IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 27 April 2024, accepted 17 May 2024, date of publication 23 May 2024, date of current version 3 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3404814

RESEARCH ARTICLE

Mitigating Insider Threat: A Neural Network Approach for Enhanced Security

P. LAVANYA, H. ANILA GLORY, AND V. S. SHANKAR SRIRAM
Center for Information Super Highway (CISH), School of Computing, SASTRA Deemed University, Thanjavur, Tamil Nadu 613401, India
Corresponding author: V. S. Shankar Sriram (sriram@it.sastra.edu)

ABSTRACT Detecting insider threats is the foremost challenge in many institutions because of the abnormal behavior of legitimate access and network crawling in the Internet of Things (IoT) environment. The insider activities of the institution's data are submerged in many regular activities, leading to a data imbalance problem. Existing insider threat detection techniques often fail to address the data imbalance problem in the insider threat data of IoT-enabled institutions, thereby causing deterioration in detection performance. Thus, this paper presents a novel Enhanced Bidirectional Generative Adversarial Network (EBiGAN) for adversarial sample generation and a Deep Neural Network (DNN) with the Probability of Improvement (PI) acquisition function of Bayesian optimization to detect insiders in an IoT enabled institutions. The proposed model involves three modules: (1) Improved PCA for extracting user functionality samples and outlier estimators of k-means for grouping scenario-based user functional data. (2) Bidirectional GAN with an additional discriminator to ensure the quality of the generated samples (3) The PI acquisition function of Bayesian Optimization for tuning the hyperparameter to improve the performance of the DNN model for insider threat detection to secure IoT-enabled institutions. The performance of the Enhanced BiGAN and DNN-PI was evaluated using a benchmark institutional dataset. The experimental results show that the proposed model identifies the suspicious behavior of insiders with a high detection rate and minimal false alarm rate in an IoT infrastructure.

INDEX TERMS Data augmentation, deep neural network, insider attack, insider threat detection, outlier estimator K-means clustering.

I. INTRODUCTION

The Internet of Things (IoT) infrastructure is a significant part of institutional applications in educational institutions, healthcare, and corporate enterprises. IoT devices are connected through sensors to gather information from the environment and are secured through access control management devices, cameras and secure IoT protocols. However, IoT-specialized institutional infrastructure is affected by cyber threats and attacks in recent times have faced many difficulties in detecting insider threats. An Insider threat is a security breach enacted by authorized access of individuals within IoT-enabled institutions [1]. Insiders may have legitimate access to confidential data from employees, contractors, or business partners within an institution. Insiders intend to exfiltrate data, sabotage, fraud, and espionage. Insiders often operate through authorized access, which complicates the discerning of legitimate users and insiders [2]. Furthermore, it is difficult to observe changes in user behavior, encrypt insider activities and analyze massive volumes of data in institutions. The 2023 insider threat report states that insider attacks are vulnerable and occur more often in 74% of organizations [3].

The institutional IoT architecture comprises of five layers: business, application, processing, transport, and perception. Depending on the IoT layer, insider threats can occur within institutions with unauthorized access to IoT devices. Employees of institutions have access to modifying the components and information of IoT sensors in the perception layer [4]. Insiders can interrupt the data transmission from an IoT-enabled gateway in the transport layer. In the processing layer, the input data may have improper analytics, or the user can inject incorrect data into the layer, leading to an insider

The associate editor coordinating the review of this manuscript and approving it for publication was Qilian Liang.

threat [5]. Data exfiltration means that confidential data are dripped owing to unauthorized employees in the application layer and institution employees who intentionally practiced data leakage in the IoT architecture's business layer [6]. Numerous traditional approaches have been proposed to combat insiders in IoT infrastructure; however, these approaches are challenging because of inadequate real-time datasets and skewed class distributions in the dataset [7].

The skewed class distribution of the insider threat dataset of IoT-enabled institutions was resolved using oversampling. The widely used oversampling techniques include the Synthetic Minority Oversampling Technique (SMOTE) [8] and the Adaptive Synthetic Sampling Technique (ADASYN) [9]. Nevertheless, these methods are unsuitable for heterogeneous datasets, because they may lead to model overfitting. It can be handled using data augmentation techniques to identify unobserved data and maintain the generalizability of the model [10]. Hence, this research article proposes an Enhanced Bidirectional Generative Adversarial Network (EBiGAN) to address the data imbalance problem of an institution's IoT-enabled insider threat dataset.

Recently, deep learning models have played an important role in insider threat detection (ITD) in IoT infrastructure to achieve a high detection rate and a lower false alarm rate. However, certain limitations exist, such as data imbalance, dimensionality reduction, computational complexity and hyperparameter tuning, which affect the performance of ITD system. Thus, improved Principal Component Analysis (IPCA) was used to extract the user functionality-based samples and outlier-resistant estimators of k-means clustering to cluster the scenario-based user functionality samples.An Enhanced BiGAN was used to balance the imbalanced data, and hyperparameter tuning of the Deep Neural Network (DNN) model was performed using the Probability of Improvement (PI) acquisition function of Bayesian optimization over the GPyOpt tool for the detection of insider threats. This increased the overall performance and decreased the loss of the objective function while validating the ITD model for the IoT-enabled infrastructure. The main objectives of the proposed model are as follows,

- IPCA for extracting user functionality-based samples and outlier estimators of k-means clustering for scenario-based user functional data were utilized to provide better quality clusters and ensure standard stability and robustness to the detection model.
- EBiGAN is composed of an additional discriminator to ensure the quality of the encoded data by assessing the correlation between the encoded data and data from the latent space.
- EBiGAN generates an adversarial sample that performs interpolation in a latent space using an improved and diversified adversarial sample generation.
- A DNN hybridized with a Probability of Improvement (PI) acquisition function in Bayesian optimization for hyperparameter tuning solves the complexity and non-smoothness problems of the objective function, which can achieve higher performance.
- An improved DNN for insider threat detection was proposed to detect suspicious behavior of users with a high detection rate and a minimal false alarm rate.
- The release of the insider threat dataset (r6.1 &r6.2) from the Computer Emergency Response Team (CERT) of Carnegie Mellon University (CMU) was incorporated into the IoT layered architecture, which was used for experimentation and evaluated in terms of accuracy, precision, detection rate, false positive rate and false negative rate.

The rest of the article is structured as follows: Section II presents a detailed study of existing ITD and ITD in IoT infrastructure. Section III presents preliminaries of the proposed model. Section IV describes the research objective and a comprehensive description of the proposed method. Intensive experimentation is presented in Section V. Section VI concludes the paper with a scope for future work.

II. RELATED WORK

Further research has been conducted to detect, mitigate, and prevent insider threat. Recent ITD studies have mainly focused on machine learning and deep learning techniques compared with traditional methods. The three significant phases involved in designing an ITD framework are (i) class imbalance, (ii) dimensionality reduction, and (iii) anomaly-based insider threat detection, as discussed in the literature.

A. INSIDER THREAT DETECTION

The approach presented in [11] is a resource access pattern network (RAP-Net) with neural network techniques and reinforcement learning-based generative adversarial networks to solve imbalanced data problems. The embedding layer "word2vec" computes distance measurements and semantic correlations to observe various user behaviors in the patterns. The obtained feature vectors for the classification models include an integrated classifier model of a 1D convolutional neural network, bi-directional long short-term memory (Bi-LSTM) and Attention Neural Network (ANN) in the CMU CERT r4.2 dataset. The proposed 1D CNN was used for sequential feature extraction, Bi-LSTM for collecting time-based user actions, and ANN for user behavior-based insider threat identification and classification of normal and malicious with better accuracy. However, the proposed model has computational complexity owing to multiple integrated neural network types. Four traditional classification models [12] for ITD, logistic regression, decision tree, random forest, and XGBoost were implemented in the CMU CERT r4.2 dataset. The SMOTE sampling technique was used to address the class-imbalance problem. The frequency-based feature extraction approach derives feature vectors and achieves sample space reduction. Each model was evaluated using standard performance metrics to detect and classify the insiders. The experimental results were obtained before and after balancing the dataset, and compared to all the models,logistic regression outperformed all other techniques. However, the SMOTE sampling technique leads to model overfitting. Dual domain graph-based convolution Network (DD-GCN) [13] was developed for an adaptive anomaly-based ITD. The similarity metric of the weighted feature method was used to compute the similarity between the features of users and their behavioral data. The weighted feature similarity metric was primarily used to obtain highly qualified structural data. Dual domains with dual convolutional graph neural networks were designed to fuse the information of structural relationships and features, which were delivered as additional detection components. The combination and difference constraints verified the consistency and disparity of the trained DD-GCN model. Furthermore, the attention model was integrated with DD-GCN to attain better accuracy for publicly available datasets. The proposed model addresses this issue by computing the similarity between user behaviors and by detecting insider threats. However, the class imbalance problem of insider threat datasets must be addressed using this approach. The model proposed [14] is an employee relationship model for the effective dissemination of insider knowledge within an organization. The proposed model was classified into two phases. In phase one, data security queries arise for user organizations based on employee and organizational aspects in order to evaluate insider threat levels. In phase two, the construction of the employee relationship model using TOPSIS defines the impact of how the insider threat is inclined within the organization. Furthermore, the graph structure of the insider is constructed using a parameterized employee relationship model and is evaluated using synthetically generated log records and psychometric tests which are processed according to the user's relationships and organizational infrastructure. Nevertheless, synthetically generated insider data can be compared with a benchmark dataset to provide insight into the proposed model. An ITD using machine learning-based user behavior analytics [15] was designed to map the behavioral changes in user activities. The mapped user behavior changes are streamed activities in a sequential form. The Recurrent Neural Network (RNN) is used for feature representations after preprocessing using min-max normalization in the CMU CERT r4.2 dataset. To define the best feature extraction for sequential event activities, temporal-based Long Short-Term Memory (LSTM) is primarily used for user behavior analytics such as pattern learning. The LSTM was found to be a detection model with a low mean squared error rate. However, the dataset contained a minority of adversarial samples that were not addressed, and the researchers concluded that the proposed detection model was less accurate with imbalanced data. The model presented in [16] is a machine learning model based on ITD using synthetically generated log records from China's Civil Aviation Flight University website. The detected anomalies are clustered without annotated data owing to user behavior modification patterns by considering the IP addresses where the insiders are identified. The performance of the proposed model was better than that of other machine learning techniques in terms of precision, recall, and f1score. However, the proposed model must define feature engineering techniques for synthetically generated datasets. The proposed method [17] suggests a machine-learning framework for an ITD. An unsupervised machine learning model, isolation forest, and elliptic envelope framework were employed to view data from various sources and detect insider threats. The proposed model was evaluated using the CMU CERT insider threat test dataset, and achieved greater accuracy, sensitivity, specificity, f1score, and MCC. When working with the dataset, the data imbalance problem must be resolved.

B. ITD IN IoT INFRASTRUCTURE

A trustworthy machine learning-based insider threat detection model [18] was developed to detect insiders by assuring both confidentiality and explication. The performance of ML models has improved through data collaboration among several owners. However, the proposed ML model concentrates on the need for more reliable insider threat detection solutions that specifically address the challenges related to trustworthy learning. Designed a taxonomy of adversarial techniques [19] that insiders can use, and examined how vulnerable machine learning models are to adversarial attacks in the context of the Internet of Things (IoT). It also attempts to increase the knowledge of the current insider threat scenario in the IoT ecosystem and to investigate defensive strategies against adversarial attacks. This study primarily examines supervised machine learning systems that are specifically linked to the Internet of Things (IoT), excluding other categories of machine learning models or applications. However, this discussion does not cover the potential ramifications or outcomes of insider threats to the IoT ecosystems. Modelled [20] a security system utilizing machine learning to identify insider assaults in IoT devices. The proposed assault exploited the weaknesses of the RPL routing protocol. The performance of machine learning ensures that insider threats can be effectively identified. The significant consequences of deploying security service that arise in IoT devices include restricted computational power, storage capacity, and communication capacity. IoT devices are susceptible to security breaches owing to inadequate physical security measures. Conventional security measures can be more efficient in mitigating attacks specifically targeting IoT devices. A novel blockchain-based anomaly detection technique [21] to mitigate insider assaults in IoT systems. It focuses on edge computing and addresses the concerns related to limited availability, potential data loss, and compromised data integrity. It uses edge computing to minimize latency and bandwidth demands, enhance availability, and prevent potential points of failure. Furthermore, it incorporates decentralized edge computing with blockchain technology to effectively identify and rectify anomalies in incoming data from the sensors. The assessment of the methodology using an actual IoT system dataset demonstrated the successful attainment of the stated objective while simultaneously guaranteeing the preservation of data integrity and availability, both of which are crucial for the implementation of IoT systems. However, the blockchain model must discuss performance metrics where full-fledged detection has not yet been established.

Moreover, a study on insider threat detection revealed that addressing imbalanced data problems is inevitable. Several detection approaches have been explored, such as profile-based user behavior, log record monitoring and analysis systems, and content-based methods. However, a single model cannot provide a security-based solution, whereas a hybrid model implements various integrated techniques to improve the accuracy and increase the computational complexity of the model. However, deep learning-based solutions have provided a new opportunity to develop a robust insider threat detection model. During implementation and validation, it was ensured that the deep learning models could significantly improve the accuracy and detection rate with reduced false positives. Thus, this research attempts to overcome the literature challenges by proposing an improved PCA that extracts user functionality-based samples and outlier estimators of k-means clustering groups scenario-based user functional data to reduce the dimensionality without changing the underlying data structure. The enhanced Bidirectional GAN-based Probabilistic of Improvement (PI) acquisition function for hyperparameter tuning of DNN to detect insiders, where the EBiGAN solves the data imbalance problem.

III. PRELIMINARIES

A. OUTLIER ESTIMATORS OF K-MEANS CLUSTERING

K-means clustering is an unsupervised algorithm for grouping similar points through iterative processes that form clustering. This technique visualizes complex data in an understandable format for predetermined clusters. K-means clustering aims to form k clusters depending on n observations, where k is the centroid, and n is the number of data points joined based on heterogeneous centroids. The formation of k clusters was achieved by abating the summation of distances between the points and their corresponding centroids using the Euclidean distance. However, Euclidean distance has more consequences when framing detection or identification algorithms for many security-based applications. Owing to the need for a strong bond between the data points and centroids, outliers can interrupt cluster formation and reduce the computational complexity. This can be addressed by outlier estimators, which fix the Huber loss to compute the distance between the data points and centroids. Huber loss combines the mean squared error (MSE) and mean absolute error (MAE) [22]. Steps for outlier estimators of k-means clustering with the objective function in (1)

Objective function

$$F = \sum\nolimits_{i=1}^{l} \sum\nolimits_{j=1}^{m} a_{ij}.outlier\ estimators\left(c_j, d_i\right) \tag{1}$$

Compute Huber loss in (2)

$$HL = HL_{\delta}(a, f(x)) \begin{cases} \frac{1}{2}(a - f(x))^2 & |y - f(x)| \leq \delta \\ \delta\,|a - f(x)| - \frac{1}{2}\delta^2 & otherwise \end{cases} \tag{2}$$

where $\frac{1}{2}(a - f(x))^2$ is the quadratic form of the MSE. It happens when small error exists or else MAE and $\delta\,|a - f(x)| - \frac{1}{2}\delta^2$ is a linear form of identified larger value using "δ " delta parameter which sets the value for MAE and MSE through number of iterations to have absolute value. Huber loss computation in (3) lly,

$$HL_{\delta} = \begin{cases} (G^{(s)})^2 & \forall|G^{(s)}| \leq \delta \\ 2\delta \left|G^{(s)}\right| - \delta^2 & \forall \left|G^{(s)}\right| > \delta \end{cases} \tag{3}$$

By differentiating between quartic and linear, the value of G is expressed in (4,5)

$$\frac{d}{dG}G^2 = \frac{d}{dG}|G|$$

For x < 0,

$$G = -\frac{1}{2} \tag{4}$$

For $x > 0$

$$G = \frac{1}{2} \tag{5}$$

To equalize both functions, differentiation was performed to ensure the junction point of quadratic and linear Huber losses. Thus, the outlier estimators of the k-means algorithms are performed using Huber loss functions and provide a strong bond between the centroid and data points [23].

B. EBiGAN

Enhanced Bidirectional Generative Adversarial Network (EBiGAN) is an extended version of the BiGAN. An additional discriminator was integrated with the BiGAN to differentiate the actual encoded data from the latent space and ensure the quality of the generated samples. The principle of EBiGAN is to improve the performance model, provide an improvised latent space, and stabilize the training of the generator and encoder. To maintain regularization, a realistic sample is generated from the generator and encoder, which are highly reliable for handling noisy data. It comprises a generator, discriminator, encoder, discriminator one and loss functions. Objective function of the trained EBiGAN in (6)

$$EBiGAN_{obj} = [min(G, E)\,, max\,(D, D_1)V\,(D, D_1, G, E)] \tag{6}$$

where the $V(D, D_1, G, E)$ is defined in (7),

$$V\,(D, D_l, G, E) = E_{a\sim pa}/E_{b\sim pE(\cdot/a)}[logD(a, b)]/ \\ \times \{(logD(a, E(a)) \mid logD_1(E(a), b)\}$$

$$+ E_{b\sim pb}\left[E_{a\sim pG(\cdot|b)}[log(1-D(a,b))]\right] \times \{log(1-D(G(b),b))\} \tag{7}$$

where G is the generator, E is the encoder, D is the discriminator and D_1 is the discriminator one, a is the real data; b is the latent space; (a, E(a)) is the encoded sample, (G(b), and b) are the generated sample and loss functions are generator loss, discriminator loss and discriminator one loss, respectively. The latent vectors from the latent space as input to the generator produce a generated sample, and the real data to the encoder produces encoded data processed by the discriminator to separate the actual and generated data. The D_1 has an input from the latent space and an encoded sample for differentiating the real and fake encoded samples to correlate the relationship between the latent space and the encoder. The primary advantage of the discriminator one is that it improves the quality of encoded samples [24].

C. DNN-PI

The Deep Neural Network comprises an input layer, two hidden layers, and an output layer with a hyperparameter learning rate, dropout, and activation function, with Gaussian optimization. Each layer in the network contained nodes that were interconnected using weights. The nodes in the input layer represent features of the input data. The input data to the hidden layer comprise sequential weights to the node, and the output layer is the data with weights in the node. Assigning weights to all nodes of the three layers and adding biases to the weights, produces flexibility in the model.

However, challenges associated with DNN include overfitting, computational complexity, hyperparameter tuning, and limited transferability. Among these challenges, hyperparameter tuning is the most complex. Tuning the DNN hyperparameters using Bayesian optimization is based on the Gaussian process optimization (GPyOpt) tool, which also models the objective function and is trained for several iterations to determine the best hyperparameters. This benefits from the continuous search space, and the performance of the model is high. Nevertheless, Bayesian optimization is challenging for high-dimensional data; the non-convexity and non-smoothness of the Gaussian process make the model complicated and discontinuous. This can be solved using an acquisition function for Bayesian optimization [25]. The Probability of Improvement (PI) acquisition function for Bayesian optimization in (8) is proposed for tuning the learning rate hyperparameter of the DNN to solve non-convex and non-smooth problems, which are primarily set for high-dimensional data.

$$PI(y) = P(f(y)f(Best\ observed\ values)) = f_{-\infty}^{\mu(y)}\Phi\left(\frac{f(best)-\mu(y)}{\sigma(y)}\right) \tag{8}$$

where PI(y) is the probability of the Improvement acquisition function at point y; f(y) is the function of the true value of the objective function at y, f (best-observed value) is the observed value of the objective function, μ (y) is the mean value provided by the surrogate model, $\sigma(y)$ is the standard deviation provided by the surrogate model and ϕ is the cumulative normal distribution. The optimization process was triggered to investigate the location where the surrogate model indicated a high possibility of improvement over the present best value via the PI acquisition function.

IV. RESEARCH OBSERVATION

RQ1: How does the improved PCA method handle timestamped user activity data points to balance information retention and computational efficiency for anomaly detection by insiders in IoT enabled institutions?

RQ2: Incorporating clustering techniques into the IoT infrastructure influences the accuracy and efficiency of outlier detection within the reduced feature space and defines the impact of the interpretability of identified outliers.

RQ3: How does the proposed data augmentation model effectively address data imbalance problems, and how does it prove that the generated adversarial samples are similar to the actual samples of insider threat data?

RQ4: How can DNN-PI be optimized to enhance the detection of insider threats and achieve an increased detection rate while minimizing false alarm rates?

RQ5: State the security provided by the implementation of the proposed model in real-time infrastructure and its critical limitations

A. PROPOSED METHODOLOGY

The proposed model is framed as an ITD with a high detection rate and a minimal false alarm rate. It is composed of three stages: (1) Extraction of user functionality-based samples using IPCA and scenario-based clustering using outlier estimators of K-means, (2) Enhanced Bidirectional GAN for solving imbalanced data problems and (3) Probability of Improvement the acquisition function for learning rate hyperparameter tunning of the DNN model to achieve better accuracy with minimal false alarm rate. The framework of the proposed detection model is illustrated in Fig.1.

1) STAGE 1: USER FUNCTIONALITY-BASED SAMPLES USING IPCA AND SCENARIO-BASED CLUSTERING USING OUTLIER ESTIMATORS OF K-MEANS CLUSTERING

a: DATASET & PREPROCESSING

IoT enabled institutional log records were collected, which including user activities with timestamps. Carnegie Mellon University's (CMU) Computer Emergency Response Team (CERT) institutional log records of insider threats [26] were correlated with the five-layered IoT architecture, as shown in Fig.2.CMU CERT log records consist of logons, files, emails, devices, and HTTP. The five layers of IoT architecture are perception, transport, processing, application and business. The portable PC in the perception layer holds the records of devices accessed by users; the transport layer has a gateway for sending and receiving mail and data; the records of web services are presented in the processing layer; the user files

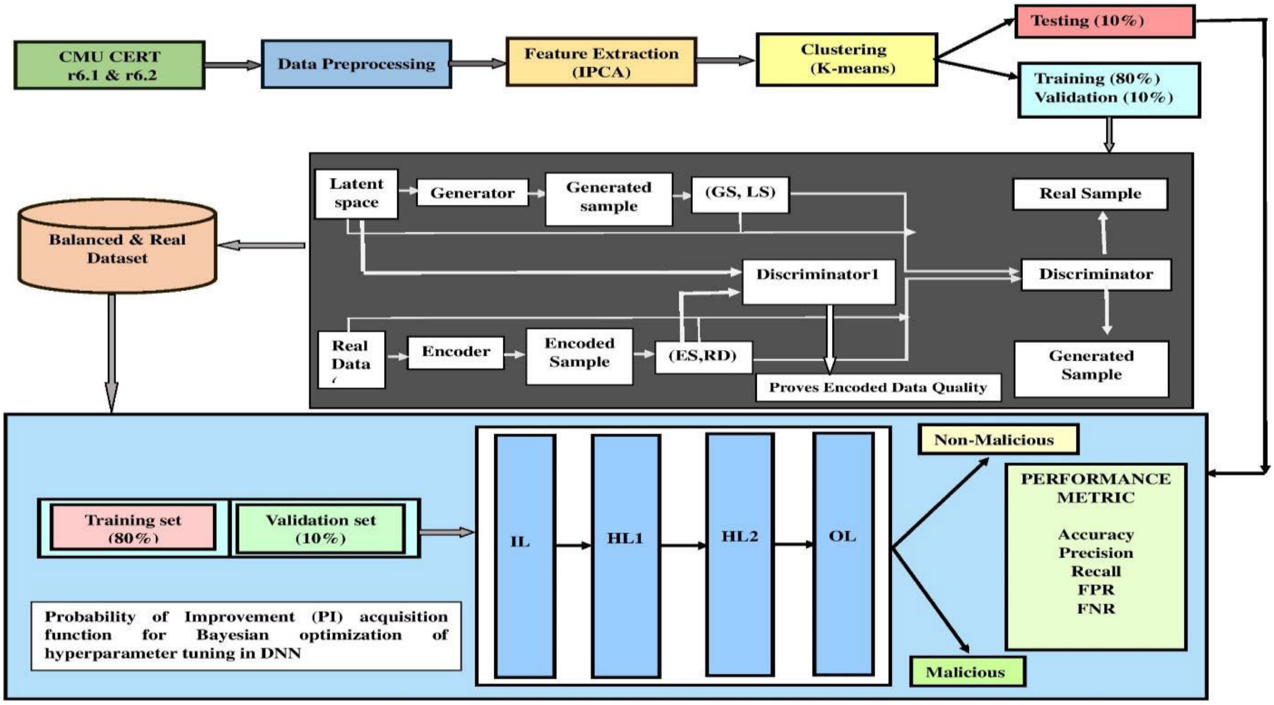

FIGURE 1. Proposed insider threat detection model architecture.

```
Algorithm 1 Data Preprocessing
    Input:CMU CERT r6.1 & 6.2 (logon, device, http, email, file→ single homogenous file)
    Output:Insiders, Non-insiders

    Data Preprocessing

    Begin
    Convert categorical, ordinal → numerical
    Ordinal encoding $(0, 1, 2, 3, \ldots\ldots\ldots n)$
    Encoded samples → robust scaling
    Robust scaling $R_s = \frac{R-R_{median}}{Iqr}$
    End
```

are enabled in the application layer; and the user accessing records are formatted in the business layer.

The log records were converted into a single homogeneous file for insider threat detection, that consisted of both categorical and ordinal values. The initial step of ITD involves data preprocessing, including data cleaning, data reduction and transformation. The timestamps are mined in terms of hour, day _of _week, day_of_ month, month and year and ordinal encoding is then used to convert all categorical values into numerical values [27].

b: FEATURE EXTRACTION

Improved Principal Component Analysis (IPCA) was used as a feature engineering technique to extract user functionality-based samples and outlier-estimators of the k-means clustering technique to cluster scenario-based user functional data within IoT infrastructure. IPCA is a dimensionality reduction technique that reduces large samples to smaller collections without disturbing the underlying structure of a dataset. The PCA steps for removing the user behavior samples are as follows,

Step 1: The ordinally labelled data are scaled using robust scaling, making the detection model more robust. It was computed between the median and interquartile range (Iqr) in (9)

$$R_s = \frac{R - R_{median}}{Iqr} \tag{9}$$

where R_{median} is the median, and Iqr is the interquartile range

Step 2: Find the covariance matrix for the normalized data to identify the highly correlated sample. The covariance matrix is given by (10)

$$Cov_a X_m = \lambda_m X_m \tag{10}$$

where $Cov_a X_m$ is the covariance matrix, X_m is the eigenvector with an eigenvalue λ_m

Step 3: Eigenvalues and eigenvectors of the covariance matrix are computed. Eigenvectors are defined as the principal components of the underlying data structure. The eigenvectors with a more significant value are the essential principal components, and those with a small value are the least critical principal components

Step 4: The number of principal components is selected based on the Kaiser criterion, which has an eigenvalue as

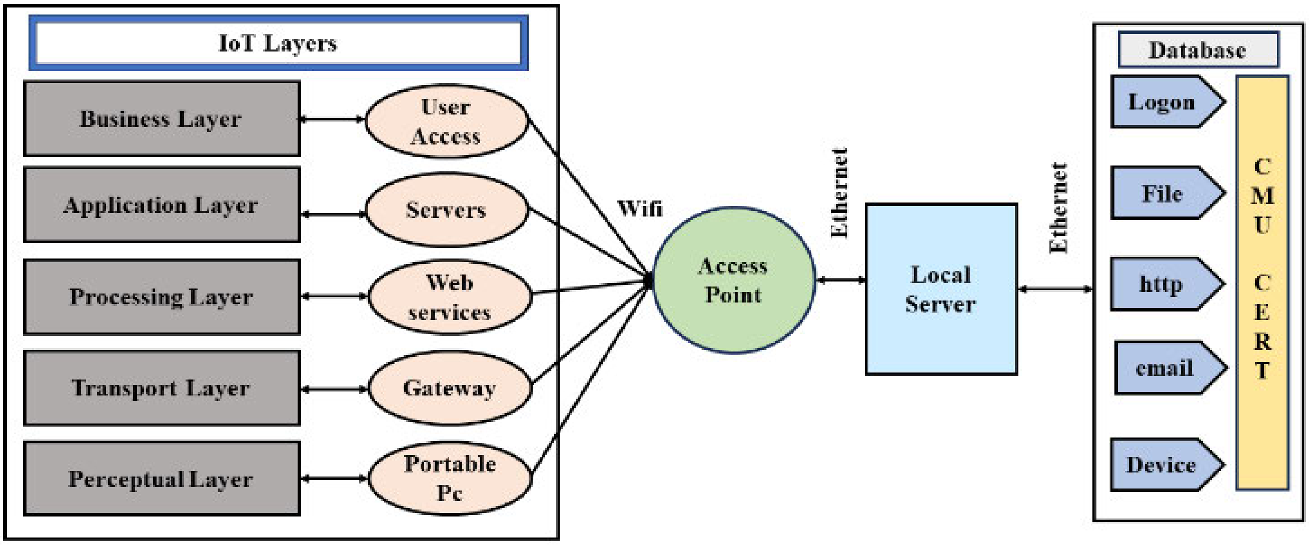

FIGURE 2. Log records from five layers of IoT.

Algorithm 2 Feature Extraction - Time based user behavior sample using IPCA

```
Begin
    Compute Covariance matrix (time stamps, activity(user))
    $Cov_a X_m = \lambda_m X_m$
    Compute eigen values and eigen vectors
        eig val → $AX = \lambda X$ &
        eig vect → $(A - \lambda I) X = 0$
        eig val >1 (Kaiser Criterion) // to find no. of
        principal components
    find the largest principal component (user
    behavior samples) // reduce dimensionality//
    ignore the smallest eigenvectors
End
```

Algorithm 3 Clustering-Outlier resistant k-means clustering

```
Begin
    K-means clustering
    Initialize centroid C // as scenario (IT admin)//Select k
    Find distance between centroid and user //centroid as
    one scenario & user who are belongs to the scenario//
    Compute Euclidean distance
    $ED = \sqrt{\sum_{x=1}^{n} (a_x - b_x)^2}$
    Cover nearby user with same scenario as cluster
    Fix iteration point
    Round as per iteration with centroid
    Form cluster//once reached the iterations//
End
```

greater than one for each attribute. This holds factors which have more variance and ignores model overfitting

Algorithm 4 Data augmentation - EBiGAN

```
Begin
G (w, b), D (w, b), E (w, b), $D_1$(w,b)
For Training
iteration i → real data = sample real data_(batch) ()
    latent space = sample latent space $\boldsymbol{LS_s}$ ()
    encoded data= encoded $\boldsymbol{E_{rd}}$(real data (batch))
    generate attack sample $\boldsymbol{G_{as}}(\boldsymbol{eLS_s})$
        dis(1) = $\boldsymbol{E_{rd}}(\boldsymbol{LS_s}())$
        $\boldsymbol{D_{loss}}$ = -log [Dis ($\boldsymbol{E_{rd}}$)-log (1-Dis ($\boldsymbol{G_{as}}$)]
        $\boldsymbol{G_{loss}}$ =-log (Dis ($\boldsymbol{E_{rd}}(\boldsymbol{LS_s})$))
        $\boldsymbol{E_{loss}}$ = || LS-En(Ge($\boldsymbol{LS_s}$))||$^2$
    /* process $\boldsymbol{D_{loss}}$, $\boldsymbol{G_{loss}}$, $\boldsymbol{E_{loss}}$parameter update */
Proceed iteration
End
```

Step 5: The feature vector matrix is constructed using larger principal component values, and smaller principal component values are neglected to reduce data dimensionality without changing the underlying structure of the data

Step 6: Coordinate the standard data to the principal components by multiplying the transpose standard matrix by the feature sample matrices [28]. Once dimensionality is reduced, the outlier estimator of k means is applied to group similar samples. The value of k was determined based on five different scenarios of the dataset [29].

2) STAGE 2: ENHANCED BIDIRECTIONAL GAN FOR SOLVING THE IMBALANCED DATA PROBLEM

a: DATA AUGMENTATION (EBIGAN)

The EBiGAN structure comprises a generator, discriminator, encoder and discriminator one with loss functions. The

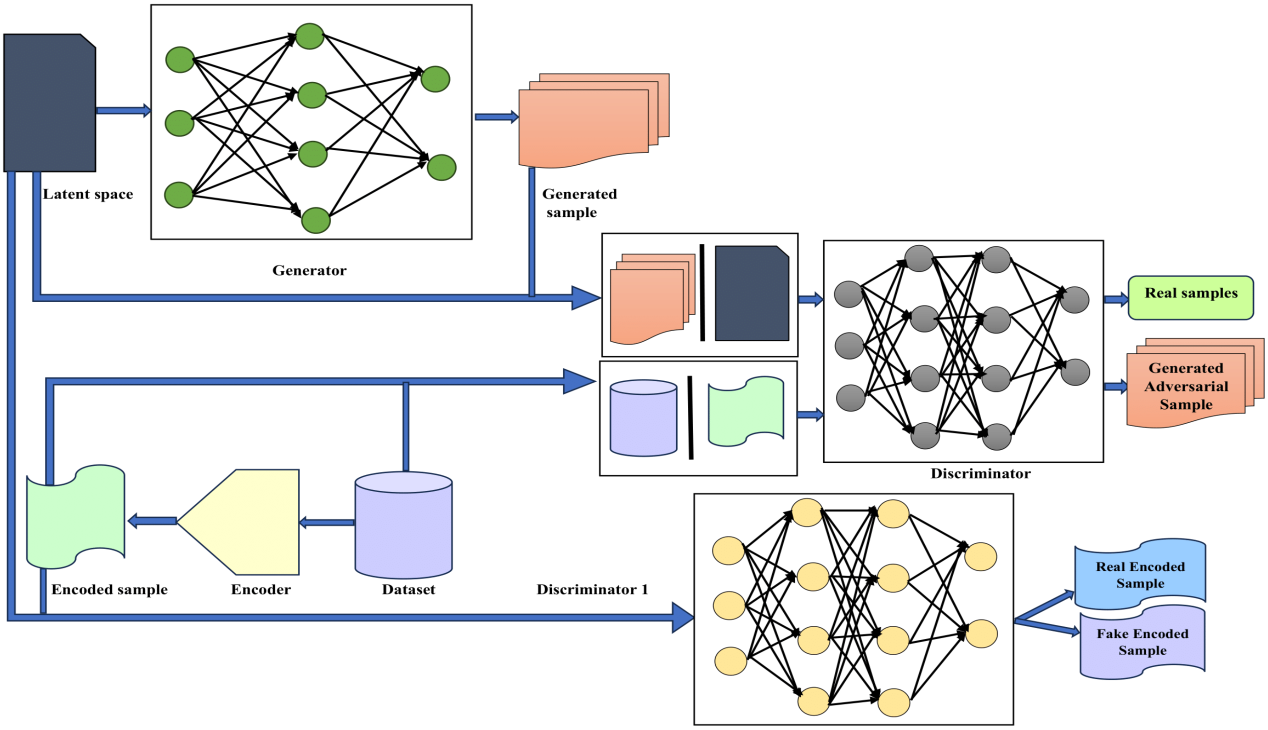

FIGURE 3. EBiGAN architecture.

TABLE 1. Computation of probability of improvement acquisition function.

Validation Accuracy	Loss
Step1: Estimate the mean (μ_{acc}) and standard deviation (σ_{acc}) for learning rate(lr) using the Gaussian process (GP)	**Step 1:** Estimate the mean(μ_{lss}) and standard deviation (σ_{lss}) for learning rate using the Gaussian process (GP)
Step 2: Calculate Z score for validation accuracy	**Step 2:** Calculate Z score for loss
$Z_{acc} = \frac{Pr_{acc} - \mu_{acc}}{\sigma_{acc}}(lr)$	$Z_{loss} = \frac{Pr_{lss} - \mu_{lss}}{\sigma_{lss}}(lr)$
Where Z_{acc} is z-score of accuracy; Pr_{acc} is present observed best accuracy value; μ_{acc} is mean accuracy;σ_{acc} is SD; lr – learning rate	Where Z_{lss} is z-score of loss; Pr_{lss} is presently observed as the best loss value; μ_{lss} is mean loss;σ_{lss} is SD; lr – learning rate
Step 3: To compute PI, Take cumulative distribution function (CDF) from Normal distribution $PI_{lr} = P(Z_{acc} > 0)$ & $P(Z_{lss} > 0)$ = [(1- $\Phi(Z_{acc})$. $(1 - \Phi(Z_{lss})$]	

latent space contains random noise, which is converted into latent vectors and forwarded to the generator that produces adversarial samples generated in Layer 1. The encoder inputs the actual data and delivers the encoded data in layer 2. The discriminator separates the actual samples from the generated models and encoded samples, which are represented by a combination of both layers [30]. Then, the discriminator one has concatenated input from the encoded data and latent space samples to evaluate the correspondence between the data, and it defines how the encoder performs the mapping of actual data into latent space, as shown in Fig.3. shows the EBiGAN architecture. The discriminator one ensures the quality of the encoded samples.

3) STAGE 3: PROBABILITY OF IMPROVEMENT (PI) FOR HYPERPARAMETER TUNING OF DEEP NEURAL NETWORK FOR ITD (DNN-PI)

The PI acquisition function of Bayesian optimization is used to tune the learning rate hyperparameter of the DNN to improve the performance while detecting insider threats in IoT-enabled institutions. The main challenges in Bayesian optimization for DNN hyperparameter tuning are the non-convexity and non-smoothness of the objective function, which affects the performance of the DNN when working with various datasets.

The PI acquisition function enumerates the probability of validation accuracy and loss of learning rate to demonstrate

Algorithm 5 Insider Threat Detection (DNN-PI)

```
Input:(Balanced dataset + (training dataset (80%) + validation dataset (10%)) & Testing dataset (10%)
Output:Binary classification (0→ malicious; 1→ non-malicious)

Begin
Initialize DNN hyperparameters
Frame DNN layers
    Input layer ($IL_1$) → { ($w_1, b_1$) ⇒ [ $IL_1$.size, $HL_1$.size] Hidden layer $\left.\begin{matrix}HL_1\\ HL_2\end{matrix}\right\}$ → { ($w_2,b_2$) ⇒ [$HL_1$.size, $HL_2$.size]}
    Output layer ($OL_1$) → { ($w_3, b_3$) ⇒ [ $HL_2$.size, $HL_2$.size]}
    Initialize training
        Update forward propagation
        for epoch (200)
        for batch size (64)
        $w_1, b_1$
        $w_2, b_2$
        $w_3, b_3$
        Compute cross entropy loss
        Deploy Back propagation
With stochastic gradient descent update w, b
Compute model performance
Initialize GpyOpt
 Set up iterations (i=5)
 Hyperparameter (learning rate)
 Obj.fun (val.acc,loss)
 Updated (lra) ⇒Bayesian opt (evaluations (PI))
Fine-tune DNN model with updated hyperparameter
 Train & Test updated DNN⇒ Detect insider
 Classify 0 (Malicious)& 1(non-malicious)
        Update DNN-PI performance
End
```

TABLE 2. Attributes of cmu cert log records in IoT enabled institutions.

Log records	Features
Logon	ID, User ID, PC, Date, Activity
File	ID, UserID, PC, Date, Filename, Activity
Email	ID, User ID, PC, Date, Mail To, Cc, Bcc, Mail,From, Attachments, Activity
HTTP	ID, User ID, PC, Date, http link, Activity
Device	ID, User ID, PC, Date, File Structure, Activity
Psychometric	ID, User ID, PC, Date, O, C, E, A, N

TABLE 3. Preprocessed timestamps attributes and its description.

Attributes	Description
hour	Users total working hours in a day
day_of_week	Users total working days in a week
day_of_month	Users total working days in a month
month	Indication of specific month of corresponding day and week
year	Indication of specific year of corresponding week and month

performance improvement in the present observed value. The Gaussian process, which is a surrogate model of Bayesian optimization, was implemented in the DNN using the GPyOpt tool of the Python library [31]. The aim of DNN-PI is to obtain the best learning rate to improve the performance of a DNN in detecting insider threats in IoT-enabled institutional log records [32].

The steps in Table 1 were used to compute the PI acquisition function for the DNN learning rate to maximize validation accuracy and minimize loss values. Thus, the steps defined in the proposed model were used to obtain learning rate of DNN for insider threat detection in IoT-enabled institutional log records. In Bayesian optimization, the objective function is enhanced using the probabilistic model to obtain the best solution in which the acquisition function improves the accuracy and decreases the loss reduction of the objective function in the holds a new set of hyperparameters for the objective function evaluation, and the data points are added to the DNN model based on iterations and the model is updated. Finally, the fine-tuned optimal learning rate hyperparameter is determined. The model was then trained and validated on an insider threat dataset to achieve better accuracy.

V. EXPERIMENTATION AND RESULTS

A. EXPERIMENTAL SETUP

The proposed model was implemented using Python 3.5 with the Keras library and TensorFlow. The experiments are carried out in an Intel core i5 processor with 8 GB RAM, and 64-bit processor in the Windows 10 operating system, and the dataset used is CMU CERT r6.1 & r6.2 dataset.

B. DATASET DESCRIPTION

The CMU CERT insider threat dataset was released from r1 to r6.2. All six releases consisted of the following folders: logon, device, email, HTTP, psychometric and file. User activities were recorded in terms of id, userid, timestamps, PC number, and activity, which are common attributes in all log records. The proposed model was implemented in r6.1 & r6.2 of five scenarios with 3995 users and five insiders. The releases cover the LDAP for 17 months from 01-11-2009 to 01-05-2011, including the attributes of user designation, projects, business, functional, department, and team. The features of all the log records are listed in Table 2. The log records of each release were constructed as a single homogenous file and then separated into training, testing and validation sets for insider threat detection.

C. DATA PREPROCESSING

The single homogenous file is pre-processed by uprooting the timestamps in the order of hour, day_of_week,

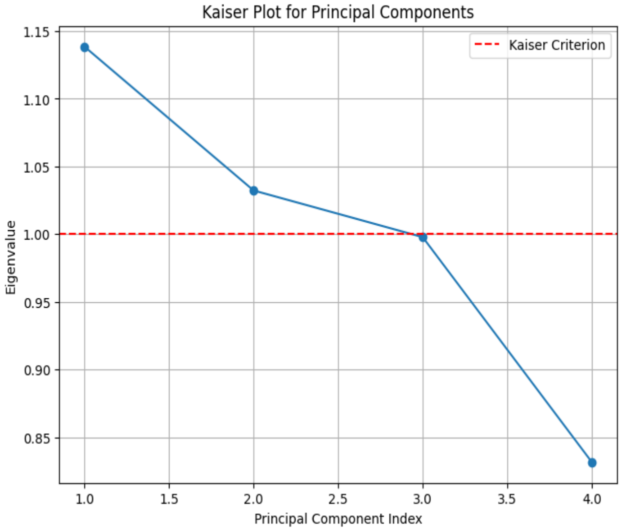

FIGURE 4. Kaiser criterion to determine number of principal components.

day_of_month, month and year. A description of the extricated features of the timestamps is provided in Table 3. Ordinal encoding is then applied to attributes id, user-id, pc, and activity, which converts the data from hour to years of uprooted timestamps.

D. RESEARCH OBSERVATIONS

RQ1:How does the improved PCA method handle timestamped user functional data points to balance information retention and computational efficiency for anomaly detection of insiders in IoT infrastructure-based institutions?

IPCA is used as a feature engineering technique to extract user functionality-based samples to reduce the dimensionality of the data deprived of the original data. Robust scaling is computed using time, functionality units and user activity to normalize, centralize and notify dissimilar functions of a user's data, which makes it less sensitive to outliers. From the obtained time-dependent user functionality of centralized data, a covariance matrix is constructed and eigenvalue decomposition is computed by taking the eigenvalue and eigenvector with time and the user functionality unit. The number of principal components was determined using Kaiser criterion. When an eigenvalue is more significant than one, the corresponding attributes are retained as principal components. Here, the number of principal components of the insider threat data is declared as two. The user functional unit and user id are taken as the two principal components. The computation of the number of principal components performed by the eigenvalue and principal component index is shown in Fig.4.

The corresponding larger eigenvector with its eigenvalue is marked as an important principal component, and small eigenvectors, which are not included for further processing, are ignored. The functional units of the user are larger eigenvectors of the principal components, without disturbing the underlying structure of the data. The interpretation of the eigenvalue denotes the time duration of the user's activity and, depending on the principal component of the user's functional unit, the eigenvector characterizes the route of the user's functionality. The principal components extract important user functionality-based samples, ensuring that the improved PCA can handle balanced information retention. The performance of eigenvalue decomposition and IPCA, especially in CMU data, processes the data dependent on the "functional_units" attribute of insider threat test data, proving the data scalability and computational efficiency within the IoT infrastructure.

RQ2: Incorporating clustering techniques in IoT infrastructure influences the accuracy and efficiency of outlier detection within the reduced feature space and defines the impact of identified outlier's interpretability

Outlier estimators of k-means clustering were applied to group the extracted time-user functionality-based samples depending on the scenarios listed in Table 4. Owing to the former acquaintance in the dataset, the five specific scenarios of insider threat data involve clustering the samples. Initializing the value of k=5 indicates the number of clusters to be formed in the extracted user functionality samples depending on the scenarios of the insider threat dataset. Thus, the implemented clustering technique ensures the accuracy and efficiency of insider detection within the reduced feature space and defines the impact of the interpretability of identified outliers. Fig.5. represents the IPCA with outlier estimators of K-means for the samples of r6.1 and r6.2, where the IPCA shows the extraction of the user's functionality-based samples, which achieves dimensionality reduction and k=5 is represented as the centroid (red color) of each cluster that indicates the scenarios. It defines the user's functional data related to a particular scenario that falls within the cluster of insider threats within the institutions of IoT infrastructure. For instance, scenario S1 as a centroid and joining all the user's functional data of S1 as a single group is performed iteratively by computing the Euclidean distance between users. The cluster S1 is formed by iteration, and continues until the nearest user is formed. The further cluster formation procedure of S1 continued for the remaining four scenarios, S2, S3, S4, and S5. Scenario-based user functional data clustering identifies the intrinsic form and structure of insider threat data in a practical and understandable manner. Scenario-based clustering in the obtained user functionality indicated a reduced feature space with important outlier data. Identifying user functionality with different scenario-based data distributions in the feature space provides to a simple method for further insider detection analysis within the IoT infrastructure.

Outlier-resistant k-means clustering for grouping the scenarios-based user functionality sample confirms the identification of outliers through the scenarios of each cluster with its local characteristics, which can improve the detection rate and accuracy of the insider threat detection model.Scenario-based clustering has increased the efficiency of insider

TABLE 4. IoT scenarios for log records and its description.

Scenarios	Description
S1	Earlier no use of removable device (or) logging system later in office hours &Uploading information through "wikileaks.org" and escape in short time from the company
S2	Skimming job sites and employment conciliatory move from a contender & user's leave company with fingerprints after ransack
S3	System admin become exasperated downloads a key and takes fingerprint then move's the collected data to organization head.
S4	One user logging into another user's system and look for some confidential files or documents and forward that to his/her personal mail. This activity will happen frequently
S5	A user from team ruined the uploaded termination documents in drop box and used it for personal reason

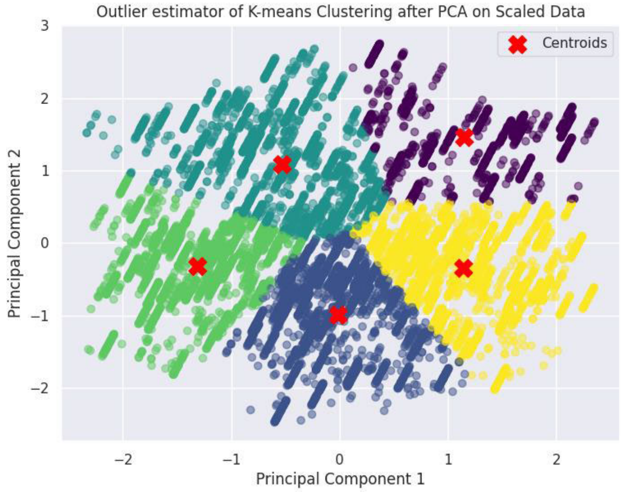

FIGURE 5. IPCA with outlier estimators of K-means clustering.

detection by reducing the time complexity and dimensionality of user functionality.

RQ3: How the proposed Data augmentation model effectively address the data imbalance problems and how does it prove the generated adversarial samples are similar to the real samples of insider threat data?

User functionality-based samples ensure that the insider threat dataset is highly imbalanced, with most normal samples and a minority of malicious samples. In this research, the Enhanced Bidirectional Generative Adversarial Network (EBiGAN) data augmentation technique is used to generate malicious samples to generate a balanced dataset that improves the performance while detecting insiders within the IoT infrastructure. The importance of EBiGAN lies in defining the quality of the encoded data from actual data to demonstrate the stability and flexibility of the model. The parameters of EBiGAN are listed in Table 5. The extracted user-functionality-based samples were partitioned into 80% training,10% validation, and 10% testing samples. It contains a majority of non-malicious samples and a minority of malicious samples. Training samples are present in the discriminator and encoder to solve the class imbalance problem. The latent space of the EBiGAN maps the latent vector onto the generator, which then generates random adversarial samples. The encoder maps the input data to the discriminator from the extracted user functionality samples, which differentiates the generated samples from the functional data of user. The discriminator one was used to map the latent samples and encoded user functionality samples to determine the quality of the encoded samples.

TABLE 5. Parameters of EBiGAN.

EBiGAN Parameters	EBiGAN Parameter Description
Generator **Discriminator** **Discriminator one**	**Input layer** – One layer (three neurons) **Hidden layer**-One layer (four neurons) **Output layer**- One layer (two neurons) **Activation Function**- ReLU in Hidden Layer &Sigmoid in output layer **Loss**- Generator loss, Discriminator loss, and Discriminator one loss
Encoder	**Input layer**- One layer (three neurons) **Hidden layer**- One layer (four neurons) **Output layer**- One layer (two neurons) **Activation Function**- ReLU in Hidden Layer &linear in output layer **Loss**-Encoder loss
Batch size	64
Learning rate	0.002
Epochs	200
Optimizer	Adam

The generated random adversarial samples of minority class balance the user's functional data of training samples. Balanced training and validation data are incorporated into the insider threat detection model. The testing data were directly forwarded to the DNN-PI detection model. The collected samples of each phase are coordinated to the next stage of the proposed model (i.e.) the extracted user functionality samples of phase 1 are taken as input to the data augmentation

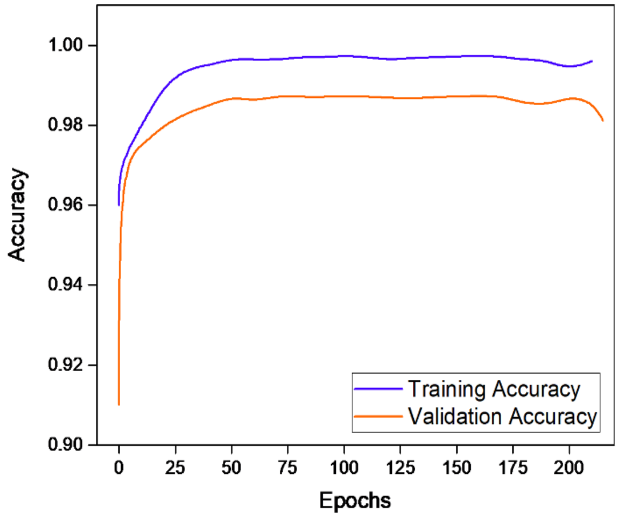

FIGURE 6. Accuracy Vs Epochs of proposed model.

model of EBiGAN in phase 2, and the balanced training samples of phase 2 are given as input to the DNN-PI of the insider threat detection model.

RQ4: How can DNN-PI be optimized to enhance the detection of insider threats and achieve an increased detection rate while minimizing false alarm rates?

A Deep Neural Network with Probability of Improvement (DNN-PI) was used to detect insider threats in the balanced insider threat data of IoT enabled institutions. In DNN-PI, the PI acquisition function of Bayesian optimization fine-tunes the learning rate hyperparameter of the DNN to improve the performance of the model. The structure of DNN-PI is composed of one input layer with three neurons, two hidden layers where the first layer has five neurons; the second layer has four neurons, and one output layer has two neurons. The input layer holds balanced training, validation, and original testing samples. Two-batch normalization hidden layers are used to convert the user's functional data in the form of anomaly identification. The user's functional data of hidden layers is fed into the output layer to identify the anomaly. It performs binary classification to detect insiders by classifying the samples as malicious or non-malicious. The hyperparameters of the DNN-PI were a learning rate lr of 0.002, small step size resulting in less convergence with a batch size of 64 for memory and processor speed, and L2 regularization enhancing the loss function with 200 epochs. It uses the Gaussian process as a surrogate model for the objective function (validation accuracy and loss) of the learning rate hyperparameter tuning in both training and testing models.

The fine-tuned DNN-PI model was trained to detect the insider threat in the extracted user functionality-based samples, achieving an increased detection rate and a minimized false alarm rate. The accuracy of the detection model reaches a maximum of 0.98 in 200 epochs, while training and validating are shown in Fig.6.

The loss of the detection model reaches 0.03 in validation and 0.04 in training for 200 epochs in Fig.6.The proposed detection model has improved accuracy and reduced loss owing to the presence of an acquisition function in the Gaussian process with parallel optimization settings. This ensured that the proposed detection model had a greater detection accuracy with fewer losses for 200 epochs. The importance of PI during optimization is a sense of balance between exploration and exploitation for the learning rate in the detection of insider threats.

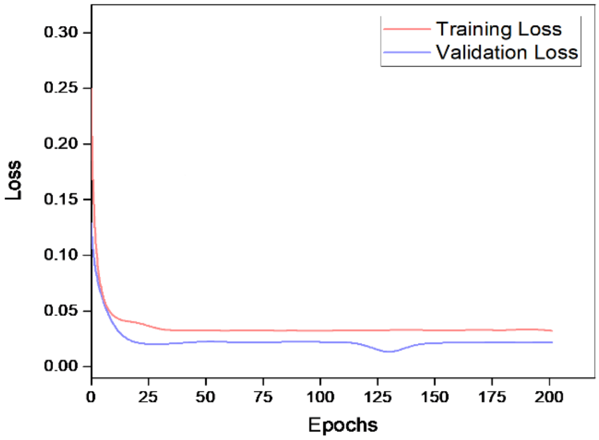

FIGURE 7. Loss Vs Epochs of proposed model.

RQ5: State the security provided by the implementation of the proposed model in real-time infrastructure and its critical limitation

When the proposed model is implemented in different types of IoT infrastructure institutions, it secures the environments from unauthorized access to devices and mail, physically interfering with the IoT devices, providing high security while configuring the access controls, and avoiding negligent insiders. However, while deploying the proposed model faces significant limitations, such as noise robustness and computational complexity, it provides an enhanced secure IoT infrastructure to institutions. The proposed model only concentrates on securing institutions from insider threats, although it is not aware of other external cyber attacks.

E. DISCUSSION

The performance of the proposed detection model is validated in terms of accuracy, precision, and detection rate with a relatively minimal false alarm rate in (11-15). The effectiveness of the proposed model depends on the efficiency, and, hyperparameter selection of the surrogate model. The exploration phase of various learning rates and exploitation was balanced to improve the performance of the DNN model by deploying the acquisition function of Bayesian optimization.

$$Accuracy = \frac{TP + TN}{(TP + TN + FP + FN)} \tag{11}$$

$$Precision = \frac{TP}{(TP + FP)} \tag{12}$$

$$Detection\ Rate\ (DR) = \frac{TP}{(TP + FN)} \tag{13}$$

TABLE 6. Comparing the performance of proposed model with existing insider threat detection models.

Neural Network Models	Dataset CMU CERT	Performance Metrics				
		Accuracy	Precision	DR	FPR	FNR
LSTM+AUTOENCODER	r4.2	0.942	0.925	0.897	0.05	0.07
ADASYN+DNN	r4.2	0.963	0.945	0.903	0.06	0.08
RNN	r4.2	0.932	0.921	0.902	-	-
CWGAN-GP+1DCNN	r4.2, r5.2	0.961	0.947	0.946	0.04	0.05
CGAN+1DCNN	r4.2	0.821	0.749	0.875	-	-
Hybrid Learning (BiLSTM+FFANN+SVM)	r4.2	0.887	90.857	0.844	-	
ResHybnet (GCN+CNN)	r4.2	0.926	0.915	0.928	-	-
BiLSTM+Hypergraph+CAE	r6.2	0.093	-	0.925	0.05	-
SPCAGAN+Hybrid BNN	r4.2, r5.2	0.904	0.895	0.695	-	-
ITDBERT	r4.2	0.950	0.930	0.915	-	-
PROPOSED MODEL	**r6.1, r6.2**	**0.988**	**0.978**	**0.967**	**0.03**	**0.04**

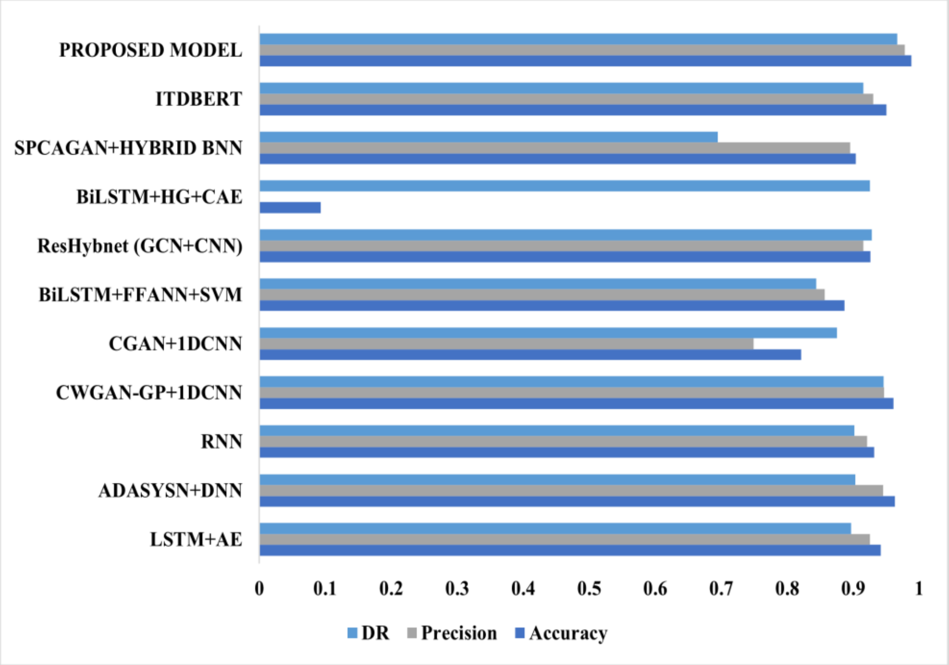

FIGURE 8. Accuracy, precision, Detection rate results of proposed model compared with existing model.

$$\textit{False Positive Rate (FPR)} = \frac{FP}{(TN + FP)} \tag{14}$$

$$\textit{False Negative Rate (FNR)} = \frac{FN}{(FN + TP)} \tag{15}$$

The proposed detection model was compared with existing deep learning models such as the Long Short-Term Memory-Autoencoder [27], which had increased FNR (0.07) affects the false alarm rate, Adaptive Synthetic sampling technique based Deep Neural Network (ADASYN-DNN) [9] has high false alarm rate (0.06) of FPR and (0.08) of FNR due to less threshold value of model, Recurrent Neural Network (RNN) [33] includes Gated Recurrent unit classifier as binary classifier has (0.902) as DR for imbalanced data fails to detect the accurate insiders. Conditional Wasserstein Generative Adversarial Network-Gradient Penalty with One-dimensional Convolutional Neural Network (CWGAN-GP -1DCNN) [34] had reached the maximum false alarm rate (0.04) of FPR and (0.05) of FNR, however, the obtained result had less DR. The Conditional Generative Adversarial Network with One-dimensional convolutional neural network [35] had a lower performance for DR (0.875) for balanced data. The hybrid learning model is composed of Bidirectional LSTM for feature extraction, Feed Forward Artificial Neural Network for feature selection, and Support Vector Machine for insider detection (BiLSTM-FFANN-SVM) [36], which achieved a lower DR (0.844), and the residual hybrid network with a graph convolutional network combined with convolutional neural network (Reshybnet-GCN-CNN) [37] which

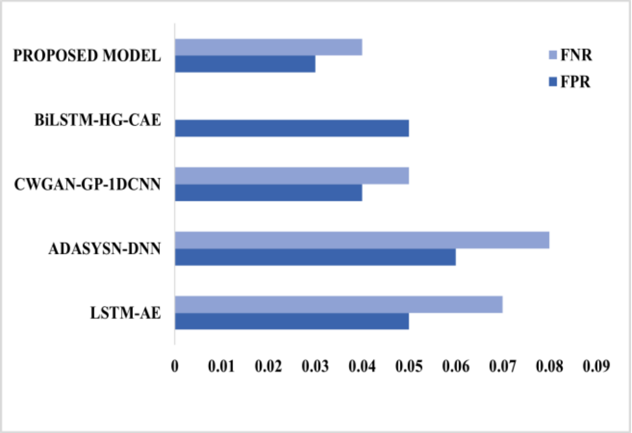

FIGURE 9. False alarm rate results of proposed model compared with existing models.

achieved the DR (0.928).However, the computation of the false alarm rate was not determined. The bidirectional LSTM for dimensionality reduction, hypergraph to achieve less FPR, and cascaded autoencoder to detect insider (BiLSTM-HG-CAE) [38] achieved a lower FPR (0.05), although it affected the model training speed. The similarity factor-based principal component analysis integrated with the Generative Adversarial Network (SPCAGAN) and Hybrid Bayesian Neural Network model [39] achieved a lower DR (0.695). The insider threat detection model of Bidirectional Encoder Representations from transformers with BiLSTM [40] had reached (0.915) DR; however, the data are purely imbalanced, which affects the performance. Table 6 lists the overall comparison of the proposed model, which was ensured by achieving a high DR with a low false alarm rate. The existing models are applied to the insider threat dataset of CMU CERT releases of r4.2 and r5.2; however, the proposed model is implemented in the latest releases of r6.1 and r6.2, which cover all five scenarios of a benchmark dataset.The proposed detection model had better accuracy (0.988), precision (0.978), and DR (0.967) for balanced insider data. Generally, accuracy defines the complete perfection of the proposed model while detecting the insider, and precision holds positive prediction values of accuracy that ignore false positive errors. DR is more significant than precision and recognizes the positive occurrences of existing insider threat detection neural network models, as shown in Fig.8. It achieves the minimum false alarm rate by analysing the FPR, which wrongly identifies the negative occurrences as positive occurrences in the rest of the compared existing insider threat detection methods with the proposed model shown in Fig.8. In insider threat detection, the FPR 0.03 is predicted owing to the fine-tuning of the DNN hyperparameter, where the model identifies the normal samples as malicious, as shown in Table 6 and depicted in Fig.9.

The proposed data augmentation technique of EBiGAN was compared with the one oversampling method called the Synthetic Minority Oversampling Technique (SMOTE) [10]. It produces synthetic samples of the minority class in the insider threat data. This was carried out by utterance among malicious samples of the minority class. The SMOTE procedure computes the Euclidean distance to find the nearest neighbors within the minority class. It produces synthetic samples, which are replicas of malicious samples and joining the samples with their neighbors. However, SMOTE has some significant complications regarding the insider threat dataset because it generates a smaller number of samples, which does not cover all the scenarios of the dataset, due to linear utterance, the generated sample does not have the resemblance of the original dataset and the major complex in handling high dimensional data because of identification of nearest neighbors.

TABLE 7. Comparison of smote and ebigan with an classifier oF DNN-PI.

	Training (80%), Validation (10%), Testing (10%)			
IDR	SMOTE+DNN-PI		Proposed Model	
	Accuracy	DR	Accuracy	DR
80:20 (raw data)	0.654	0.687	0.734	0.754
70:30	0.684	0.701	0.786	0.791
60:40	0.715	0.726	0.889	0.934
50:50	0.792	0.798	0.988	0.967

Based on imbalanced data ratio (IDR) values in (16), as the presence of samples in the majority of the normal class divided by the presence of samples in the minority class. SMOTE and EBiGAN are compared, and the performance of the classifier model ensures the identification of a better model.

$$IDR = \frac{Non - MaliciousS^{+}}{MaliciousS^{-}} \tag{16}$$

where S^{+} is the majority class and S^{-} is the minority class. In general, the majority class is always a non-malicious sample, and the minority class is malicious sample. When the IDR value is greater than or equal to one, there exist high level of class imbalance problem. Table 7 compares the four IDR values of SMOTE and EBiGAN with the proposed classification model. The classified IDR ratios (80:20, 70:30,60:40, 50:50) compute the accuracy and DR as performance metrics for the proposed model with SMOTE- generated samples.

The imbalanced data of the CMU CERT insider threat dataset was balanced using different data augmentation techniques. Recently, GAN [10] and CWGAN-GP [34] have to balance an insider threat dataset with a neural network model to detect insider threats [10] proposed a deep adversarial insider threat detection framework (DAITD), that includes the LSTM-autoencoder and GAN, which is a DAITD framework comprising three phases. In phase 1, the user behavior series is encoded into the LSTM - autoencoder, which shows critical user behavior representations. GAN was applied to analyze and generate anomalous user behavior samples to balance the imbalanced dataset. Then, the real and balanced

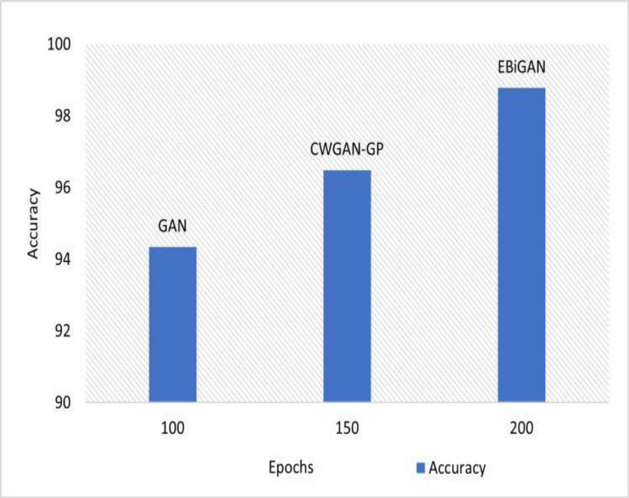

FIGURE 10. Proposed EBiGAN with existing data augmentation techniques used for insider threat data.

datasets are forwarded to the classifier to detect insider threats and classify malicious and non-malicious users. However, the performance of GAN poses instability and mode collapse problem while generating data. Reference [34] developed a conditionalized Wasserstein GAN with a gradient penalty for synthetically generating attack samples and deployed a multi-class classifier for insider threat detection. When the samples are generated with CWGAN-GP, non-targeted test time attack predictions are performed using a multiclass classifier model to detect insiders. However, the performance of the proposed model is computationally complex. To overcome the challenges of data augmentation and existing detection models for insider threats, the proposed detection model has better stability and improved generalizability. EBiGAN ensures that the generated adversarial samples improve the performance of the models and defines the quality of the real and encoded samples compared to GAN and CWGAN- GP.Fig.10. shows that EBiGAN is more accurate than in comparison with other GAN techniques.

F. ABLATION STUDY

To prove the modules of the proposed detection model, each module of the proposed model was evaluated for each scenario listed in Table 4 in terms of performance metrics namely accuracy, precision and detection rate DR. The four consecutive sets of experiments are shown in Table 8.

- A simple DNN is used to detect the anomalous behavior of insiders in the IoT infrastructure. This resulted in model overfitting and a lower overall accuracy of 0.75.
- To ignore the model overfitting, the data augmentation technique of BiGAN is used to balance the skewed data, which provides better results with an accuracy of 0.82 in insider threat detection. However, it is challenging to converge BiGAN and it cannot determine the quality of encoded and generated samples.
- The enhanced BiGAN is about the additional discriminator added to the BiGAN structure to notify the quality of the generated samples, which increases the accuracy by 0.86 when detecting the insider threat. However, the overall performance of the insider threat detection model was less.
- To increase the overall performance and data quality, EBiGAN was augmented with the probability of improvement (PI) acquisition function of the Bayesian optimized DNN to detect insiders within IoT enabled institutions.

TABLE 8. Abalation study of proposed model based on scenarios of insider threat data.

Training models	r6.1					
	Accuracy		Precision		DR	
Scenario1	**r6.1**	**r6.2**	**r6.1**	**r6.2**	**R6.1**	**R6.2**
DNN	0.756	0.768	0.733	0.758	0.709	0.755
BiGAN+DNN	0.789	0.797	0.749	0.787	0.712	0.770
EBiGAN+DNN	0.845	0.857	0.828	0.832	0.795	0.802
EBiGAN+DNN-PI	**0.926**	**0.932**	**0.917**	**0.921**	**0.898**	**0.918**
Scenario 2						
DNN	0.786	0.798	0.754	0.778	0.723	0.755
BiGAN+DNN	0.812	0.826	0.798	0.821	0.776	0.811
EBiGAN+DNN	0.856	0.865	0.837	0.845	0.816	0.827
EBiGAN+DNN-PI	**0.946**	**0.951**	**0.936**	**0.943**	**0.914**	**0.918**
Scenario 3						
DNN	0.755	0.755	0.724	0.737	0.701	0.714
BiGAN+DNN	0.798	0.812	0.776	0.805	0.752	0.794
EBiGAN+DNN	0.834	0.886	0.816	0.856	0.789	0.832
EBiGAN+DNN-PI	**0.938**	**0.945**	**0.908**	**0.916**	**0.898**	**0.893**
Scenario 4						
DNN	0.769	0.774	0.743	0.754	0.722	0.732
BiGAN+DNN	0.815	0.828	0.803	0.805	0.785	0.794
EBiGAN+DNN	0.867	0.889	0.846	0.864	0.832	0.836
EBiGAN+DNN-PI	**0.958**	**0.937**	**0.938**	**0.925**	**0.916**	**0.919**
Scenario 5						
DNN	0.734	0.780	0.718	0.758	0.701	0.739
BiGAN+DNN	0.837	0.897	0.821	0.878	0.806	0.845
EBiGAN+DNN	0.876	0.904	0.854	0.887	0.836	0.876
EBiGAN+DNN-PI	**0.956**	**0.964**	**0.938**	**0.947**	**0.915**	**0.927**

VI. CONCLUSION

In the IoT infrastructure, insider threat detection is challenging because authorized employees access sensitive data within institutions. Cybersecurity researchers have developed effective ITD models using neural networks to protect IoT enabled institutions. However, the existing ITD models have complications when working with a benchmark insider dataset, such as data imbalance problems and confrontation, to maintain the model's generalizability and interpretability. The issues discussed above are addressed by the proposed detection model to detect insider threats in the institutional log records of CMU CERT. The proposed detection model comprises three modules: IPCA for extracting the important user functionality-based samples and outlier estimators of k-means clustering for grouping the scenario-based user functionality samples to achieve dimensionality reduction and EBiGAN data augmentation to avoid a skewed class distribution in the dataset and ensure the encoded quality of the generated adversarial samples. The DNN-PI identifies the insiders of the IoT infrastructure and improves the overall performance of the model by achieving a high detection rate and minimal false alarm rate. The proposed model works well for the standard insider threat data of IoT enabled institutions. However, the model's limitations remain: it is difficult to adapt to different environments, it requires infeasible resources, and it lacks interpretability in real-time environment. The model ensures enhanced security of infrastructure although it faces complexity. In the future, the proposed model can be employed to generate samples and detect suspicious activities in other real-time scenarios.

REFERENCES

[1] A. Y. Khan, R. Latif, S. Latif, S. Tahir, G. Batool, and T. Saba, "Malicious insider attack detection in IoTs using data analytics," *IEEE Access*, vol. 8, pp. 11743–11753, 2020, doi: 10.1109/ACCESS.2019.2959047.

[2] A. R. Marbut and P. D. Harms, "Fiends and fools: A narrative review and neo- socio analytic perspective on personality and insider threats," *J. Bus. Psychol.*, vol. 39, pp. 679–696, May 2023, doi: 10.1007/s10869-023-09885-9.

[3] Y. Storchak. (2024). *Insider Threat Statistics for 2024: Facts, Reports & Costs*. Accessed: Nov. 13, 2023. [Online]. Available: https://www.ekransystem.com/en/blog/insider-threat-statistics-facts-andfigures

[4] M. Burhan, R. Rehman, B. Khan, and B.-S. Kim, "IoT elements, layered architectures and security issues: A comprehensive survey," *Sensors*, vol. 18, no. 9, p. 2796, Aug. 2018.

[5] B. Pahlevanzadeh, S. Koleini, and S. I. Fadilah, "Security in IoT: Threats and vulnerabilities, layered architecture, encryption mechanisms, challenges and solutions," in *Communications in Computer and Information Science*. Singapore: Springer, 2021, pp. 267–283.

[6] R. B. Peccatiello, J. J. C. Gondim, and L. P. F. Garcia, "Applying one-class algorithms for data stream-based insider threat detection," *IEEE Access*, vol. 11, pp. 70560–70573, 2023, doi: 10.1109/ACCESS.2023.3293825.

[7] A. Kim, J. Oh, J. Ryu, and K. Lee, "A review of insider threat detection approaches with IoT perspective," *IEEE Access*, vol. 8, pp. 78847–78867, 2020, doi: 10.1109/ACCESS.2020.2990195.

[8] T. Al-Shehari and R. A. Alsowail, "An insider data leakage detection using one-hot encoding, synthetic minority oversampling and machine learning techniques," *Entropy*, vol. 23, no. 10, p. 1258, Sep. 2021.

[9] M. N. Al-Mhiqani, R. Ahmed, Z. Zainal, and S. N. Isnin, "An integrated imbalanced learning and deep neural network model for insider threat detection," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 1, pp. 573–577, 2021, doi: 10.14569/ijacsa.2021.0120166.

[10] F. Yuan, Y. Shang, Y. Liu, Y. Cao, and J. Tan, "Data augmentation for insider threat detection with GAN," in *Proc. IEEE 32nd Int. Conf. Tools With Artif. Intell. (ICTAI)*, Nov. 2020, pp. 632–638.

[11] D. Zhu, X. Huang, N. Li, H. Sun, M. Liu, and J. Liu, "RAP-Net: A resource access pattern network for insider threat detection," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2022, pp. 1–8.

[12] T. K. Rao, N. Darapaneni, A. R. Paduri, A. Kumar, and G. Ps, "Insider threat detection: Using classification models," in *Proc. 15th Int. Conf. Contemp. Comput.*, Aug. 2023, pp. 307–312.

[13] X. Li, X. Li, J. Jia, L. Li, J. Yuan, Y. Gao, and S. Yu, "A high accuracy and adaptive anomaly detection model with dual-domain graph convolutional network for insider threat detection," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 1638–1652, Feb. 2023, doi: 10.1109/TIFS.2023.3245413.

[14] H. Sepehrzadeh, "A method for insider threat assessment by modeling the internal employee interactions," *Int. J. Inf. Secur.*, vol. 22, no. 5, pp. 1385–1393, Oct. 2023.

[15] A. Alshehri, N. Khan, A. Alowayr, and M. Y. Alghamdi, "Cyberattack detection framework using machine learning and user behavior analytics," *Comput. Syst. Sci. Eng.*, vol. 44, no. 2, pp. 1679–1689, 2023, doi: 10.32604/csse.2023.026526.

[16] Y. Li and Y. Su, "The insider threat detection method of university website clusters based on machine learning," in *Proc. 6th Int. Conf. Artif. Intell. Big Data (ICAIBD)*, May 2023, pp. 560–565.

[17] R. Yousef, M. Jazzar, A. Eleyan, and T. Bejaoui, "A machine learning framework & development for insider cyber-crime threats detection," in *Proc. Int. Conf. Smart Appl., Commun. Netw. (SmartNets)*, Jul. 2023, pp. 1–6.

[18] M. Amiri-Zarandi, H. Karimipour, and R. A. Dara, "A federated and explainable approach for insider threat detection in IoT," *Internet Things*, vol. 24, Dec. 2023, Art. no. 100965.

[19] F. Aloraini, A. Javed, O. Rana, and P. Burnap, "Adversarial machine learning in IoT from an insider point of view," *J. Inf. Secur. Appl.*, vol. 70, Nov. 2022, Art. no. 103341.

[20] M. Chowdhury, B. Ray, S. Chowdhury, and S. Rajasegarar, "A novel insider attack and machine learning based detection for the Internet of Things," *ACM Trans. Internet Things*, vol. 2, no. 4, pp. 1–23, Nov. 2021.

[21] Y. M. Tukur, D. Thakker, and I. Awan, "Edge-based blockchain enabled anomaly detection for insider attack prevention in Internet of Things," *Trans. Emerg. Telecommun. Technol.*, vol. 32, no. 6, p. e4158, Jun. 2021.

[22] B. Sowmiya, K. Saminathan, and M. C. Devi, "Classification of paddy leaf diseases with extended Huber loss function using convolutional neural networks," *ICTACT J. Soft Comput.*, vol. 3, no. 3, pp. 1–9, 2023, doi: 10.21917/ijsc.2023.0403.

[23] A. I. Iliev and A. Anand, "Huber loss and neural networks application in property price prediction," in *Proc. Future Inf. Commun. Conf.* Cham, Switzerland: Springer, 2023, pp. 242–256.

[24] M. O. Kaplan and S. E. Alptekin, "An improved BiGAN based approach for anomaly detection," *Proc. Comput. Sci.*, vol. 176, pp. 185–194, Jan. 2020.

[25] H. Sadoune, R. Rihani, and F. S. Marra, "DNN model development of biogas production from an anaerobic wastewater treatment plant using Bayesian hyperparameter optimization," *Chem. Eng. J.*, vol. 471, Sep. 2023, Art. no. 144671.

[26] B. Lindauer. (Sep. 30, 2020). *Insider Threat Test Dataset*. Pittsburgh, PA, USA: Carnegie Mellon Univ. Accessed: Sep. 10, 2023. [Online]. Available: https://kilthub.cmu.edu/articles/dataset/Insider_Threat_Test_Dataset/12841247/1

[27] R. Nasir, M. Afzal, R. Latif, and W. Iqbal, "Behavioral based insider threat detection using deep learning," *IEEE Access*, vol. 9, pp. 143266–143274, 2021, doi: 10.1109/ACCESS.2021.3118297.

[28] I. Ullah, K. Mengersen, R J Hyndman, and J. McGree, "Detection of cybersecurity attacks through analysis of web browsing activities using principal component analysis," 2021, *arXiv:2107.12592*.

[29] T. Chadza, K. G. Kyriakopoulos, and S. Lambotharan, "Analysis of hidden Markov model learning algorithms for the detection and prediction of multi-stage network attacks," *Future Gener. Comput. Syst.*, vol. 108, pp. 636–649, Jul. 2020.

[30] Q. Sun, R. Tao, Y. Shi, and X. Shang, "Add-BiGAN: An add-based bidirectional generative adversarial networks for intrusion detection," in *Proc. Int. Conf. Knowl. Manage. Organizations*. Cham, Switzerland: Springer, 2023, pp. 360–374.

[31] N. Sandholtz, Y. Miyamoto, L. Bornn, and M. A. Smith, "Inverse Bayesian optimization: Learning human acquisition functions in an exploration vs exploitation search task," *Bayesian Anal.*, vol. 18, no. 1, pp. 1–24, Mar. 2023.

[32] Y. Fang, M. Niu, P. Cheung, and L. Lin, "Extrinsic Bayesian optimizations on manifolds," 2022, *arXiv:2212.13886*.

[33] M. Nasser Al-Mhiqani, R. Ahmad, Z. Z. Abidin, W. Yassin, A. Hassan, and A. N. Mohammad, "New insider threat detection method based on recurrent neural networks," *Indonesian J. Electr. Eng. Comput. Sci.*, vol. 17, no. 3, p. 1474, Mar. 2020.

[34] R. G. Gayathri, A. Sajjanhar, and Y. Xiang, "Adversarial training for robust insider threat detection," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2022, pp. 1–8.

[35] R. G. Gayathri, A. Sajjanhar, Y. Xiang, and X. Ma, "Anomaly detection for scenario-based insider activities using CGAN augmented data," in *Proc. IEEE 20th Int. Conf. Trust, Secur. Privacy Comput. Commun. (TrustCom)*, Oct. 2021, pp. 718–725.

[36] M. Singh, B. M. Mehtre, S. Sangeetha, and V. Govindaraju, "User behaviour based insider threat detection using a hybrid learning approach," *J. Ambient Intell. Humanized Comput.*, vol. 14, no. 4, pp. 4573–4593, Apr. 2023.

[37] W. Hong, J. Yin, M. You, H. Wang, J. Cao, J. Li, M. Liu, and C. Man, "A graph empowered insider threat detection framework based on daily activities," *ISA Trans.*, vol. 141, pp. 84–92, Oct. 2023.

[38] Y. Wei, K.-P. Chow, and S.-M. Yiu, "Insider threat prediction based on unsupervised anomaly detection scheme for proactive forensic investigation," *Forensic Sci. Int., Digit. Invest.*, vol. 38, Oct. 2021, Art. no. 301126.

[39] R. G. Gayathri, A. Sajjanhar, and Y. Xiang, "Hybrid deep learning model using SPCAGAN augmentation for insider threat analysis," *Expert Syst. Appl.*, vol. 249, Sep. 2024, Art. no. 123533.

[40] W. Huang, H. Zhu, C. Li, Q. Lv, Y. Wang, and H. Yang, "ITDBERT: Temporal-semantic representation for insider threat detection," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Sep. 2021, pp. 1–7.

P. LAVANYA received the bachelor's degree in information technology and the master's degree in computer science and engineering (specialization in networks) from Anna University, Chennai, India, in 2016 and 2019, respectively. She is currently a Research Scholar with the Centre for Information Super Highway (CISH), School of Computing, SASTRA Deemed University, Thanjavur, Tamil Nadu, India. Her current research interests include cybersecurity, information and network security, machine learning, deep learning, and artificial intelligence.

H. ANILA GLORY received the bachelor's and master's degrees in computer science and engineering from Anna University, Chennai, India, in 2012 and 2014, respectively. She is currently an Assistant Professor with the School of Computing, SASTRA Deemed to be University, Thanjavur, India. She has been in academia for the past eight years. Her current research interests include brain–computing interface, data mining, information and network security, machine learning, deep learning, big data analytics, and artificial intelligence.

V. S. SHANKAR SRIRAM received the Ph.D. degree in information and network security from the Birla Institute of Technology, Mesra, India. He is currently the Dean of Computer Science and Engineering and the TATA Communication Chair Professor of Cyber Security with the School of Computing, SASTRA Deemed University, Thanjavur, Tamil Nadu, India. He has been in academia for the past 20 years and was awarded the IBM Shared University Research Award, in 2017. His research interests include information and network security, cloud computing, bioinformatics, big data analytics, machine learning, deep learning, and graph-based data mining.