Central Power Research Institute

Vol. 20(1)/07-15, June 2024
ISSN (Print) : 0973-0338
DOI : 10.33686/pwj.v20i1.1162

Power Research

Statistical Approach with Machine Learning-Based Intrusion Detection System for CyberAttack Discrimination in the Smart Grid

M. Nakkeeran[1*], V. Anantha Narayanan[1], P. Bagavathi Sivakumar[1] and S. Balamurugan[2]

[1]Department of Computer Science and Engineering, Amrita School of Computing, Amrita Vishwa Vidyapeetham, Coimbatore - 641112, Tamil Nadu, India; m_nakkeeran@cb.students.amrita.edu

[2]Department of Electrical and Electronics Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore – 641112, Tamil Nadu, India

Abstract

With rapid power grid digitalisation, keeping the private communications network utilities separate from the public communications networks is increasingly more challenging. It paves the way for the attacker to intrude into the industrial control system by compromising the networks. The proposed framework of Statistical Approach with a Machine Learning classifier (SAML) with Synthetic Minority Oversampling Technique (SMOTE) aims to improve early cyberattack discrimination in the smart grid with optimal hyperparameterized tuning of Principal Component Analysis (PCA) with ExtraTrees and AdaBoost Classifier for Feature Extraction (Dimensionality Reduction), bagging, and boosting, respectively. The significance of the SAML-PCA is that it can handle missing rates by replacing INFinity seen attack records with Zero for the specific column of apparent impedance of the relay to avoid blackouts and cascading failures. The proposed SAML-PCA model achieves a higher accuracy of 95.28% for ExtraTrees with Adaboost Classifier than the ML Classifiers and existing approaches.

Keywords: Blackouts, Cascading Failures, Dimensionality Reduction, Machine Learning, Statistical Approach

1. Introduction

The migration of industries to Industry 4.0 involves rapid deployment of Operational Technology (OT) and Information Technology (IT) convergence for smarter, faster, more cost-efficient, and better monitoring and control. Especially in the energy sector, the conventional power grid is changing into a smart grid for monitoring and control purposes from remote locations, which makes the grid more susceptible to cyberattacks. Due to the integration of OT and IT networks, the attack surface of the smart grid is growing significantly. Even though industries with firewalls and high-security passwords handle many security measures. There is always a pitfall whenever bidirectional communication is involved in the control system through the internet for remote control operations. The attackers might exploit the system's vulnerability through social engineering due to careless personnel passwords, bad practices of default or guessable passwords to all substation equipment, bypassing the controls with the turned-off security measures, and inadequate technology[1]. Therefore, the smart grid is becoming increasingly susceptible to cyber attack, which impose immediate danger to the nation's mission-critical infrastructure.

One of the earliest cyber-attacks on the power grids was the Aurora Generator Test in 2007, carried out as an experiment by the US Department of Energy. In this Attack, they targeted the control systems of a 2MW diesel generator. As a result, the generator started shaking and smoking, which caused physical damage[2]. In 2010, *Stuxnet* malware targeted a nuclear facility in Iran. It aims to take control and damage the facility's Industrial Control System (ICS) to impede the nuclear enrichment process. It was the first and most sophisticated malware used to target industrial control systems. There is also a possibility of targeting the whole

**Author for correspondence*

grid infrastructure rather than only individual components. Suppose the attackers can intrude on the control centres for power grid monitoring and control. In this case, the attacker can maliciously inject false data injection attacks to disconnect transmission lines, generators, substations, and other power system components.

The smart grid system is at risk from false data injection assaults since these attacks include manipulating and altering parameters such as current, voltage, phase angles, and sequence components. The intention is to deceive the operator without triggering any alarms. It can potentially cause a series of failures in the smart grid, ultimately leading to a complete loss of electricity in the system[3]. The impact creates catastrophic daily activities where the people depend on electricity. The power outages in Ukraine that happened in 2015 and 2016 back-to-back year incidents are proof of cyberattacks on the power grid. The attackers targeted the distribution system by taking unauthorized control of the ICS. These cyberattacks result in power outages, affecting hundreds of thousands of customers[4]. Mumbai, a major Indian city, lost electricity for about 12 hours on October 12, 2020[5]. According to a recent study, "RedEcho," a proactive hacker team, was connected to the power outage. The hackers used sophisticated malware to specifically target a regional control centre in a protracted assault that lasted for many months[6]. Therefore, protecting power systems from cyber threats has become a rapidly evolving and crucial field of study[7].

Signature-based IDS (firewalls) cannot capture advanced cyber-attacks and require frequent updating. Specification-based IDS, such as state estimation techniques, require more system expertise, complex logic, resource-intensive, and scalability issues. These challenges have motivated researchers to prefer Machine Learning techniques to provide defence-in-depth solutions with generalization and scalability[8].

The proposed work aims to provide a comprehensive and adaptable solution for the smart grid when the system is threatened by an attacker or intruder, whether from inside or outside the system. The key contributions of this study are outlined below:

- The proposed framework of **S**tatistical **A**pproach with a **M**achine **L**earning classifier **(SAML-PCA)** with SMOTE aims to improve early cyberattack discrimination in the smart grid with optimal hyperparameterized tuning of Principal Component Analysis with ExtraTrees and AdaBoost Classifier for Feature Extraction (Dimensionality Reduction), bagging, and boosting, respectively.
- The missing rate handled for the relay's apparent impedance with INFinity saw Attack records as Zero to avoid blackouts and cascading failures.
- SMOTE is applied to balance the dataset for model robustness and higher accuracy.

The content of this paper is divided into seven sections: **Section 2** explores the existing research work on Triple Class classification with the techniques used, limitations, and the drawbacks to be addressed. **Section 3** specifies an overview of the system architecture, dataset description, and recommended framework with the process flow diagram. **Section 4** outlines the methodology of the statistical approach with Principal Component Analysis (PCA) for extracting the top *k* principal components of features and the steps involved. **Section 5** describes the implementation detail of data preparation, the tools used for implementation, and the metrics used to evaluate the results. **Section 6** provides a comprehensive examination and discussion of the results, including tables and graphs. **Section 7** analyses the result and outlines the potential areas for further research.

2. Related Work

Some recent researchers have contributed to solving the cyberattack problem in smart grid systems. The techniques and limitations of the existing approach, as well as the challenges to be addressed, are discussed in this section.

Hink *et al.*[9] developed the **initial datasets** specifically emphasizing instances where a compromised system or insider attack targets a smart grid. The team analyzed power system interruptions using machine learning methodologies to distinguish cyberattacks from other causes. The dataset offered by this author's group provides the initial evidence for doing machine learning application research in Smart Grid to create an Intrusion Detection System (IDS). An inherent **limitation** of this article is its analysis of a mere 1% of randomly chosen data items from a comprehensive collection of 15 datasets of Triple Class. Using the Information Gain metric, the top 40 optimal features were selected and used to classify the Triple Class. They achieved 95.0% accuracy with the Adaboost + JRipper machine learning classifier. The authors propose evaluating future work possibilities using extensive power system data, machine learning algorithms, classification methodologies, and different quantities of labelled data.

Ankitdeshpandey and Karthi[10] used Principal Component Analysis (PCA) as a feature extraction technique for extracting features to decrease the number of dimensions to 31 Principal components for the Triple Class dataset. They applied various Machine Learning (ML) and Deep Learning (DL) algorithms. Finally, they achieved 91.14% accuracy with the Random Forest classifier. The paper's shortcoming lies in its testing methodology, which only included a limited sample size of around 13,200 samples. These samples were randomly selected from a total of 15 datasets.

Sunku Mohan *et al.*[11] applied the power domain knowledge to select the features manually. They have chosen the 36 potential impact features of +ve, -ve, and zero sequence components, log features to discriminate cyberattacks from natural events and normal events with various load variations. A Rule-based Machine Learning classifier (Random Forest) was used to discriminate the Triple Class, achieving an accuracy of 97.25%. The limitation of this study is that the manual selection of features is primarily tailored to a specific architecture, leading to an Intrusion Detection System (IDS), which is partly defined despite using a machine learning classifier for classification.

The overall drawback of the existing work is that the feature selection is not accurate enough to discriminate the cyberattack. Manual Feature selection is not feasible to address the problem of different architectures. Moreover, handling of the missing rate with the ***INF**inity* seen attack records of the apparent impedance of the relay is not specified. So, these challenges are addressed in our proposed framework of SAML-PCA.

3. System Architecture

3.1 Power System Framework

The power system framework configuration of the 3 bus/2 generator system was developed by authors[12] and shown in Figure 1[13]. The assumptions are based on the premise that an unauthorized individual has successfully breached the system, gained entry to the substation network, and sent instructions to the substation switch. The invader may originate from an external entity outside the network, a former employee of the organization, or a current employee. Due to the absence of an internal validation mechanism in IEDs (Intelligent Electronic Devices) to differentiate between authentic and deceptive faults, they use a distance protection approach to activate the breakers upon detecting defects. To perform maintenance, the operators can manually deactivate the breakers BR1 through BR4 by issuing commands to the Intelligent Electronic Devices (IEDs) R1 through R4. The manual override is typically carried out during line maintenance or when other system components malfunction.

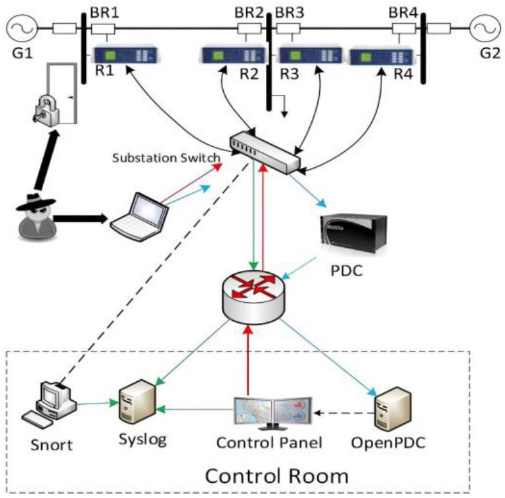

Figure 1. The power system framework configuration (3-Bus/2-generator System)[13].

Figure 1 of the system architecture[13] consists of various operational scenarios of *Single Line to Ground, Line Maintenance, Remote Tripping Command Injection Attack, Relay Setting Change Attack, and False Data Injection Attack.* The Phasor Measurement Unit (PMU) device synchronizes with a shared time source to calculate a phasor quantity's magnitude and phase angle of voltage or current. Each of the four PMU/relays is equipped with an integrated system that measures 29 features per PMU, which consists of 116 feature columns with 12 columns of control panel logs, snort alerts, relay logs, and the marker/target column. The complete description of the features dataset is available in[13].

3.2 ICS CyberAttack Power System Dataset

The ICS Cyber Attack Power System Dataset[13], accessible to the public, was developed in 2014 with a collaboration between Mississippi State University and Oak Ridge National Laboratory (US).

The SAML-PCA evaluates 15 combined Triple Class datasets to discriminate Cyberattacks from Natural Events and No Events. Each dataset contains around 5000 records with 128 feature columns and one marker/target label for classification. Table 1 depicts the 41 event scenarios split into Triple Class events of No Events, Natural Events, and Attack Events from the Multiclass label and Binary Class Label of the IEEE 3 Bus System[13].

- **No Events** - refers to the standard functioning of the system without any changes in the loads.
- **Natural Events** - refers to a system involving a Single Line-to-Ground (SLG) failure with different fault

locations in L1 and L2 and Line Maintenance in both L1 and L2.

- **Attack Events** - refer to three types of attacks: Data Injection Attacks (SLG fault replay), Remote Tripping Command Injection Attacks, and Relay Setting Change Attacks. These attacks include manipulating fault locations with different percentages.

Table 1. SAML-PCA with event scenario split for triple class

Types of Scenarios	Multiclass Labels	Binary Class	Triple Class
Normal Operation	41	Normal	No Events
Single Line-to- Ground Fault	1 to 6		Natural Events
Line Maintenance	13, 14		
False Data Injection Attack	7 to 12	Attack	Attack Events
Remote Tripping Command Injection Attack	15 to 20		
Relay Setting Change Attack	21 to 30, 35 to 40		

3.3 Proposed SAML-PCA Framework for IDS in Smart Grid

The proposed framework of the **SAML-PCA** approach, illustrated with the process flow diagram shown in Figure 2, is aimed at discriminating Cyber Attacks from Natural Events and No Events. This framework involves carrying out data wrangling pre-processing methods by considering “INFinity” attack observations as Zero for the feature columns of “PA: Z” (Apparent Impedance for Four Relays). The feature engineering method encompasses many steps, including SMOTE / Without SMOTE, train-test split, feature scaling, label encoding, and applying without PCA / PCA by Optimal Hyperparameter tuning with ML Classifier.

The proposed framework considers four possible combinations of without/with SMOTE and PCA for the performance comparison. Using stratified sampling, SMOTE equalizes the dataset by ensuring an equivalent number of records for the three class labels. Following the use of SMOTE, the dataset is divided into training and testing sets using an 80:20 ratio by the Pareto Principle. Feature Scaling using a Standard Scaler with Z-Score Normalization is applied to trained sets to standardize the ranges, making them comparable and preventing certain features from dominating others due to their scale. The label encoder is applied to the marker/target column to convert categorical data into numerical values to help facilitate the conversion, and this is then followed by dimensionality reduction (feature extraction) using PCA (principal component analysis). During the model selection and parameter tuning stage, PCA extracts features optimally.

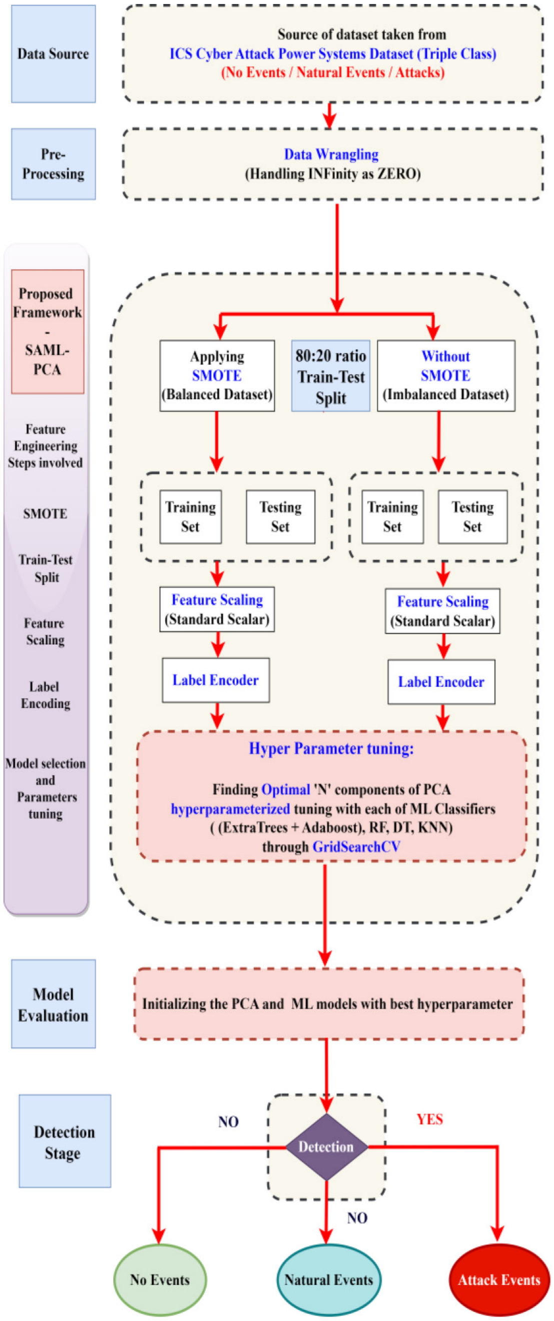
Data Source
Source of dataset taken from
ICS Cyber Attack Power Systems Dataset (Triple Class)
(No Events / Natural Events / Attacks)
Pre-Processing
Data Wrangling
(Handling INFinity as ZERO)
Proposed Framework - SAML-PCA
Feature Engineering Steps involved
SMOTE
Train-Test Split
Feature Scaling
Label Encoding
Model selection and Parameters tuning
Applying SMOTE (Balanced Dataset)
80:20 ratio Train-Test Split
Without SMOTE (Imbalanced Dataset)
Training Set
Testing Set
Training Set
Testing Set
Feature Scaling (Standard Scalar)
Feature Scaling (Standard Scalar)
Label Encoder
Label Encoder
Hyper Parameter tuning:
Finding Optimal 'N' components of PCA
hyperparameterized tuning with each of ML Classifiers
((ExtraTrees + Adaboost), RF, DT, KNN)
through GridSearchCV
Model Evaluation
Initializing the PCA and ML models with best hyperparameter
Detection Stage
NO
YES
Detection
NO
No Events
Natural Events
Attack Events

Figure 2. The proposed framework of the SAML-PCA process flow diagram to discriminate cyberattacks from natural and no events.

GridSerachCV finds the optimal *'N'* Component hyperparameter tuning for each ML Classifier used. *GridSerachCV* exhaustively searches for the best parameters from the provided grid parameters. The ML Classifiers applied for training and testing the datasets are ExtraTrees with Adaboost Classifier (ET + AdB), Random Forest (RF), and Decision Tree (DT)[14]. The Model Evaluation Stage initializes the PCA and ML models with the best hyperparameter to evaluate the test data. The detection stage performs the discrimination of Cyberattacks from Natural Events and No Events to achieve better performance metrics of higher accuracy and less execution time.

4. Methodology

4.1 Principal Component Analysis

PCA[10] is a dimensionality reduction (feature extraction) technique of statistical approach that can be employed to handle multivariate features of power system data for attack detection. The curse of dimensionality refers to the phenomenon where an exponential rise in the number of features (dimensions) leads to a proportional increase in the quantity of data needed to achieve correct generalization with accuracy. High-dimensional data can lead to overfitting and increased computational complexity in the power system datasets of the 'N' bus system. It often contains correlated or redundant features. PCA identifies the principal components, which are uncorrelated and capture the maximum variance in the data. Removing redundant information can lead to more straightforward and interpretable models with improved generalization performance.

PCA helps address the curse of dimensionality by reducing the number of features into components while retaining most of the variability in the data. Reducing the number of features with PCA makes the training process faster, so the model can be trained faster with fewer computation resources.

PCA projects the matrix into a linear space of lower dimensionality. The process converts a group of variables associated with each other into a new set of variables with no correlation, referred to as principal components.

The major components are arranged in a decreasing order based on their variance, thereby collecting the most essential information from the top k components.

4.2 Steps Involved

The steps involved in the dimensionality reduction of power system feature columns are shown below:

Step 1: Standardization

Before using PCA, the data undergoes standardization using Standard Scalar with Z-score Normalization. This involves subtracting the mean and dividing it by the standard deviation for each feature. It guarantees that every feature makes an equal contribution to the analysis.

Step 2: Covariance Matrix Calculation

The covariance matrix contributes to understanding how different features in the data are related. The covariance between the two features X_i and X_j is given in (1).

$$C_{ij} = \frac{1}{n-1}\sum_{k=1}^{n}(X_{ki}-\bar{X}_i)(X_{kj}-\bar{X}_j) \tag{1}$$

Here, $\bar{X}_i$ and $\bar{X}_j$ are the means of variables X_i and X_j, respectively. Where, k represents the index of each observation in the dataset.

Step 3: Eigenvalue Decomposition

Following this, the eigenvectors and eigenvalues of the covariance matrix are determined. The eigenvectors correspond to the principal components or directions, while the eigenvalues indicate the extent of variation along those directions. The relationship between the eigenvector $\boldsymbol{v}$ and eigenvalues λ described in (2).

The equation for eigen decomposition is :

$$Cv = \lambda v \tag{2}$$

The eigenvectors are typically normalized to unit length, and the eigenvalues are sorted in descending order.

Step 4: Selection of Principal Components

Sorting the eigenvectors in a decreasing order depending on their associated eigenvalues. The eigenvector corresponding to the largest eigenvalue represents the primary principal component, the eigenvector associated with the second largest eigenvalue represents the secondary principal component, and so on. A higher eigenvalue indicates that the related main component captures a greater amount of variation. The process of selecting the top k eigenvectors, also known as principal components, involves choosing the k eigenvectors with the highest eigenvalues. These eigenvectors are then used to create the transformation matrix P.

Step 5: Projection for Dimensionality Reduction

The original data is then projected into a new subspace with the top k eigenvectors (principal components). The transformation of a data point $\boldsymbol{x}$ to the new subspace is given in (3).

$$\text{Projected Data} = P^T x \tag{3}$$

where, P^T is the transpose of the transformation matrix P.

In summary, PCA involves standardizing the data, computing the covariance matrix, and finding the eigenvectors and eigenvalues. Sorting, creating a projection matrix, and finally, projecting the data onto the new subspace. The resulting transformed data retains the essential information while reducing dimensionality.

5. Implementation Detail

The **SAML-PCA** approach is implemented to discriminate between three types of events: No Events, Natural Events, and Attack Events.

Table 2 depicts the before and after SMOTE operation with an 80:20 ratio train-test split with the merged 15 datasets of Triple Class for training and testing purposes.

5.1 Implementation Tool and Evaluation Metrics

The free and open-source Google Colab Data Analytics platform is used for the implementation. The system employs Python 3 on Google Compute Engine, with a RAM capacity of 13 GB, a 2-core Xeon CPU running at 2.20 GHz, and a 108 GB hard drive. The key metrics used for evaluating the performance of the proposed SAML-PCA model are accuracy (4), precision (5), recall (6), and F1-score (7). These metrics are determined using the generic representation of the confusion matrix[15], as shown in Table 3.

Table 2. SAML-PCA before and after SMOTE records with the train-test split from the merged 15 datasets

Dataset Used	**IEEE 3 Bus System[13]**							
Feature EngineeringAspect	**Without SMOTE (Original Dataset- Imbalanced)**				**With SMOTE (Balanced Dataset)**			
Event Types	No Events records	Natural Events records	Attack records	**Total Records**	No Events records	Natural Events records	Attack records	**Total Records**
Training Samples (1)	3524	14647	44530	62701	44530	44530	44530	133590
Testing Samples (2)	881	3662	11133	15676	11133	11133	11132	33398
No. of records (1+2)	4405	18309	55663	**78377**	55663	55663	55662	**166988**

Table 3. Confusion Matrix

No. of testing samples (N records)		**Predicted class**	
		Classified as normal	**Classified as attack**
Actual Class	Normal Data	True Negative (TN)	False Negative (FN)
	Attack Data	False Positive (FP)	True Positive (TP)

(i) Accuracy: Accuracy is the ratio of accurately predicted samples to the total number of predictions.

$$\text{Accuracy(A)} = \frac{(\text{TP}+\text{TN})}{(\text{TP}+\text{TN}+\text{FP}+\text{FN})} * 100 \tag{4}$$

(ii) Precision: Ability to predict correctly.

$$\text{Precision(P)} = \frac{\text{TP}}{(\text{TP}+\text{FP})} * 100 \tag{5}$$

(iii) Recall: Ability to detect correctly.

$$\text{Recall(R)} = \frac{\text{TP}}{(\text{TP}+\text{FN})} * 100 \tag{6}$$

(iv) F1-Score: Harmonic mean of precision and Recall.

$$\text{F1-Score(F1)} = \frac{1}{\frac{(\frac{1}{\text{P}}+\frac{1}{\text{R}})}{2}} * 100 \tag{7}$$

6. Result Analysis and Discussion

The **SAML-PCA** approach presents a robust framework for discriminating cyberattacks from natural events, and no events are represented using tables and graphs. Figure 3 represents the Triple Class datasets before and after the SMOTE operation. Stratified sampling in SMOTE considers equal samples from each of the three classes. So, the model best fits trained samples, which can be evaluated with test samples.

Table 4 represents the before and after SMOTE operation without/with the PCA technique with an 80:20 ratio train-test split for the four possible combinations. The four possible combinations show the importance of SMOTE and PCA by comparing them with the performance metrics of accuracy (vs.) testing time (execution time). The three labels in the test records show the count on each label used where '0' represents Attack, '1' represents Natural Events, and '2' represents No Events.

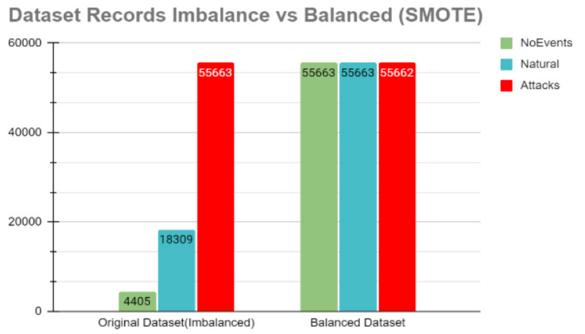

Figure 3. Represents the dataset with the original (imbalanced) and balanced (SMOTE) dataset.

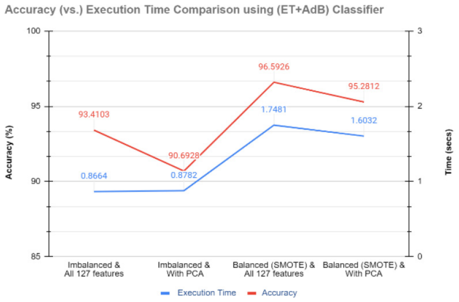

Figure 4. Accuracy (vs.) Execution Time metric comparison with the possible combination of before and after (SMOTE) and without/with PCA using (ET+AdB) classifier.

Table 4. Performance comparison before and after SMOTE and without/with PCA technique applied

Possible Combinations With (ET + Adaboost) ML Classifier	Training Records	Test Records	Test Records label count	Training Time (sec)	Testing Time (sec)	Accuracy (%)
Imbalanced and All 127 features	62701	15676	0: 11133 1: 3662 2: 881	26.363	**0.8663**	**93.41**
Imbalanced and With PCA	62701	15676	0: 11133 1: 3662 2: 881	17.08	**0.8782**	**90.69**
Balanced (SMOTE) and All 127 features	133591	33398	0: 11133 1: 11133 2: 11132	57.31	**1.7481**	**96.59**
Balanced (SMOTE) and With PCA	133591	33398	0: 11133 1: 11133 2: 11132	39.83	**1.6032**	**95.28**

Figure 4 shows the results of four possible combinations of accuracy (vs.) execution time. From Figure 4, it is inferred that the accuracy of the imbalanced data is 93.41% and 90.69%, less than 96.59% and 95.28% with SMOTE (balanced). Balanced (SMOTE) with all 127 features yields a higher accuracy of 96.59% than balanced (SMOTE) with a PCA of 95.28%. The testing time of balanced (SMOTE) with all 127 features consumes 1.74 seconds, which is higher than balanced (SMOTE) with PCA of 1.60 seconds. The conclusion can be made that Balanced (SMOTE) with PCA seems to be better and robust, with an Accuracy of 95.28% and a testing time of 1.60 secs.

Inference from Figure 5 shows that the (ExtraTrees + AdaBoost) Classifier achieves higher accuracy, precision, recall, and F1-score of 95.28% compared to the other three ML Classifiers of Random Forest (RF) and Decision Tree (DT) with 94.46% and 88.81% accuracy respectively. The other performance metrics of precision, recall, and F1-score are also more or less the same with accuracy for both the RF and DT Classifier.

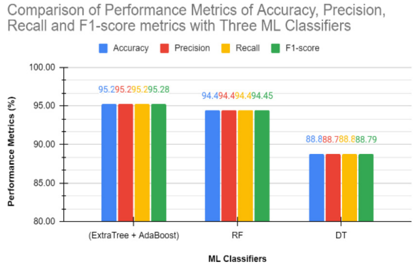

Figure 5. Performance metrics comparison across three ML Classifiers with SMOTE and PCA applied.

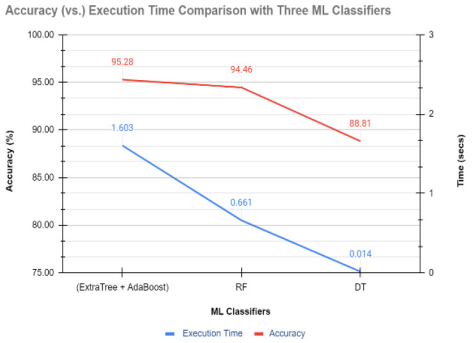

Figure 6. Accuracy (vs.) execution time metric across three ML Classifiers with SMOTE and PCA applied.

Inference from Figure 6 shows that the (ExtraTrees + AdaBoost) Classifier achieves a higher accuracy of 95.28% with an execution time of 1.603 secs. Meanwhile, RF achieves 94.46% with 0.661 secs execution time and DT with 88.81% accuracy with 0.014 secs execution time. Even though the execution time of (ExtraTrees + AdaBoost) is slightly higher than RF and DT, better and more robust performance metrics are achieved through the (ExtraTrees + AdaBoost) Classifier.

Table 5 represents the accuracy metric comparison of the proposed approach of SAML-PCA with the existing work of Pros and Cons discussed in detail. The proposed approach of SAML-PCA is robust enough to discriminate cyberattacks with higher accuracy compared to the existing approaches. Specifically, we addressed the problem of "**INF**inity," seen attack records are handled by replacing with ZERO to avoid missing rates. Other existing approaches have not addressed the missing rates.

Table 5. Accuracy metric comparison of the proposed approach with the existing work

Reference paper	Feature selection/ extraction	Number of features selection (or) extraction	Machine learning classifiers	Accuracy (%)	Pros/cons of the techniques used in the existing work and compared with our proposed work of SAML-PCA
Borges *et al.*[9] **(Original author dataset)**	Information gain	40	Adaboost + JRipper	95.00	• Only 1% of randomly sampled records from the 15 datasets were tested and the handling of "INFinity" seen attack records was not specified**.**
Ankitdesh pandeyand Karthi[10]	PCA	31	Random Forest	91.14	• Tested for reduced samples of 4400 records from each of the three classes. Lack of model robustness with untrained entire samples and **not specified about handling "INFinity" seen attack records.**
Mohan and Sankaran[11]	Manually selected features using Power Domain Knowledge	36	Rule-Based + ML Classifier (Random Forest)	97.25	• Feature Selection is done manually with Power System Domain knowledge selecting the features (+ve, -ve, Zero Seq. Components and Logs). This model **cannot be generalizable and scalable** for different architectures and requires complex logical calculations. • **Not Specified about the handling of "INFinity" seen attack records.**
SAML-PCA (Proposed Work)	**PCA with SMOTE**	**35**	**(ET + AdB)**	**95.28**	• Our proposed statistical approach is **robust** enough to discriminate cyberattacks with high accuracy and reasonable execution time. • This model can also be extended to different architectures for **scalability** since it deals statistically with data. • "**INF**inity" seen attack records "Apparent impedance" Relay feature are handled by replacing them with ZERO to avoid missing rates.

7. Conclusion and Future Work

The smart grid's mission-critical infrastructure requires more attention, where a few misclassifications of cyberattack incidents might create fatal consequences for the power system's stability and reliability, leading to blackouts and cascading failures.

The proposed framework of Statistical Approach with a Machine Learning classifier based on Principal Component Analysis (SAML-PCA) with SMOTE, ExtraTrees, and AdaBoost Machine Learning Classifier with optimal hyperparameters tuning achieved a higher accuracy of 95.28% with execution time of 1.60 secs. The SAML-PCA provides a robust solution to address the missing rate with early discrimination of cyberattacks from natural events and no events in the smart grid. The proposed model can be extendable to future work by optimizing the features. Further, the scalability of the IEEE 'N' bus system can be adopted for different architectures.

8. Funding

This work was supported by the Amrita Vishwa Vidyapeetham (Deemed University) under the Amrita Seed Grant Fund Program (File No: ASG2022051)

9. References

1. Sridhar S, Hahn A, Govindarasu M. Cyber–physical system security for the electric power grid. Proc of the IEEE. 2012; 100(1):210-24. https://doi.org/10.1109/JPROC.2011.2165269
2. Liu CC, Stefanov A, Hong J, Panciatici P. Intruders in the grid. IEEE Pow Energ Mag. 2012; 10(1):58-66. https://doi.org/10.1109/MPE.2011.943114
3. Amin BMR, Hossain MJ, Anwar A, Zaman S. Cyber attacks and faults discrimination in intelligent electronic device-based energy management systems. Electron. 2021; 10(6):650. https://doi.org/10.3390/electronics10060650
4. Hemsley KE, Fisher RE. History of industrial control system cyber incidents. Techn Rep: Hist Indust Cont Syst Cyber Incid. 2018. https://doi.org/10.2172/1505628
5. Rajkumar VS, Ştefanov A, Presekal A, Pálenský P, Rueda JL. Cyber attacks on power grids: Causes and propagation of cascading failures. IEEE Access. 2023; 11:103154-76. https://doi.org/10.1109/ACCESS.2023.3317695
6. Recorded Future. Continued targeting of Indian power grid assets by Chinese state-sponsored activity group [Internet]. 2022. [cited 7 Dec 2022]. Available from: https://go.recordedfuture.com/hubfs/reports/ta-2022-0406.pdf
7. Peng C, Sun H, Yang M, Wang YL. A survey on security communication and control for smart grids Under malicious cyber attacks. IEEE Trans Syst, Man, Cybernet: Syst. 2019; 49(8):1554-69. https://doi.org/10.1109/TSMC.2018.2884952
8. Sahani N, Zhu R, Cho JH, Liu CC. Machine learning-based intrusion detection for smart grid computing: A survey. ACM Transact Cyber-Phy Syst. 2023; 7(2):1-31. https://doi.org/10.1145/3578366
9. Hink RCB, Beaver JM, Buckner MA, Morris T, Adhikari U, Pan S. Machine learning for power system disturbance and cyber-attack discrimination. 2014 7th Inter Symp Resil Cont Syst; 2014. https://doi.org/10.1109/ISRCS.2014.6900095
10. Ankitdeshpandey, Karthi R. Development of intrusion detection system using deep learning for classifying attacks in power systems. Adv Intell Syst Comput. 2020:755-66. https://doi.org/10.1007/978-981-15-4032-5_68
11. Mohan VS, Sankaran S. Intelligent approach for analysis and diagnosis of attack, fault and load Variation in SCADA systems: A power system application. Lect Note Elect Eng. 2022:1-28. https://doi.org/10.1007/978-981-16-6081-8_1
12. Pan S, Morris T, Adhikari U. Classification of disturbances and cyber-attacks in power systems using heterogeneous time-synchronized data. IEEE Transact Ind Inform. 2015; 11(3):650-62. https://doi.org/10.1109/TII.2015.2420951
13. Adhikari U, *et al.* Industrial Control System (ICS) cyber attack datasets used in the experimentation. [Internet]; 2014. Available from: http://www.ece.uah.edu/~thm0009/icsdatasets/triple.7z at website: https://sites.google.com/a/uah.edu/tommy-morris-uah/ics-data-sets Dataset Description: http://www.ece.uah.edu/~thm0009/icsdatasets/PowerSystem_Dataset_README.pdf
14. Balan A, Srujan TL, Manitha PV, Deepa K. Detection and analysis of faults in transformer using machine learning. 2023 Internat Conf Intell Data Comm Technol Internet Things (IDCIoT), India: Bengaluru; 2023. https://doi.org/10.1109/IDCIoT56793.2023.10052786
15. Nakkeeran M, Narayanan VA. Anomaly detection in SCADA industrial control systems using bi-directional long short-term memory. Lect Note Elect Engineer. 2023; 415-36. https://doi.org/10.1007/978-981-99-3481-2_33