INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 19, Issue: 5, Month: October, Year: 2024
Article Number: 6767, https://doi.org/10.15837/ijccc.2024.5.6767

Evaluating Dimensionality Reduction Methods for the Detection of Industrial IoT Attacks in Edge Computing

Minh T. Hoang, Nhan V. Nguyen, Thu A. Pham, Tra T. Nguyen
Tuan M. Dang, Hoang N. Nguyen

Minh T. Hoang
Telecommunication Faculty No1
Posts and Telecommunications Institute of Technology, Hanoi, Vietnam
hoangtrongminh@ptit.edu.vn

Nhan V. Nguyen
Telecommunication Faculty No1
Posts and Telecommunications Institute of Technology, Hanoi, Vietnam
nhannv.b23chte007@stu.ptit.edu.vn

Thu A. Pham
Telecommunication Faculty No1
Posts and Telecommunications Institute of Technology, Hanoi, Vietnam
thupa@ptit.edu.vn

Tra T. Nguyen
Telecommunication Faculty No1
Posts and Telecommunications Institute of Technology, Hanoi, Vietnam
trant@ptit.edu.vn

Tuan M. Dang
Department of Microelectronics and Telecommunications
CMC University, Hanoi, Vietnam
dmtuan@cmc-u.edu.vn

Hoang N. Nguyen*
Faculty of Electronics and Telecommunications Networking
VNU-University of Engineering and Technology, Hanoi, Vietnam
*Corresponding author: hoangnn@vnu.edu.vn

Abstract

Edge computing is essential for 6G mobile networks for improving reliability, reducing data rates and latency, and enhancing mobile connectivity. Edge computing is also meant to meet the increasing demands of the Internet of Things (IoT)/ Internet of Everything (IoE). In these approaches, IIoT systems necessitate precision, reliability, and scalability, while vulnerabilities in IIoT systems may lead to financial losses and safety hazards. To tackle this, Edge AI/ML-based IDSs provide adaptability and robustness for IIoT security challenges. These solutions improve security levels, threat detection rates, and response times. However, main issues such as limited resources and the accuracy of attack detection rate are challenged nowadays. In this paper, we present contributions in proposing an intrusion detection system (IDS) for edge devices deploying a lightweight deep neural network to detect IIoT attacks. We present performance analysis of the typical dimensionality reduction methods and balancing data features of the Edge-IIoT dataset to get higher performance metrics than other state-of-the-art studies.

Keywords: Industrial IoT, attacks, intrusion detection systems, dimensionality reduction, and deep neuron networks.

1 Introduction

1.1 Background

Edge technology is essential for 6G mobile network reliability, data rate reduction, latency reduction, and worldwide connection [9] [14]. In edge computing environment, edge devices are essential to these advances which are projected to be widely deployed as integral nodes in mobile networks to fulfill the rising demands of the IoT becoming the IoE [21] [34]. Edge computing and 6G networks can offload tasks to edge servers for improving resource usage and performance in autonomous vehicles and mobile devices in real-time task offloading. Edge devices and 6G technology must work together to maximize IoT/IoE applications and drive linked technology innovation. Industrial Internet of Things (IIoT) systems require precision, reliability, and scalability beyond traditional IoT systems, requiring increased security [23] [5]. IIoT vulnerabilities can cause economic losses, safety risks, and reputational damage [25]. AI/ML-based Intrusion Detection Systems (IDS) for IIoT can help industrial IoT systems secure sensitive data, comply with regulations, and mitigate infrastructure attacks [6] [7]. Edge AI/ML-based IDSs provide adaptability and robustness for IIoT environments for ensuring the continuity of industrial processes and reducing latency, improving privacy, scalability, and cost-effectiveness [22]. These solutions protect sensitive data, speed up threat detection and response, and offer security management flexibility [1]. However, processing power, system complexity, data management concerns, and security vulnerabilities can hinder edge AI/ML-based IDS adoption [20, 31]. These barriers must be overcome to exploit these applications and ensure efficient and appropriate security when navigating edge computing environments. Dimensionality reduction techniques are essential for reducing high-dimensional data, which improves data preparation. These methods reduce computational complexity during training and prediction by decreasing characteristics [10, 15]. Dimensionality decrease reduces overfitting and improves machine learning models [26]. These techniques also help visualize data and understand model behavior, which can be helpful in many applications [32]. Hence, dimensionality reduction simplifies computation and improves model performance and interpretability. Machine learning deployed on network edge devices requires data dimension reduction due to resource constraints [30]. Edge data reduction improves quality of applications, IoT system bottlenecks, data management, storage, and computation time. Deep Neural Network (DNN) models for medical picture recognition use EMD, Principal Component Analysis (PCA), and correlation [4]. These methods minimize dimensionality and select key characteristics to improve performance of medical picture detection machine learning. However, machine learning for IIoT's IDS does not have comprehensive dimensionality reduction studies. In this paper, we will deploy a DNN architecture with actual EdgeIoT data for an IDS system and study the performance of dimensionality reduction methods. Simulation results show that the IDS system identifies assaults even after decreasing dimension parameters of the edge device.

1.2 Related Works

Preprocessing data is vital before model training. Dimensionality reduction is one of major preprocessing techniques, which reduces the number of features in a dataset while retaining the most essential information [8]. This leads to more accurate and efficient models by eliminating redundant and noisy features and simplifying methods that struggle with high dimensions [28]. Dimension reduction can also aid noise reduction, data visualization, cluster analysis, and other investigations. Many dimensionality reduction approaches exist, including feature selection and extraction. PCA, Factor Analysis (FA), Linear Discriminant Analysis (LDA), SVD, Kernel PCA, t-SNE, MDS, and Isomap are commonly used data analysis methods[18, 24, 27]. Dimensionality reduction techniques like PCA have been shown to improve the performance of machine learning models in various applications. A study in [16] looks at how well the PCA dimensionality reduction algorithm works with autoencode, neural networks (NN), and support vector machines (SVM) to predict cancer. The results showed that the proposal reduced the complexity and increased the performance of ML models. The authors in [17] evaluated PCA algorithms with different ML techniques to enhance intrusion detection models' performance. Evaluation measures, including accuracy, precision, and recall on the NSL-KDD dataset, demonstrated clear performance benefits. In [2], AE and PCA techniques were used to reduce the dimensionality of features in the CICIDS2017 dataset. The results show that the features of the CICIDS2017 dataset have been reduced from 81 to 10 while maintaining a high accuracy of 99.6% in multiclass and binary classification. When testing dimensionality reduction of up to 10 features with four IoT attack datasets (NSL, NB15, BoTIoT BoTNeTIoT), the author in [3] proposed data dimensionality reduction methods including PCA, Random-Forest and Pearson Correlation.AE and PCA techniques were also used to reduce the dimensionality of features in the CICIDS2017 dataset. Experimental results show that no solution stands out when it comes to meeting both performance and accuracy goals. The authors in [12] used the Local Intrinsic Dimensionality (LID) method to test the algorithm on the ToN IoT, NF Bot-IoT, and IoT-23 datasets. This method was better than traditional ones at finding attacks on IoT. Our previous work on the IoT-23 dataset using the PCA algorithm combined with a DNN achieved an attack detection rate of 99% [13]. In [11], the Edge-IIoTset dataset was explicitly designed for evaluating intrusion detection systems in the context of edge computing in Industrial IoT (IIoT) environments. It details the dataset creation process, including the types of attacks and network traffic scenarios. For IoT and IIoT applications, [11] has proposed the Edge-IIoTset dataset as a comprehensive, useful cybersecurity dataset. The Edge-IIoTset dataset has been recognized as a "Document in the top 1% of the Web of Science." The authors in [19, 29] made tests that showed the Kmeans-SMOTE method, which combines the K-means clustering algorithm and the SMOTE oversampling algorithm, works better than random oversampling, SMOTE and borderline-SMOTE when working with uneven data during dimensionality reduction processes. The detailed survey indicates that the Kmeans-SMOTE method efficiently addresses imbalances within and between layers while preventing the creation of unnecessary noise. As a result, we employ this method in our proposed IDS model evaluations to enhance the attack detection rate.

1.3 Contributions

The main contributions of this paper are as follows:

- This study proposes an IDS system for edge devices, including preprocessing, data transformation, and a lightweight deep neural network, to detect security attacks in industrial IoT environment.
- The study evaluates dimensionality reduction methods and balancing data features of the Edge-IIoT dataset to make recommendations for the selection of dimensionality reduction methods that are able to improve performance for detecting industrial IoT attacks.

1.4 Organization

The organization of this paper is as follows: The following section briefly introduces the proposed system model, which contains the main processing phases. Session 3 will describe the Edge IoT dataset used in the IDS model with essential dimensionality reduction methods and unbalanced data processing. Session 4 will show simulation and evaluation findings. Our conclusions and future research are presented in the last section.

2 System model

2.1 The outline of the system model

The proposed system model for detecting IIoT attacks on edge devices incorporates a lightweight Deep Neural Network (DNN) to optimize performance and resource utilization. This approach is critical for edge computing environments, where computational and memory resources are limited.

The system model's workflow is shown in Figure 1 and comprises three stages: preprocessing, feature selection, and evaluation. The evaluation stage focuses on reducing the number of dimensions and dealing with class imbalance.

The initial step involves acquiring data from the Industrial Internet of Things (IIoT) ecosystem to construct a comprehensive dataset. The dataset is then subjected to a data preprocessing procedure to transform it into an appropriate format for machine learning and deep learning methodologies. This preprocessing stage includes the removal of duplicate rows, handling missing or infinite values, and encoding categorical features using a binary encoder.

Next, a dimensionality reduction methods is used to eliminate some features while keeping the important data. The method can be one of typical principal component analysis (PCA), linear discriminant analysis (LDA), independent component analysis (ICA), factor analysis (FA), and uniform manifold approximation and projection (UMAP) methods. The dataset is partitioned into training, validation, and test sets to facilitate model development and evaluation. In the first phase, each dimensionality reduction method is evaluated using a lightweight deep neural network (DNN) to determine the most effective technique.

The lightweight DNN in this paper is designed for edge computing environments with limited computational resources and power consumption [33]. It maintains high accuracy in detecting attacks while minimizing

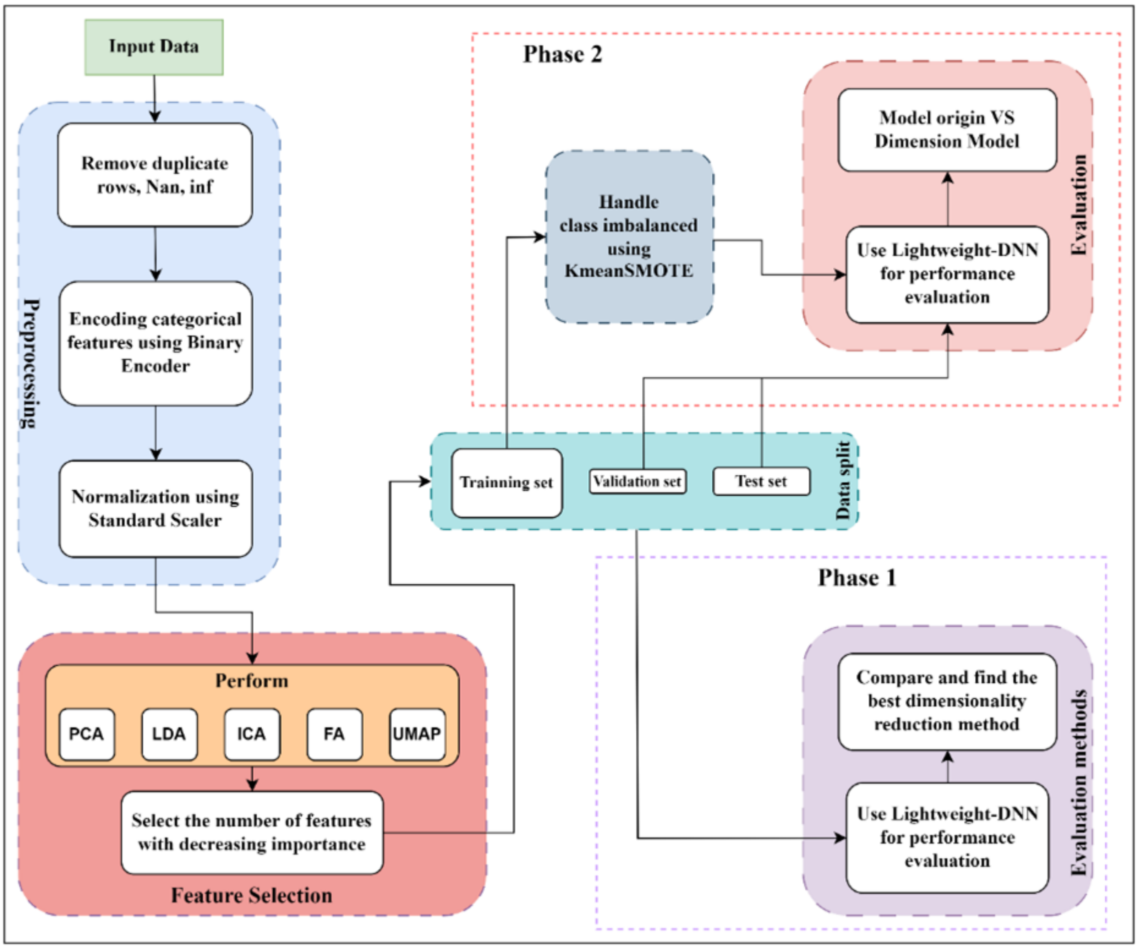

Input Data
Preprocessing
Remove duplicate rows, Nan, inf
Encoding categorical features using Binary Encoder
Normalization using Standard Scaler
Perform
PCA
LDA
ICA
FA
UMAP
Select the number of features with decreasing importance
Feature Selection
Phase 2
Handle class imbalanced using KmeanSMOTE
Model origin VS Dimension Model
Use Lightweight-DNN for performance evaluation
Evaluation
Trainning set
Validation set
Test set
Data split
Phase 1
Compare and find the best dimensionality reduction method
Use Lightweight-DNN for performance evaluation
Evaluation methods

Figure 1: System model.

parameters, making it ideal for edge devices. By integrating dimensionality reduction, the DNN processes fewer features, reducing computational load and memory usage, resulting in faster inference times and lower power consumption. This is essential for real-time IIoT applications like automated industrial systems and real-time security monitoring. The DNN model is a fully connected neural network starting with 64 nodes and compressed to 32 nodes, using the 'gelu' activation function and a five percent dropout layer. For the multi-class classification problem, the model's output is a softmax function with 15 nodes for 15 attack types. We use the 'Adam optimizer' and cross-entropy function to adjust optimization weights, identifying a learning rate of 0.0001 as optimal for achieving the highest detection rate. To minimize overfitting, we implement early stopping during training, halting when the validation error does not decrease after a set number of iterations, allowing us to adjust the number of epochs to achieve the best accuracy during testing.

The second phase involves applying the optimal dimensionality reduction method and addressing class imbalance using KmeansSMOTE, focusing on the training set. This balanced dataset is then used to train the DNN for attack classification, with the outcomes detailed in Section 3.2. This comprehensive workflow ensures the data is effectively processed and analyzed, leading to robust and accurate machine-learning models for IIoT applications.

2.2 Edge-IIoT Dataset and Preprocessing

In 2022, the Edge-IIoTset dataset was introduced to provide a realistic and comprehensive data source for evaluating AI-based intrusion detection systems focusing on realtime challenges [11]. This dataset includes a wide range of devices, sensors, protocols, and cloud/edge computing configurations, making it ideal for testing the intrusion detection capabilities of machine learning models. The initial raw dataset comprised up to 1,176 features. After thorough analysis, the research team filtered 61 relevant features, resulting in two CSV files. In this paper, we use the "DNN-EdgeIIoT-dataset.csv" to evaluate various dimensionality reduction techniques on a deep neural network (DNN) for anomaly detection. The dataset classifies attacks into 15 distinct categories, encompassing 14 specific attack types and one benign label, offering valuable insights into the security landscape of the IIoT ecosystem.

The data from the original Industrial Internet of Things (IIoT) dataset undergoes a series of preprocessing steps to ensure data integrity and suitability for analysis.

- The "Attack_type" label and data from the *DNN-EdgeIIoT-dataset.csv* are fed into a multi-layer classification model.
- Entries with errors labeled as "NAN" (Not a Number) or "INF" (Infinity) are removed to maintain data accuracy and prevent duplication.
- Duplicate rows and features with identical unique values are eliminated to streamline the dataset.
- Irregularities in the dataset are addressed using binary encoding techniques to classify behaviors.
- The data are standardized using a Standard scale to normalize the feature values.
- Dimensionality reduction techniques are applied to decrease the number of model parameters and computational requirements.
- The dataset is split randomly into training, validation, and test sets using the `train_test_split` function, with an 80:10:10 ratio.

The specific distribution of attack types is detailed in Table 1.

2.3 Dimensionality reduction techniques

PCA involves identifying principal components and linear combinations of the original features to maximize the dataset's variance. The fundamental concept of PCA revolves around transforming data into a new coordinate system defined by these principal components, with the primary objective being the preservation of global variance rather than intricate local correlations. The critical steps outlining the PCA process include calculating the covariance matrix, eigenvalue decomposition, selecting principal components, and transforming data into a lower-dimensional space.

LDA is another significant dimension reduction technique that is particularly advantageous in supervised learning scenarios. LDA emphasizes enhancing separability among distinct classes by transforming data into a space that optimally discriminates between predefined classes. The methodology encompasses calculating mean vectors for each class, computing scatter matrices to comprehend class variance, determining eigenvectors and eigenvalues through eigenvalue decomposition, forming a transformation matrix using selected eigenvectors, and ultimately transforming data to a new lower-dimensional space to improve class separability.

ICA is a computational method that separates a multivariate signal into additive, independent non-Gaussian components. In contrast to the PCA, which focuses on maximizing variance, the ICA aims to identify components that exhibit statistical independence. The essential steps involved in ICA include (1) subtracting the mean to achieve data centering, (2) applying whitening to convert the data into uncorrelated components with unit variance, and (3) utilizing algorithms such as FastICA to estimate the independent components through the maximization of non-Gaussianity.

FA is a statistical technique that improves principal component analysis, also known as multivariate statistical analysis. It groups variables based on correlations to reveal intercorrelations among attributes by reducing them to fewer factors. The method reduces dataset dimensionality by approximating the covariance matrix. The FA produces a table with original variables as rows and factors as columns, displaying factor loadings. These loadings show variable-factor correlations and identify strong associations. The steps include calculating the initial factor loading matrix using the principal component or principal axis factoring methods. Factor rotation enhances interpretability by ensuring high loadings on one factor per variable through orthogonal or oblique rotation. Factor scores are calculated for each factor, with the total variance explained by the original variables exceeding 75%. The number of factors chosen typically corresponds to eigenvalues greater than 1.

UMAP is a technique for dimensionality reduction and visualization. UMAP maintains local and global structures in high-dimensional datasets better than other techniques. UMAP creates a topological representation of data and projects it onto a lower-dimensional space. Unlike PCA, UMAP focuses on preserving neighborhood relationships to capture non-linear patterns effectively. UMAP involves constructing a topological graph, optimizing for low-dimensional representation, and embedding data into a lower-dimensional space for visualization.

3 Numerical results

This section involves a comparative analysis of different dimensionality reduction methods applied to the Edge-IIoT dataset mentioned before. The efficacy of these methods is evaluated based on classification accuracy, the count of model parameters, and after dimensionality reduction. The experiments used Python 3.7, the TensorFlow framework, and the Keras module for deep learning tasks. The experimental configuration included a PC with an Intel® Core i9 10900K processor, 48GB of RAM, and an Nvidia® RTX A4000 GPU. The training process consisted of 200 epochs in Phase 1, then extended to 1000 epochs in Phase 2. Before evaluating the effectiveness of dimensionality reduction methods, we present the data imbalance handling method as follows: The KmeanSMOTE method is utilized to address class imbalances in the dataset. This technique ensures that the classification model does not exhibit bias towards the majority class by accounting for the unequal representation of different classes in the dataset, which is critical during Phase 2 of the workflow. KmeanSMOTE uses both K-means clustering and the Synthetic Minority Over-sampling Technique (SMOTE) to make fake samples for the minority groups, which evens out the dataset. Table 1 below provides a detailed breakdown of the number of instances for each attack type after the dataset has been divided into training, validation, and test sets. In the dataset, some attack types have significantly more training data than others. By addressing class imbalances, the KmeanSMOTE method plays a crucial role in enhancing the robustness and accuracy of the machine learning model, leading to better performance in detecting various types of attacks within the industrial IoT security landscape.

Table 1: Distribution of attack types in each data split

Attack_types	**Train**	**Val**	**Test**
Backdoor	1,148,439	143,555	143,555
DDoS_HTTP	97,253	12,157	12,157
DDoS_ICMP	93,147	11,644	11,643
DDoS_TCP	40,783	5,098	5,098
DDoS_UDP	40,066	5,008	5,008
Fingerprinting	40,050	5,006	5,006
MITM	40,025	5,003	5,003
Normal	39,870	4,983	4,984
Password	29,791	3,724	3,724
Port_Scanning	19,230	2,404	2,404
Ransomware	15,992	1,999	1,999
SQL_injection	12,070	1,509	1,509
Uploading	7,760	970	970
Vulnerability_scanner	688	86	86
XSS	320	40	40
SUM	**1,625,484**	**203,186**	**203,186**

To compare dimensionality reduction methods, we looked at how many features were kept after reduction and how they affected a deep neural network (DNN) model that had already been built [13]. We also experimented with dimensions starting from two, incrementally increasing until the model achieved an accuracy of approximately 99%. This would help to ensure minimal loss of accuracy relative to the number of model parameters. By using the same model, reduced dimensionality also implied reduced model complexity. The primary goal of employing dimensionality reduction was to decrease the model size while maintaining robust classification accuracy.

3.1 Results of dimensionality reduction methods

The results of dimensionality reduction using PCA are presented in Table 2.

Table 2: Comparison Accuracy for PCA, LDA, ICA, FA, UMAP for different numbers of features

Features	**2F**	**5F**	**7F**	**8F**	**10F**	**14F**	**15F**
PCA	0.9306	0.9808	0.9881	0.9931	0.9981	-	0.9992
LDA	0.8166	0.7990	0.9757	0.7311	0.9949	0.9952	-
ICA	0.9170	0.9362	0.9341	0.9414	0.9627	-	0.9930
FA	0.8952	0.9587	0.9650	0.9810	0.9963	-	0.9990
UMAP	0.8662	0.9813	0.9968	0.9979	0.9984	-	0.9993

The combined table provides a comprehensive comparison of PCA, LDA, ICA, FA, and UMAP across various feature counts (2F, 5F, 7F, 8F, 10F, 14F, and 15F). For LDA, the maximum number of dimensions after using reduction is the number of classes minus one, resulting in a maximum of 14 features. The results showed in Table 2 indicate that model accuracy was generally improved as the number of features increases, highlighting the importance of feature richness in enhancing model performance. Specifically, for PCA, the model's classification accuracy increases gradually as the number of features increases, achieving 99% accuracy with 8 features, which balances the number of model parameters and accuracy. LDA shows varied results, peaking at 10F and 14F with accuracies of 0.9949 and 0.9952, respectively, but performs poorly at 8F (0.7311). The ICA method requires more features than PCA and LDA, needing 15 features to achieve 99% accuracy. FA shows a proportional relationship between the number of features and model accuracy, starting with approximately 90% accuracy at 2 features and reaching 99% accuracy at 10 features. Lastly, UMAP exhibits strong performance, with accuracy rapidly increasing from 0.8662 at 2 features to 0.9984 at 10F.

These findings underscore the diverse requirements and effectiveness of different dimensionality reduction techniques in optimizing model performance. They emphasize the importance of selecting the appropriate method and feature count for specific applications. Based on this comparison, we can determine the optimal feature count for each method to achieve both high performance and lower complexity. The most efficient feature counts identified are PCA_8F, LDA_10F, ICA_15F, FA_10F, and UMAP_7F for their respective techniques.

3.2 Confusion Matrices

The specific results for the classification of each attack and the confusion matrix for each technique are shown in Figures 2, 3,4,5, and 6.

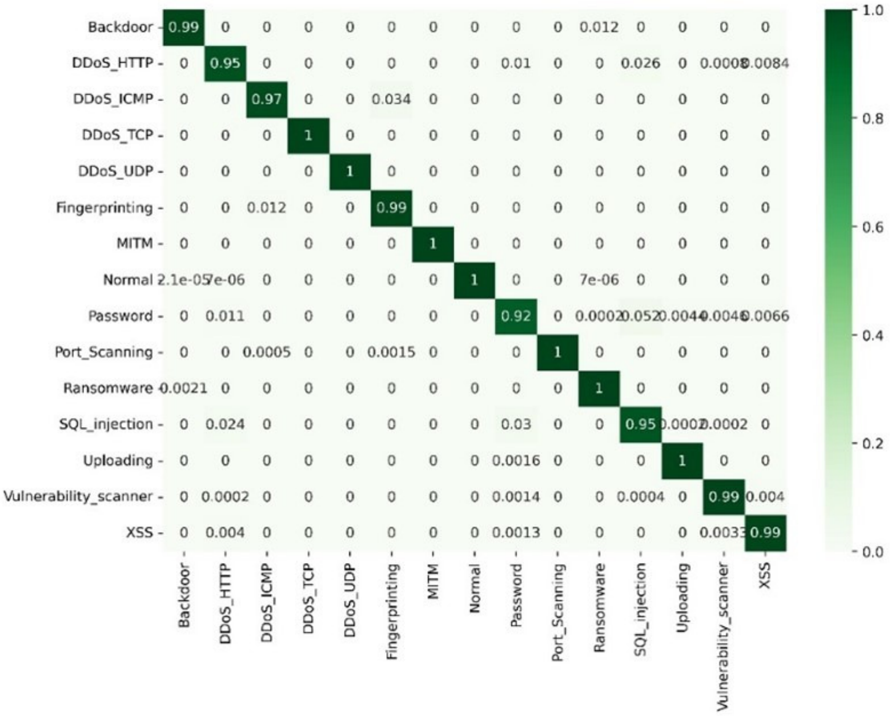


Figure 2: Confusion matrix of PCA_8F

From Figure 2, the PCA_8F model demonstrates consistent performance across most attack types, with the exception of 'Password', which has the lowest detection rate at only 90%. The LDA_10F model, shown in Figure 3, also performs consistently except for 'Fingerprinting,' where the LDA model has 34% of confusing 'Fingerprinting' with DDoS_ICCMP, despite increasing the number of features to 10F. This indicates LDA has more false positives and only captures two-thirds of actual instances, resulting in a low F1 score of 0.10.

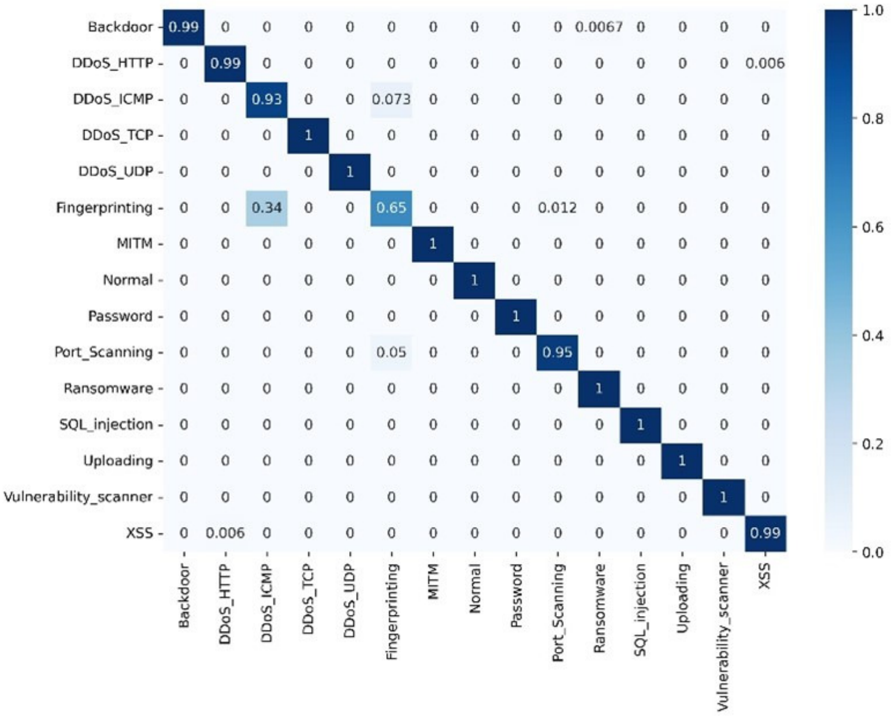


Figure 3: Confusion matrix of LDA__10F

The ICA model, based on Figure 4, shows overall a slightly lower performance compared to PCA__8F and LDA__10F models while needing more features. The FA__10F model, according to 5, performs significantly better compared to the LDA__10F model, considering they use the same number of features. Lastly and most importantly, UMAP__7F, as seen in Figure 6, outperforms other methods with fewer features.

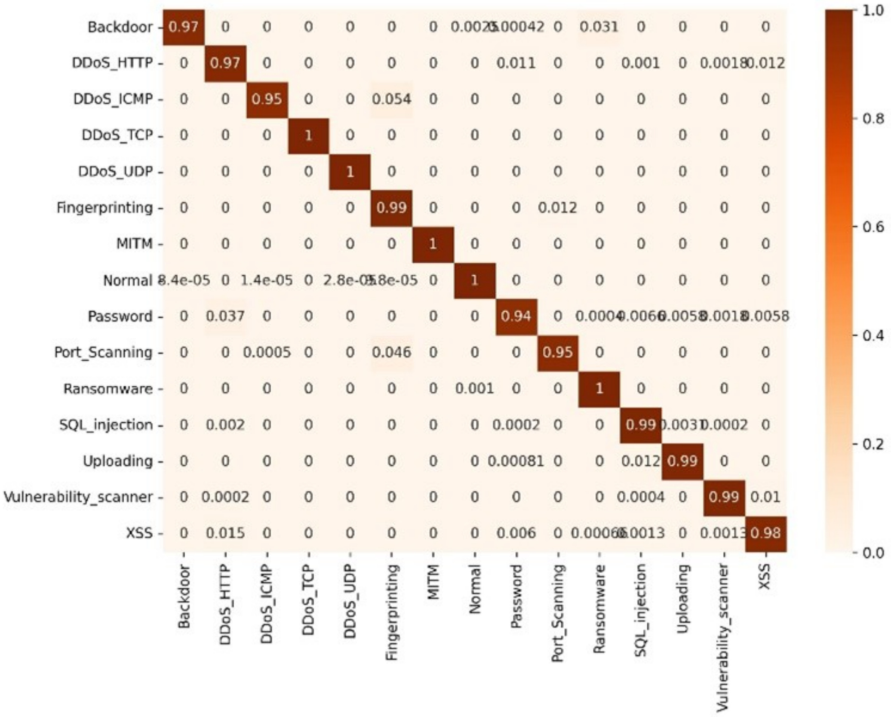


Figure 4: Confusion Matrix of ICA__15F

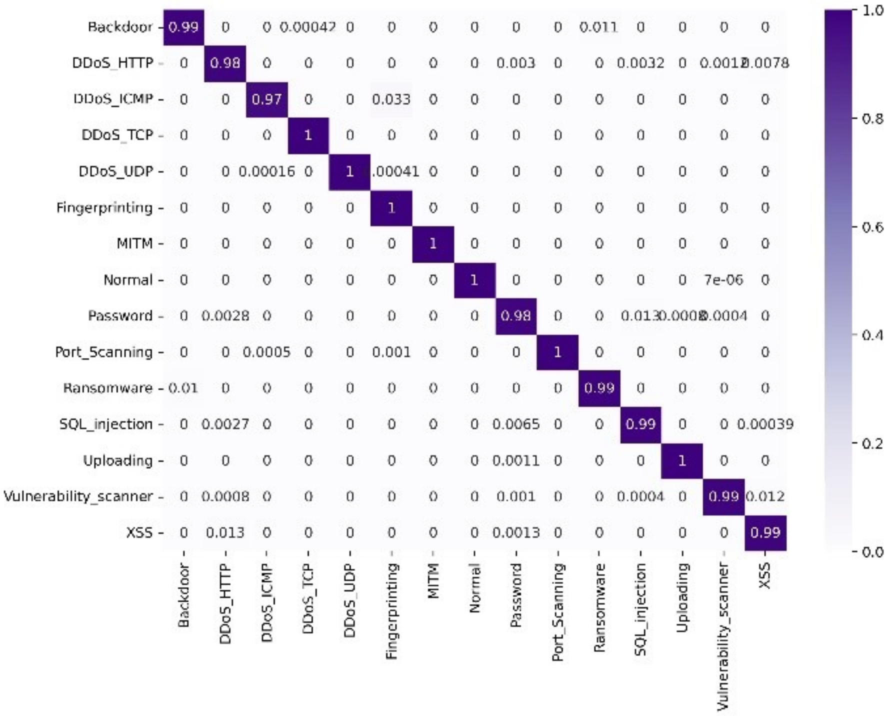


Figure 5: Confusion Matrix of FA_10F

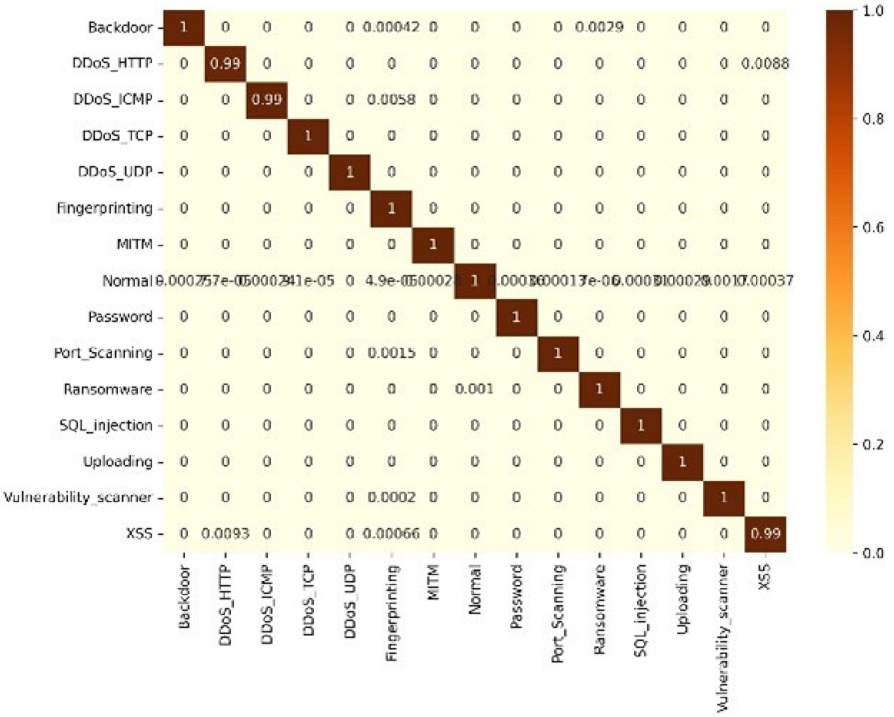


Figure 6: Confusion Matrix of UMAP_7F

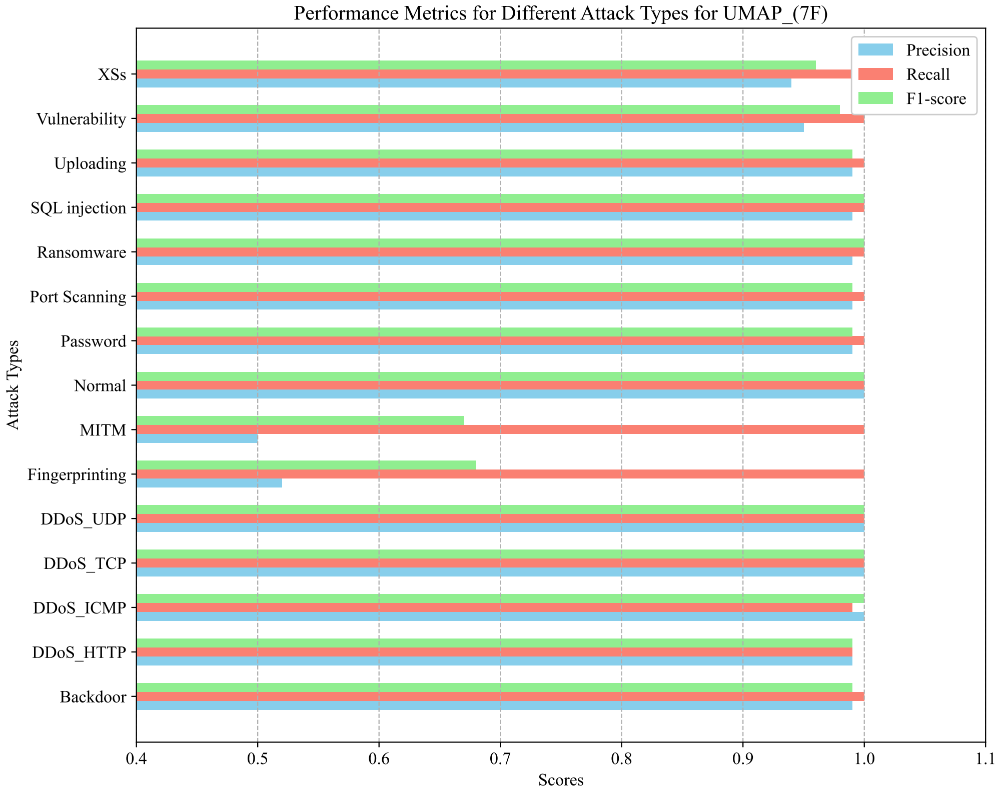


Figure 7: Preformance of UMAP_(7F)

Overall, UMAP_7F delivers superior performance compared to four other methods as it achieved a 'Macro avg (F1-Score)' of 0.95 and a 'Weighted avg (F1-Score)' of 1.00 with fewer features. Therefore, Figure 7 was created to show the detailed performance metrics for UMAP_(7F), demonstrating its effectiveness. For the Recall metric, the lowest score is 0.99 (for detecting DDoS_HTTP, DDoS_ICMP, and XSs), while the rest achieves perfect scores, indicating that the model accurately recognizes and classifies most attack types. Additionally, UMAP_(7F) improves the Precision score for the 'Fingerprinting' class to 0.52, although MITM's Precision is slightly lower at 0.50 compared to other methods.

3.3 Overall performance evaluation

From the evaluation results of the above dimensionality reduction methods, we continue to evaluate the overall performance of the DNN-based IDS system with the non-dimensionality-reduced Edge-IIoT data set (DNN-Origin) with the system using our best dimensionality reduction results and supports data balancing.

The confusion matrices of the two models, DNN and DNN_UMAP_KmeanSMOTE, are shown in Figure 8. Figure 8 shows that the model after dimensionality reduction performs very well compared to the model before reduction. On the main diagonal, only 2 class scores are 0.99, while the rest achieves 1.00 compared to the overall 1.00 of the original model. The number of misclassifications for the original model is 7, whereas for the reduced model, it is 11. This is acceptable as the number of incorrect predictions remains very small while the number of features learned has reduced significantly. Another critical parameter for quickly deploying the model on edge devices is the model parameter count. The number of parameters that are reduced after reducing dimension is shown in Table 3.

Table 3: Parameter Count of DNN_origin and DNN _UMAP(7F)_KmeanSMOTE Models

Model	**DNN_origin**	**DNN-UMAP(7F)-KmeanSMOTE**
Parameter	15,439	3,087
Accuracy	99.98%	99.96%

After dimensionality reduction, the model has demonstrated excellent classification performance (accuracy) at the specific class level. The average F1-Score (Macro avg) is 0.99, and the weighted F1-Score (Weighted avg) is 1.00 in the classification report, showing that the model maintains precise and consistent classification capability across all classes. More importantly, the model has significantly reduced the number of parameters

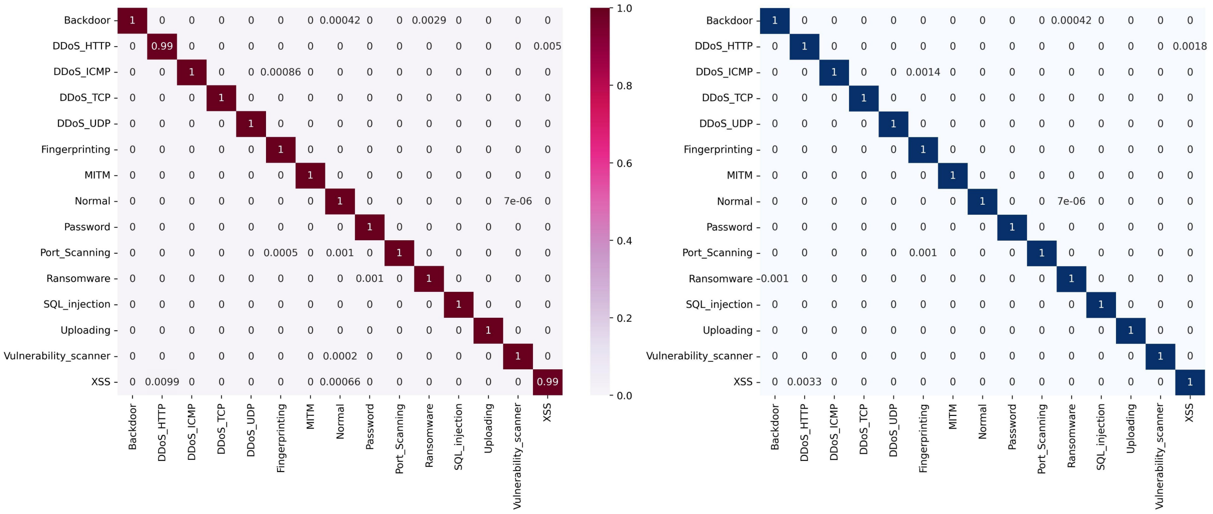
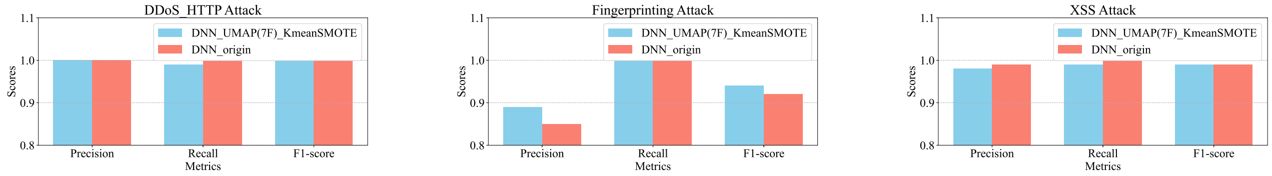


Figure 8: Confusion matrix of DNN_UMAP(7F)_KmSMOTE and DNN_origin

to approximately one-fifth of the original model. It shows that UMAP not only optimizes resources but also enhances the deployability and performance of the model in practical applications.



(a) DDoS_HTTP Attack

(b) Fingerprinting Attack

(c) XSS Attack

Figure 9: Notable difference in the performance of DNN_UMAP(7F)_KmeanSMOTE and DNN_origin

Figure 9 highlights some notable differences between the proposed model and the original model. For all attack types except one, both models achieve scores close to or exactly 1.0 across the three performance metrics. The proposed model not only increases the depth of the model but also maintains or enhances classification performance. The most significant improvement is observed in the "Fingerprinting" class, as shown in Figure 9b, where precision is typically low in most models. This enhancement demonstrates that dimensionality reduction can be an effective strategy for optimizing models, especially when handling large datasets with numerous parameters.

4 Conclusion

The paper has presented and illustrated the effectiveness of dimensionality reduction techniques in improving the detection of attacks in industrial Internet of Things (IIoT) systems. The investigation emphasizes the capacity of these approaches not only to enhance model accuracy but also to notably decrease the number of model parameters, thus optimizing resource utilization for deployment on edge devices. For instance, the DNN-UMAP (7F)-KmeanSMOTE model has a significantly reduced number of parameters while achieving nearly the same level of accuracy as the original model. The model uses fewer resources than the other DNN-based IDS using the original edge IIoT dataset that was applied to IDS on edge devices. This underscores the practical applicability of these methods, which opens a chance to act in real-world IIoT scenarios where computational efficiency and model performance are crucial. Future research will focus on the development of lightweight learning mechanisms for continuous model updates without impacting edge device performance and the integration of explainable AI techniques for a better understanding of attack detection decisions in IIoT systems.

References

[1] Abdelmoniem, A.M.: Leveraging the edge-to-cloud continuum for scalable machine learning on decentralized data. arXiv preprint arXiv:2306.10848 (2023)

[2] Abdulhammed, R., Musafer, H., Alessa, A., Faezipour, M., Abuzneid, A.: Features dimensionality reduction approaches for machine learning based network intrusion detection. Electronics 8(3), 322 (2019)

[3] Alhowaide, A., Alsmadi, I., Tang, J.: Pca, random-forest and pearson correlation for dimensionality reduction in iot ids. In: 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS). pp. 1–6. IEEE (2020)

[4] Ali, I., Wassif, K., Bayomi, H.: Dimensionality reduction for images of iot using machine learning. Scientific Reports 14(1), 7205 (2024)

[5] Alshahrani, H., Khan, A., Rizwan, M., Reshan, M.S.A., Sulaiman, A., Shaikh, A.: Intrusion detection framework for industrial internet of things using software defined network. Sustainability 15(11), 9001 (2023)

[6] Alzahrani, A., Aldhyani, T.H.: Design of efficient based artificial intelligence approaches for sustainable of cyber security in smart industrial control system. Sustainability 15(10), 8076 (2023)

[7] Bibi, I., Akhunzada, A., Kumar, N.: Deep ai-powered cyber threat analysis in iiot. IEEE Internet of Things Journal (2022)

[8] Chen, M.F., et al.: Train and you'll miss it: Interactive model iteration with weak supervision and pre-trained embeddings. ArXiv abs/2006.15168 (2020)

[9] Dai, M., Xu, S., Wang, Z., Ma, H., Qiu, X.: Edge trusted sharing: task-driven decentralized resources collaborate in iot. IEEE Internet of Things Journal 10(14), 12077–12089 (2021)

[10] Dakshinamurthy, S., Gera, B.M., Kayarvizhy, N.: Analytical performance of traditional feature selection methods on high dimensionality data. In: 2023 IEEE 8th International Conference for Convergence in Technology (I2CT). pp. 1–8. IEEE (2023)

[11] Ferrag, M.A., Friha, O., Hamouda, D., Maglaras, L., Janicke, H.: Edge-iiotset: A new comprehensive realistic cyber security dataset of iot and iiot applications for centralized and federated learning. IEEE Access 10, 40281–40306 (2022)

[12] Gorbett, M., Shirazi, H., Ray, I.: Local intrinsic dimensionality of iot networks for unsupervised intrusion detection. In: IFIP Annual Conference on Data and Applications Security and Privacy. pp. 143–161. Springer International Publishing (2022)

[13] Hoang, T.M., Pham, T.A., Do, V.V., Nguyen, V.N., Nguyen, M.H.: A lightweight dnn-based ids for detecting iot cyberattacks in edge computing. In: 2022 International Conference on Advanced Technologies for Communications (ATC). pp. 136–140. IEEE (2022)

[14] Huang, H., Ye, Q., Zhou, Y.: 6g-empowered offloading for realtime applications in multi-access edge computing. IEEE Transactions on Network Science and Engineering 10(3), 1311–1325 (2022)

[15] Jia, W., Sun, M., Lian, J., Hou, S.: Feature dimensionality reduction: a review. Complex & Intelligent Systems 8(3), 2663–2693 (2022)

[16] Kabir, M.F., Chen, T., Ludwig, S.A.: A performance analysis of dimensionality reduction algorithms in machine learning models for cancer prediction. Healthcare Analytics 3, 100125 (2023)

[17] Kayode-Ajala, O.: Anomaly detection in network intrusion detection systems using machine learning and dimensionality reduction. Sage Science Review of Applied Machine Learning 4(1), 12–26 (2021)

[18] Khodr, J., et al.: Dimensionality reduction on hyperspectral images: A comparative review based on artificial datas. In: 2011 4th International Congress on Image and Signal Processing. vol. 4, pp. 1875–1883 (2011)

[19] Last, F., Douzas, G., Bacao, F.: Oversampling for imbalanced learning based on k-means and smote. arXiv preprint arXiv:1711.00837 2 (2017)

[20] Loseto, G., Scioscia, F., Ruta, M., Gramegna, F., Ieva, S., Fasciano, C., Bilenchi, I., Loconte, D., Di Sciascio, E.: A cloud-edge artificial intelligence framework for sensor networks. In: 2023 9th International Workshop on Advances in Sensors and Interfaces (IWASI). pp. 149–154. IEEE (2023)

[21] Lv, P., Xu, W., Nie, J., Yuan, Y., Cai, C., Chen, Z., Xu, J.: Edge computing task offloading for environmental perception of autonomous vehicles in 6g networks. IEEE Transactions on Network Science and Engineering 10(3), 1228–1245 (2022)

[22] MENDİ, A.F.: Edge ai technology in the defense industry via reinforcement learning in simulation environments. Gümüşhane Üniversitesi Fen Bilimleri Dergisi 13(3), 718–732 (2023)

[23] Mohy-Eddine, M., Guezzaz, A., Benkirane, S., Azrour, M., Farhaoui, Y.: An ensemble learning based intrusion detection model for industrial iot security. Big Data Mining and Analytics 6(3), 273–287 (2023)

[24] Nanga, S., et al.: Review of dimension reduction methods. Journal of Data Analysis and Information Processing (2021)

[25] Rullo, A., Bertino, E., Ren, K.: Guest editorial special issue on intrusion detection for the internet of things. IEEE Internet of Things Journal 10(10), 8327–8330 (2023)

[26] Salam, M.A., Azar, A.T., Elgendy, M.S., Fouad, K.M.: The effect of different dimensionality reduction techniques on machine learning overfitting problem. Int. J. Adv. Comput. Sci. Appl 12(4), 641–655 (2021)

[27] Uriot, T., et al.: On genetic programming representations and fitness functions for interpretable dimensionality reduction. In: Proceedings of the Genetic and Evolutionary Computation Conference (2022)

[28] Velliangiri, S., et al.: A review of dimensionality reduction techniques for efficient computation. Procedia Computer Science (2019)

[29] Wang, Q., Li, L., Jiang, B., Lu, Z., Liu, J., Jian, S.: Malicious domain detection based on k-means and smote. In: Computational Science–ICCS 2020: 20th International Conference, Amsterdam, The Netherlands, June 3–5, 2020, Proceedings, Part II. vol. 20, pp. 468–481. Springer International Publishing (2020)

[30] Wu, Q., Zhu, Y., Shi, W., Wang, T., Huang, Y., Jiang, D., Liu, X.: A new data dimension reduction method based on convolution in the application of authenticity identification of traditional chinese medicine longgu. In: Journal of Physics: Conference Series. vol. 2504,1, p. 012035. IOP Publishing (2023)

[31] Xu, D., Li, T., Li, Y., Su, X., Tarkoma, S., Jiang, T., Crowcroft, J., Hui, P.: Edge intelligence: Architectures, challenges, and applications. arXiv preprint arXiv:2003.12172 (2020)

[32] Yaicharoen, A., Hashikura, K., Kamal, M.A.S., Murakami, I., Yamada, K.: Effects of dimensionality reduction on classifier training time and quality. In: 2023 Third International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP). pp. 53–56. IEEE (2023)

[33] Zhang, Q., Han, R., Xin, G., Liu, C.H., Wang, G., Chen, L.Y.: Lightweight and accurate dnn-based anomaly detection at edge. IEEE Transactions on Parallel and Distributed Systems 33(11), 2927–2942 (2021)

[34] Zhang, Z., Xiao, Y., Ma, Z., Xiao, M., Ding, Z., Lei, X., Karagiannidis, G.K., Fan, P.: 6g wireless networks: Vision, requirements, architecture, and key technologies. IEEE vehicular technology magazine 14(3), 28–41 (2019)

Cite this paper as:

Hoang, M.T.; Nguyen, N.V.; Pham, T.A.; Nguyen, T.T.; Dang, T.M.; Nguyen, H.N. (2024). Evaluating Dimensionality Reduction Methods for the Detection of Industrial IoT Attacks in Edge Computing, *International Journal of Computers Communications & Control*, 19(5), 6767, 2024.

https://doi.org/10.15837/ijccc.2024.5.6767