharya S, Maddikunta PKR, et al. Load balancing of energy cloud using wind driven and firefly algorithms in internet of everything. *Journal of Parallel and Distributed Computing* 2020.

16. Bera S, Misra S, Rodrigues JJ. Cloud computing applications for smart grid: A survey. *IEEE Transactions on Parallel and Distributed Systems* 2014; 26(5): 1477–1494.

17. Zhang Q, Sun Y, Cui Z. Application and analysis of ZigBee technology for Smart Grid. In: IEEE. ; 2010: 171–174.

18. Maddikunta PKR, Gadekallu TR, Kaluri R, Srivastava G, Parizi RM, Khan MS. Green communication in IoT networks using a hybrid optimization algorithm. *Computer Communications* 2020.

19. Kappagantu R, Daniel SA. Challenges and issues of smart grid implementation: A case of Indian scenario. *Journal of Electrical Systems and Information Technology* 2018; 5(3): 453–467.

20. Mittal M, Iwendi C, Khan S, Rehman Javed A. Analysis of security and energy efficiency for shortest route discovery in low-energy adaptive clustering hierarchy protocol using Levenberg-Marquardt neural network and gated recurrent unit for intrusion detection system. *Transactions on Emerging Telecommunications Technologies* 2020: e3997.

21. Deepa N, Prabadev B, Kumar Reddy P, et al. An AI-based Intelligent System for Healthcare Analysis Using Ridge–Adaline Stochastic Gradient Descent Classifier. *Journal of Supercomputing* 2020.

22. Gadekallu TR, Khare N, Bhattacharya S, Singh S, Maddikunta PKR, Srivastava G. Deep neural networks to predict diabetic retinopathy. *JOURNAL OF AMBIENT INTELLIGENCE AND HUMANIZED COMPUTING* 2020.

23. Patel H, Singh Rajput D, Thippa Reddy G, Iwendi C, Kashif Bashir A, Jo O. A review on classification of imbalanced data for wireless sensor networks. *International Journal of Distributed Sensor Networks* 2020; 16(4): 1550147720916404.

24. Tian W. A Review of Smart Grids and Their Future Challenges. In: . 173. EDP Sciences. ; 2018: 02025.

25. Alazab M, Khan S, Krishnan SSR, Pham QV, Reddy MPK, Gadekallu TR. A Multidirectional LSTM Model for Predicting the Stability of a Smart Grid. *IEEE Access* 2020.

26. Venayagamoorthy GK. Dynamic, stochastic, computational, and scalable technologies for smart grids. *IEEE Computational Intelligence Magazine* 2011; 6(3): 22–35.

27. Bouktif S, Fiaz A, Ouni A, Serhani MA. Optimal deep learning lstm model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches. *Energies* 2018; 11(7): 1636.

28. Javed F, Arshad N, Wallin F, Vassileva I, Dahlquist E. Forecasting for demand response in smart grids: An analysis on use of anthropologic and structural data and short term multiple loads forecasting. *Applied Energy* 2012; 96: 150–160.

29. Hernández L, Baladrón C, Aguiar JM, Carro B, Sánchez-Esguevillas A, Lloret J. Artificial neural networks for short-term load forecasting in microgrids environment. *Energy* 2014; 75: 252–264.

30. Ahmad T, Chen H. Potential of three variant machine-learning models for forecasting district level medium-term and long-term energy demand in smart grid environment. *Energy* 2018; 160: 1008–1020.

31. Shayeghi H, Ghasemi A, Moradzadeh M, Nooshyar M. Simultaneous day-ahead forecasting of electricity price and load in smart grids. *Energy conversion and management* 2015; 95: 371–384.

32. Khan AR, Mahmood A, Safdar A, Khan ZA, Khan NA. Load forecasting, dynamic pricing and DSM in smart grid: A review. *Renewable and Sustainable Energy Reviews* 2016; 54: 1311–1322.

33. Raza MQ, Nadarajah M, Ekanayake C. On recent advances in PV output power forecast. *Solar Energy* 2016; 136: 125–144.

34. Raza MQ, Khosravi A. A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings. *Renewable and Sustainable Energy Reviews* 2015; 50: 1352–1372.

35. Gupta S, Kambli R, Wagh S, Kazi F. Support-vector-machine-based proactive cascade prediction in smart grid using probabilistic framework. *IEEE Transactions on Industrial Electronics* 2014; 62(4): 2478–2486.

36. Pan X, Lee B. A comparison of support vector machines and artificial neural networks for mid-term load forecasting. In: IEEE. ; 2012: 95–101.

37. Mitchell G, Bahadoorsingh S, Ramsamooj N, Sharma C. A comparison of artificial neural networks and support vector machines for short-term load forecasting using various load types. In: IEEE. ; 2017: 1–4.

38. Ali AS, Azad S. Demand forecasting in smart grid. In: Springer. 2013 (pp. 135–150).

39. Khan MS, Coulibaly P. Application of support vector machine in lake water level prediction. *Journal of Hydrologic Engineering* 2006; 11(3): 199–205.

40. Khemchandani R, Jeyadeva , Chandra S. Regularized least squares fuzzy support vector regression for financial time series forecasting. *Expert Systems with Applications* 2009; 36(1): 132–138.

41. Li W, Yan N, Zhang Zg. Study on long-term load forecasting of MIX-SVM based on rough set theory. *Power System Protection and Control* 2010; 38(13): 31–34.

42. Tay FE, Cao L. Modified support vector machines in financial time series forecasting. *Neurocomputing* 2002; 48(1-4): 847–861.

43. Hong WC. A hybrid support vector machine regression for exchange rate prediction. *International Journal of Information and Management Sciences* 2006; 17(2): 19–32.

44. Hong WC, Dong Y, Chen LY, Panigrahi BK, Wei SY. Support Vector Regression with Chaotic Hybrid Algorithm in Cyclic Electric Load Forecasting. In: Springer. ; 2012: 833–846.

45. Cao L, Gu Q. Dynamic support vector machines for non-stationary time series forecasting. *Intelligent Data Analysis* 2002; 6(1): 67–83.

46. Mohandes MA, Halawani TO, Rehman S, Hussain AA. Support vector machines for wind speed prediction. *Renewable energy* 2004; 29(6): 939–947.

47. Tripathy B, Acharjya D, Cynthya V. A framework for intelligent medical diagnosis using rough set with formal concept analysis. *arXiv preprint arXiv:1301.6011* 2013.

48. Reddy GT, Reddy MPK, Lakshmanna K, et al. Analysis of dimensionality reduction techniques on big data. *IEEE Access* 2020; 8: 54776–54788.

49. Tripathy B, Arun K. A new approach to soft sets, soft multisets and their properties. *International Journal of Reasoning-based Intelligent Systems* 2015; 7(3/4): 244–253.

50. Gola J, Webel J, Britz D, et al. Objective microstructure classification by support vector machine (SVM) using a combination of morphological parameters and textural features for low carbon steels. *Computational Materials Science* 2019; 160: 186–196.

51. Gopi AP, Jyothi RNS, Narayana VL, Sandeep KS. Classification of tweets data based on polarity using improved RBF kernel of SVM. *International Journal of Information Technology* 2020: 1–16.

52. Xie Y, Liu J, Huang T, et al. Outdoor thermal sensation and logistic regression analysis of comfort range of meteorological parameters in Hong Kong. *Building and Environment* 2019; 155: 175–186.

53. Kwon Y, Kwasinski A, Kwasinski A. Solar irradiance forecast using naïve Bayes classifier based on publicly available weather forecasting variables. *Energies* 2019; 12(8): 1529.

54. Zhang Q, Yang Y, Ma H, Wu YN. Interpreting cnns via decision trees. In: ; 2019: 6261–6270.

55. Javed AR, Sarwar MU, Khan S, Iwendi C, Mittal M, Kumar N. Analyzing the Effectiveness and Contribution of Each Axis of Tri-Axial Accelerometer Sensor for Accurate Activity Recognition. *Sensors* 2020; 20(8): 2216.

56. Javed AR, Beg MO, Asim M, Baker T, Al-Bayatti AH. AlphaLogger: Detecting motion-based side-channel attack using smartphone keystrokes. *Journal of Ambient Intelligence and Humanized Computing* 2020: 1–14.

57. Arzamasov V, Böhm K, Jochem P. Towards concise models of grid stability. In: IEEE. ; 2018: 1–6.

58. Pisica I, Eremia M. Making smart grids smarter by using machine learning. In: VDE. ; 2011: 1–5.

59. Hosseinzadeh J, Masoodzadeh F, Roshandel E. Fault detection and classification in smart grids using augmented K-NN algorithm. *SN Applied Sciences* 2019; 1(12): 1627.

60. Quinlan J. Program for machine learning. *C4. 5* 1993.

61. Kim G, Lee S, Kim S. A novel hybrid intrusion detection method integrating anomaly detection with misuse detection. *Expert Systems with Applications* 2014; 41(4): 1690–1700.

62. Mulay SA, Devale P, Garje G. Intrusion detection system using support vector machine and decision tree. *International Journal of Computer Applications* 2010; 3(3): 40–43.

63. Gush T, Bukhari SBA, Mehmood KK, Admasie S, Kim JS, Kim CH. Intelligent Fault Classification and Location Identification Method for Microgrids Using Discrete Orthonormal Stockwell Transform-Based Optimized Multi-Kernel Extreme Learning Machine. *Energies* 2019; 12(23): 4504.