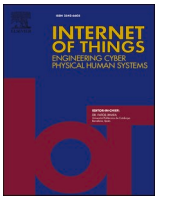
Internet of Things 30 (2025) 101492

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.sciencedirect.com/journal/internet-of-things



HYRIDE: HYbrid and Robust Intrusion DEtection approach for enhancing cybersecurity in Industry 4.0

Shubham Srivastav [a], Amit K. Shukla [b,*], Sandeep Kumar [a], Pranab K. Muhuri [a]

[a] Department of Computer Science, South Asian University, Rajpur Road, Maidan Garhi New Delhi, 110068, India
[b] School of Technology and Innovations, University of Vaasa, Wolffintie 34, FI-65200 Vaasa, Finland

ARTICLE INFO

Keywords:
Feature selection
Local outlier factor
Elliptic envelope
Histogram based outlier score
Unsupervised intrusion detection
Industry 4.0

ABSTRACT

The interconnectedness and smartness aspect between several components of Industry 4.0 has caused sudden increase in data and its exchange, which has resulted in significant cybersecurity challenges. Thus, a better threat intelligence technique is required for monitoring and identifying malicious cyberattacks. However, distinguishing between a normal event and a cyberattack can be difficult because label information is mostly unavailable. Therefore, it is imperative to develop a threat intelligence system that operates more effectively without supervision, i.e., without a label. Additionally, reducing the false positive rate in cyber threat detection is a more promising step for a safer and more reliable environment. Also, the enormous number of features in the data for intrusion detection tasks sometimes results in significant computing costs. Therefore, a novel hybrid feature selection based unsupervised intrusion detection system is proposed, which is termed as HYbrid and Robust Intrusion DEtection (HYRIDE), that uses a wide variety of feature selection techniques to obtain the fewest, best possible features. The local outlier factor, elliptic envelope, and histogram-based outlier score models are then trained using these features to identify threats in network traffic automatically. As a result, HYRIDE can effectively and efficiently distinguish between normal events and intrusions. The proposed methodology is empirically evaluated using popular datasets such as Telemetry datasets of Internet of Things (IoT) services, Operating systems datasets of Windows and Linux, as well as datasets of Network traffic (TON_IoT), University of New South Wales-Network Benchmark (UNSW-NB15), and Canadian Institute of Cybersecurity Intrusion Detection System (CICIDS 2017).

1. Introduction

The fusion of information and operational technology has led to the increasing data density in Industry 4.0 (I4.0), causing significant cybersecurity challenges [1–4]. This surge has been complemented by sudden rise in cyberattacks which emphasizes the urgent need for robust cybersecurity measures to prevent misuse of data. Additionally, the rise of sophisticated hacker attacks, coupled with inadequate safety measures in the cyber space, necessitates protection against evolving threats and vulnerabilities, particularly in the context of the Internet of Things (IoT). As cyberattacks escalate, stakeholders utilizing IoT systems, especially large companies face substantial financial and operational risks including data corruption, system crashes, privacy breaches, and market losses. While many organizations have strengthened cybersecurity capabilities, the continuous growth of IoT-connected devices, estimated to reach

* Corresponding author.
E-mail addresses: amit.shukla@uwasa.fi (A.K. Shukla), pranabmuhuri@cs.sau.ac.in (P.K. Muhuri).

https://doi.org/10.1016/j.iot.2025.101492

Available online 7 January 2025

billions in the coming years which calls for a comprehensive protection system to mitigate the attractiveness of internet-based systems for cyberattacks. Such comprehensive cybersecurity system typically encompasses network security where Intrusion Detection Systems (IDS) play a crucial role in fortifying computer networks against external attacks. IDS is specifically adept at detecting various forms of malicious network communications with the primary objective of identifying malicious behavior within a network. Additionally, an IDS aids in discovering and identifying unauthorized system behavior, including unauthorized access modification, and destruction [5]. As a result, the development of an effective IDS to detect diverse cyberattacks and anomalies within a network is essential for strengthening overall system security.

Recent years have seen a significant integration of traditional machine learning (ML) techniques in the field of cybersecurity [6,7,8,9], to prevent intrusions due to the cyberattacks in the I4.0. With respect to ML, the detection of these intrusions is an application of anomaly detection where malicious activity or intrusions behavior are few and noticeably different from normal activity. This challenge is aided with the issue of high dimensionality of intrusion data, causing additional computational costs and noise and thus affecting overall model accuracy. It can be addressed to attain efficiency in accurately predicting the correct patterns of anomalies by selecting the "right" set of features [10,11]. Various feature selection (FS) approaches are present in the literature, which vary in the way optimal features are computed. To overcome the limitation of individual FS techniques, we utilize variety of FS approaches such as Independent Component Analysis (ICA), Principal Component Analysis (PCA), Chi-Squared (CHI2), Random Forest (RF), and Auto Encoder (AE) features. These techniques have three different types of taxonomies, where CHI2 and PCA features are embedded features, ICA and RF features are selected directly from the raw dataset, and AE features are deep learning [12,13], based features. Collectively, the combination of these FS techniques has the ability to handle high-dimensional data, identify complex relationships, and provide feature importance scores. These techniques align with the goals of intrusion detection and are suitable for wide set of dataset characteristics. Overall, the combination of these FS approaches provides an inclusive strategy to extract meaningful and relevant features from the data which eventually contributes to the overall effectiveness and adaptability of the intrusion detection approaches.

In a real-world applicative context, the existing IDS suffers from several challenges including lack of labelled data in IoT and industrial control systems where security breaches may go undetected for a longer period of time. Its attributed to the fact that intrusion activities are rare and distinct from normal events which makes the labeled attack data hard to identify. Also, traditional systems often produce high false positive rates, which results in increasing cost and unreliable cybersecurity measures. Therefore, its crucial to devise an unsupervised and effective IDS [14,15]. Unlike supervised ML approaches [16–18], the unsupervised approaches build IDSs without relying on labeled data. A few popular these approaches for detecting anomalies are Isolation Forest (IF), One-class SVM (OCSVM), Local Outlier Factor (LOF), Elliptic Envelop (EE), and Histogram based outlier Score (HBOS) models, etc. We have used the LOF, EE, and HBOS models to detect intrusions due to the reasons discussed as follows [19–21]:

- Their property in resolving complex types of intrusions (or anomalies) efficiently as compared to their counterparts.
- These algorithms are designed to have relatively low computational complexity and high scalability, making them suitable for large and high-dimensional datasets.
- They are recognized for their robustness and efficient performance in unsupervised settings.
- These algorithms may enhance the accuracy by capturing a wide range of anomaly patterns with density-based, distributional, and histogram approaches.

With this work, our goal is to build an efficient and robust unsupervised IDS that detects intrusions (or anomalies or outliers) effectively and reduces the false positive rate. Evidently, we study a hybrid feature selection based unsupervised intrusion detection approach for improved cybersecurity in an I4.0 system. The diversified approaches for unsupervised intrusion detections signifies the robustness of the proposed approach, capable of capturing different types of intrusions. Therefore, this paper proposes a hybrid and robust intrusion detection (HYRIDE) approach for detecting intrusions in I4.0, where optimal features are extracted using ICA, PCA, CHI2, RF, and AE, and intrusions are identified with unsupervised LOF, EE, and HBOS models. The hybrid nature of the approach signifies the exploitation of both unsupervised (ICA, PCA, AE) and supervised (CHI2, RF) FS techniques. HYRIDE is experimented on the wide variety of datasets in I4.0, including Telemetry datasets of Internet of Things (IoT) services, Operating systems datasets of Windows and Linux, as well as datasets of Network traffic (TON_IoT) [22], University of New South Wales-Network Benchmark (UNSW-NB15), and Canadian Institute of Cybersecurity Intrusion Detection System (CICIDS 2017). These datasets were chosen for their established benchmarks and wide acceptance in the research community, which provides a solid foundation for initial evaluation and comparison of FS and anomaly detection techniques. Additionally, these diverse datasets offer a robust foundation for developing and evaluating intrusion detection methods due to their diversity of attack types, realistic network traffic, comprehensive feature sets, and relevance to IoT and industrial environments. Consequently, the major contributions of this manuscript are described as follows:

1. A hybrid and robust intrusion detection system, HYRIDE, is proposed, which focuses on relevant features and optimal parameter tuning to lower the false positive rate, and thus enhancing reliability and accuracy in I4.0.
2. It utilizes important features extracted from ICA, PCA CHI2, RF, and AE approaches and identify intrusions using LOF, EE, and HBOS.
3. To assess the suitability of the proposed HYRIDE approach, an extensive experiments and detailed comparison has been performed with the state-of-the-art approaches.
4. The well-known real-world industrial datasets of TON_IoT, UNSW-NB15, and CICIDS 2017 are used for experimentation.

5. The comparative results show the superiority of HYRIDE as its effective and computationally faster in efficiently identifying intrusion detections.

This paper is organized into five sections. The introduction is presented in Section 1. Recent prominent methodologies and related work are discussed in Section 2. Section 3 describes the proposed HYRIDE approach using detailed flow chart, pseudocode, and step-by-step discussion. The corresponding experimental simulations and results are presented in Section 4. Section 5 concludes the paper with insights and detailed discussions.

2. Related work

This section explores the related research being done in the field of modern unsupervised intrusion detection systems in I4.0, and also discusses the recent novel algorithms and their extension in related tasks. Xu et al. [23] proposed a k-nearest neighbors (KNN)-based LOF algorithm, which is first applied to divide different areas for outlier densities with different distributions. A hierarchical adjacency order is proposed to calculate the neighborhood range link distance for assigning different weights to the data of different neighborhood regions. Omar et al. [24] and Ashari et al. [25] utilized LOF method to detect new malware anomalies and fraudulent claims in auto insurance, respectively. For detecting suspicious activities in insurance companies, useful features such as time, type of incident and frequency of financial losses are the most critical factors. Luan et al. [26] proposed an outlier detection for Deep Neural Networks (DNN) using IF and LOF method. It detects outliers in DNN by monitoring two or more of its hidden layers. It shows the effectiveness of LOF methods over IF in detecting accuracy, precision and recall. Peng et al. [27] detected electricity theft in advanced metering infra-structure (AMI) using the clustering based LOF method with only electricity data consumption. Gu et al. [28] used an online change point detection method based on an EE model with a DNN to identify real-time anomalies. Ashrafuzzaman et al. [29] proposed an intrusion detection system using the EE method in smart grid control systems. State estimation is a vital process in power transmission systems where stealthy false data injection attacks against state estimation using an EE have been developed. Aguilar et al. [30] proposed a decision tree (DT)-based AE for anomaly detection and compared it with OCSVM, EE, LOF, and gaussian mixture models. Pranto et al. [31] evaluated different FS techniques with a DT, RF, and naïve bias methods for network intrusion detection. Paulauskas et al. [32] detected network anomalies using a histogram-based outlier score method using the NSL-KDD dataset. Fernandez et al. [33] proposed different outlier detection-based supervised algorithms for classification and regression and HBOS, IF,

Table 1
Comparative summarization of the existing and the proposed approaches.

Refs.	FS Technique	ID Methodology	Dataset	Application	Remark
[34]	PIO-hill climbing, TS-PIO	IForest, LOF, OC-SVM	UNSW-NB15, NSL-KDD, KDDCUP99, Bot-IoT	Intrusion detection	Uses 4–24 features of hill climbing-PIO and 3–20 features of Tabu search-PIO in different dataset
[35]	Filter method (CHI2, PCC, MI), NSGA-II	SVM	TON-IoT dataset	Cyberattack detection	13,18 features are selected using NSGA-II and filter method
[36]	SMOTE, XG-Boost,	RF, DT, KNN, MLP, CNN, ANN	KDDCUP99, CIC-MalMem-2022	Intrusion detection	Uses 20 features are selected from the dataset with importance score using xgboost
[37]	PCA, AE, LDA	DFN, CNN, RNN, DT, LR, NB	UNSW-NB15, ToN-IoT, CSE-CIC-IDS2018	Intrusion detection	Set of different PCA, AE and LDA feature use in different classifier
[38]	PCA, RFE	NB, DT, KNN	NSL-KDD, CICIDS2017	Anomaly detection	13 features of PCA, RFE are used in NSL-KDD while 10 features are used for CICIDS2017 dataset
[39]	Wrapper method, DT based feature selection	SVM, ANN, KNN, RF, NB	UNSW-NB15, NSL-KDD	Intrusion detection	20 features of each dataset are selected using DT based FS
[40]	CHI2 feature	Bagging, XGBoost, RF, ET, AdaBoost	TON-IoT dataset	Intrusion detection	Chi2 features are used in ensemble learning based tree models
[41]	SMOTE, Mutual Information	RF, LR, KNN, DT, SVM	TON-IoT, NSL-KDD, CICIDS2017	Intrusion detection	15 features are extracted using mutual information method for all the datasets.
[42]	Filter based approach, 5 univariate filters and 3 multivariate filters	Multilayer perceptron, SVM, XGBoost, RF	CICIDS2017, CSE- CIC-IDS2018, CIC-TON-IOT	Intrusion detection	13, 6, 16 features of each dataset respectively used for model classification
[43]	Net flow based feature sets	Extra tree classifier	UNSW-NB15, BOT-IOT, TON-IOT, CSE-CIC-IDS2018	Intrusion detection	Net flow-based features are highly practical and scalable and new NIDS dataset is generated from existing dataset
[44]	Wrapper based FS, Tabu search based hybrid technique, CAT-S	Random Forest	TON-IOT	Intrusion detection	CAT-S approach reduces the total number of features by 70 %.
[45]	PCA, GIWRF	LR, DT, NB, RF, KNN, SVM	TON-IOT, UNSW-NB15, Bot-IoT	Intrusion detection	6 different classifiers used with 2 feature selection technique to find network intrusion attacks
HYRIDE	**ICA, PCA, CHI2, RF, AE**	**LOF, EE, HBOS**	**UNSW-NB15, TON-IoT, CICIDS2017**	**Intrusion detection**	**Nine features extracted from all FS technique and LOF, EE, HBOS is used to find anomalies**

and LOF are used for comparison.

Alghanam et al. [34] proposed local search-based pigeon inspired optimization (LS-PIO) for real world network intrusion detection (ID) datasets. LS-PIO algorithm uses a smaller number of features from the dataset using Tabu search (TS) and hill climbing search algorithm. LOF, OC-SVM and IF are used as a one-class classifier to detect intrusions attacks effectively. Dey et al. [35] proposed an NSGA-II based hybrid meta heuristic approach-based FS algorithm to detect cyberattacks in network intrusion based IoT enabled environment. A hybrid feature selection technique is used to extract filter-based features such as CHI2, Pearson's correlation coefficient and mutual information, which are further combined with NSGA-II algorithm. An SVM classifier is used in an optimized feature set to classify intrusions effectively. Talukder et al. [36] proposed a machine learning (ML) and deep learning (DL)-based models to implement secure network intrusion detection system (NIDS). For resolving the unbalancing problem in the dataset, Synthetic Minority Oversampling Technique (SMOTE) technique is used and XGBoost is used for FS. Sarhan et al. [37] proposed a generalized framework for robust NIDS using PCA, AE and LDA features which are used in various ML classifier such as deep feed forward (DFN), convolution neural network (CNN), Recurrent neural network (RNN), DT, Logistic Regression (LR) and Naive Bias (NB). Sah et al. [38] proposed real-time traffic-based IDS, where recursive feature elimination (RFE) and PCA features are used. For classifying real-time traffic IDS data, DT, NB and KNN algorithms are used. Umar et al. [39] focused on the impact of FS and normalization on IDS models using four distinct datasets and five ML algorithms. Findings reveal that normalization plays a more crucial role than FS in enhancing model performance and computational efficiency.

Awotunde et al. [40] proposed an ensemble approach for IDSs in IIoT networks where CHI2 statistical method is utilized for FS, and various ensemble classifiers, including XGBoost, Bagging, extra trees (ET), RF, and AdaBoost are applied to the TON_IoT datasets. Authors proposed to address the limitation of handling class imbalance as the future work of the study. Amaouche et al. [41] proposed an enhanced ID framework for Vehicular Ad Hoc Networks (VANETs), which utilizes mutual information for FS and SMOTE to handle class imbalance, with RF as the classifier. In future research author will address, the rapid mobility of nodes and dynamic network conditions for enhanced VANET security and resilience. Zouhri et al. [42] proposed an IDS by evaluating and comparing the impacts of various FS techniques on the performance of four classifiers (Multilayer Perceptron, SVMs, XGBoost, and RF) across three ID datasets. The five univariate filters and three multivariate filters, with XGBoost and RF, trained with multivariate methods such as CON and

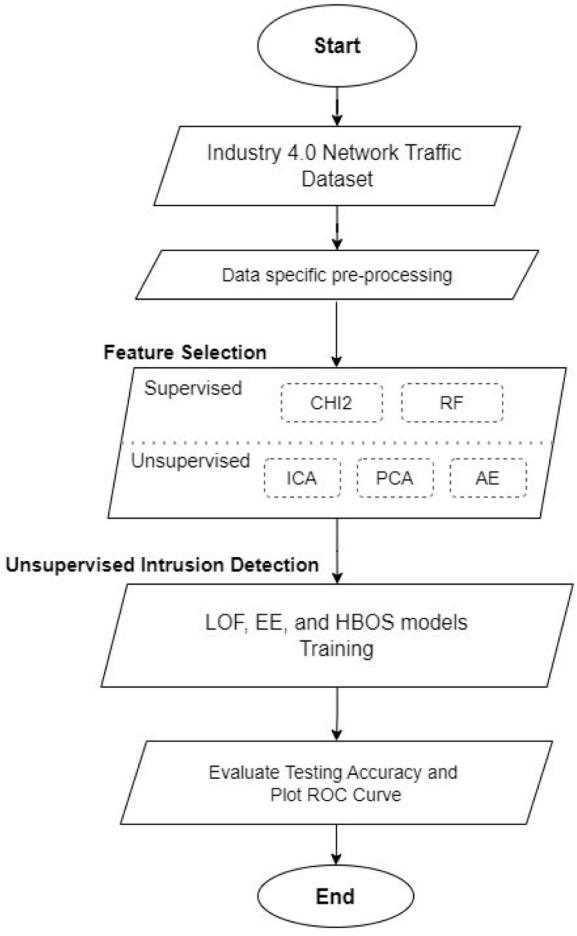


Fig. 1. Framework of the HYRIDE methodology.

DISR, effectively reduce the number of features without compromising classification performance. Sarhan et al. [43] proposed NetFlow-based standard feature sets for NIDS datasets, offering two variants with 12 and 43 features. Four widely used NIDS datasets are converted to these new feature sets, and the classification performance is assessed using an extra tree classifier. Nazir et al. [44] proposed a hybrid wrapper-based FS algorithm, named CAT-S, designed to enhance classification accuracy and reduce the number of features and false positive rates in the context of IoT networks. The algorithm integrates Cellular Automata (CA) with the TS meta-heuristic technique and employs an RF classifier for feature evaluation. Alhanaya et al. [45] proposed six ML models, including LR, DT, NB, RF, KNN, and linear SVM, which are evaluated with PCA and Gini Impurity-Based Weighted Forest (GIWRF) feature extraction methods across three global ToN-IoT, UNSW-NB15, and Bot-IoT datasets.

Table 1 describes and compares existing literature with the proposed approach in terms of FS techniques, intrusion detection (ID) methodology, datasets, application and important remark. The studies show the use of filter, wrapper, and embedded methods for FS, such as CHI2, MI, PCA, RFE, and DT-based approaches. These techniques are then combined with ML algorithms like SVM, RF, DT, KNN, and ANN to classify the network traffic. These works utilize diverse datasets, including UNSW-NB15, NSL-KDD, KDDCUP99, Bot-IoT, CIC-MalMem-2022, TON-IOT, and CICIDS2017. The analysis and outcome of these studies identify the need to address the challenge of evolving network threats and the increasing complexity of IDSs.

3. HYRIDE: HYbrid and Robust Intrusion DEtection approach

The hybrid nature of the proposed approach is envisioned in the combination of the feature section approaches including supervised and unsupervised FS approaches. It presents a comprehensive methodology to extract relevant features from the complex network traffic data of I4.0. Integration of varied types of intrusion detection algorithms ensures the robustness of the HYRIDE approach, thus overall, we have an effective and adaptable IDS. Fig. 1 graphically illustrates the several steps of the proposed approach, which is then explained in a detailed manner in this section.

3.1. Step-1: data preprocessing and optimal feature selection

Before feature selection steps, a pre-processing step is performed so that the dataset is in a clean, consistent, and appropriate state to undergo feature selection. This step before FS is necessary since it can enhance the performance and interpretability of the selected features. Though, the sub-steps under data pre-processing may vary depending on the data, we have utilized label encoding and performed some initial steps wherever essential including dropping null values row, removing zero variance values and identical columns. Also, we aggregated different attack types into a single "attack" class for the purpose of binary classification.

For feature selection, several approaches are utilized to extract optimal features. PCA converts the raw features into orthogonal transformed uncorrelated features, CHI2 features are based on top important features based on performing CHI2 test, RF features are top features while learning random forest that gives feature importance score of each feature, ICA feature are based on independent component analysis method, and AE features are autoencoder based neural network important features. We handle the noise in the data using appropriate feature selection and data scaling using min-max scaler. It is essential to select only necessary features as redundant and irrelevant features might reduce the anomaly detection capability to identify a network threat. The combination of these FS techniques with different unsupervised anomaly detection techniques makes a generalized methodology for intrusion detection that is more reliable.

3.2. Step-2: training of unsupervised intrusion detection models

In the second step, the LOF, EE, and HBOS model are trained in unsupervised way using the optimal feature sets from above step, by not providing any label information. The optimal hyperparameters of these models are selected based on the best training accuracy. The functioning of three models are described as follows:

- LOF is an unsupervised anomaly detection algorithm that gives better results when anomalies are mixed with normal data points [46] i.e., difficult to identify differences between normal points and anomalies. It uses both the K-nearest neighbors method and the local density method to find local anomalies in the dataset effectively. LOF is based on the concept of local density, which makes it more robust to global density variations. The choice of k is crucial for the performance of LOF as a small k can make the algorithm sensitive to noise, while a large k can make it less sensitive to outliers.
- On the other hand, EE is an unsupervised anomaly detection algorithm that detects intrusions effectively. This method selects anomalies with the assumption that the dataset is Gaussian distributed. It draws an ellipse around the dataset and thereby separates anomalies from the normal data. The shape and size of the elliptical envelope are determined by the mean and covariance matrix of the data. The EE can be used to detect both global and local outliers.
- HBOS is another unsupervised anomaly detection algorithm that detect anomalies by plotting the histogram for each feature and normalize each feature in the range of [0,1]. In HBOS, outlier score of each datapoint is calculated based on estimated density. For each feature, a histogram is plotted to calculate density estimate, using the frequency of samples in a bin with particular or variable bin width. If in a dataset there are d features then the Histogram based outlier score of a point x_j is defined as $HBOS(x_j) =$

$\sum_{i=0}^{d} \log\left(\frac{1}{hist_i(x_j)}\right)$. This algorithm is a powerful tool for identifying outliers or anomalies in the data that is not normally distributed or contains a large number of outliers.

3.3. Step-3: testing

In this step, LOF, EE, and HBOS models are evaluated for their intrusion detection capability using different FS techniques. After finding optimal parameter for the model, the evaluation process is repeated 20 times to assess the average testing accuracy or robustness and variability or standard deviation of the results. Maximum testing accuracy and training time is also computed from the set of five feature selection with 3 different outlier/intrusion detection algorithms.

3.4. Step-4: performance evaluation and analysis

In this step, the appropriate set of features that performed best in the three unsupervised intrusion detection models are analyzed to identify the best-performing feature and model combination in HYRIDE. The feasibility and importance of different types of feature set in each model are studied rigorously, and the findings are reported. Statistical analysis is performed and testing accuracy with a Receiver Operating Characteristic (ROC) curve are used as a performance evaluation metric. The False Positive Rate (FPR) and True Positive Rate (TPR) are plotted in the ROC curve. A ROC curve helps to illustrate the trade-off between true positive rate (sensitivity) and false positive rate at various classification thresholds. The Area Under Curve (AUC) quantifies the overall performance, with higher values indicating better discrimination and effective classification model.

The proposed HYRIDE approach is procedurally described in Algorithm 1. Our approach selects the best feature set for a model that gives a smaller computation cost because of a reduced set of optimal features. Moreover, it is a highly robust and effective unsupervised approach that works best with any type of real-world I4.0 data.

4. Experiments and results

This section is divided into two sections, where the first section discusses the parameters for the feature extraction methods and hyperparameters for the training process. The other section is further sub-classified by the datasets, where we have described the characteristics of the data, experiment specifications, results from the proposed HYRIDE approach and related analysis.

4.1. Feature extraction and hyperparameter details

PCA, ICA and CHI2 features are embedded features extracted from the datasets, using the scikit-learn library. The highest covariance principle is used to extract PCA features, independent component analysis method is used to extract ICA features while the Chi-Square test is used to obtain CHI2 features. RF features are extracted by training random forest and based on each raw feature's, feature importance score, top 9 features are extracted. AE features are autoencoder-based features that are trained with AE hidden layer's architecture as [64,180,100,180,64]. AE features are extracted by training autoencoder with batch-size 512, and number of training epochs as 100. AE features are optimized using Adam optimizer, where all layers are learned using ReLU activation functions

Algorithm 1
HYRIDE

Input: I4.0 network traffic Dataset (D_S): Training Dataset {D_{Train}} ∪ Testing Dataset {D_{Test}}, Parameters: {No._of_neigbours (n), Number of bins (b), Contamination Value (C_{value})}, Feature set: {F-RF, F-ICA, F-PCA, F-CHI2, F-AE}
Output: Testing accuracy in intrusion detection system
Training Stage:
1. Dataset specific pre-processing before feature selection
2. *F= {F-RF, F-ICA, F-PCA, F-CHI2, F-AE}, F_i= Extracted feature type from a feature set F, N =No. of extracted features of a particular feature F_i*
3. $F_i = (f_{ij})$, j=1 to N,
4. for each F_i:
5. Use min-max scaler as a data scaling for feature F_i
   a. Train Local outlier factor model M_1 with parameters as number of neighbors and Contamination value (C_{value}) range from [0.01:0.05:0.5]
6. Use min-max scaler as a data scaling for feature F_i
   a. Train an Elliptic Envelope model M_2 with parameters as C_{value} range from [0.01:0.05:0.5]
7. Use min-max scaler as a data scaling for feature F_i
   a. Train the HBOS model M_3 with parameters as number of bins and C_{value} range from [0.01:0.05:0.5]
8. for C_{value} in range of [0.01:0.05:0.5]:
   a. Training Dataset (D_{train}) is fitted with respective parameters of model {M1, M2, M3} to find optimal parameters that give the best training accuracy in models.
   b. Evaluate optimal parameters of each LOF, EE, and HBOS model in training different C_{value}.
9. Output:
   Model {M_1, M_2, M_3} with optimal parameter (n, b, C) using features set:
   {F-RF, F-ICA, F-PCA, F-CHI2, F-AE}
Testing Stage:
1. for each trained model $M_i(i \in \{1,2,3\})$:
   a. Evaluate average testing accuracy using D_{TEST} Dataset with trained model M_i

except the last layer, which is learned using the sigmoid activation function.

In the training process, for the LOF model, the no of neighbors and contamination value are the hyperparameters. The percentage of abnormalities or anomalies in the dataset is used to set as the contamination value. The contamination value in the context of outlier detection algorithms like LOF, EE, and HBOS typically refers to the estimated proportion of outliers in the dataset. It is not directly related to the percentage of attack-related samples but rather to the estimated proportion of outliers within the entire dataset. For finding the best training accuracy, the optimal no of neighbors is selected by an extensive hyperparameter tuning in the range of [1,2,3,4,5,10,15,20,25,30,35,40,50, 60, 70] and optimal contamination value is selected based in the range of [0.01, 0.015, 0.02, 0.025, 0.03, 0.035, 0.04, 0.045, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.15, 0.20, 0.25, 0.30, 0.33,0.35, 0.40, 0.42, 0.44, 0.46, 0.48, 0.50]. In the EE model, the contamination value is a hyperparameter, obtained in similar fashion as LOF model. The number of bins and contamination value are hyperparameters for HBOS models which are fine tuned. Here, the optimal number of bins is selected from the range of [10,20,30,40,50, 60, 70, 80] by hyperparameter tuning [47], and the optimal contamination value is chosen in similar way as LOF and EE model.

In LOF, the number of neighbors and contamination value were tuned using grid search to find the optimal values that maximized performance. A grid search is a comprehensive search over a predefined grid of hyperparameter values. In EE method, the contamination values are also identified using grid search method. Similarly, this method is again used in HBOS for finding or tuning the number of bins and contamination value.

4.2. Datasets, experiments and results

This section contains the details about every dataset considered for experimentations and respective results obtained with the HYRIDE. All the algorithms are implemented on the Anaconda (Spyder) IDE in python and experiments are conducted on a workstation with an Intel Core i7- 8700 CPU at 3.20 GHz and memory space of 16 GB.

4.2.1. TON- IoT dataset

TON-IoT dataset is a new I4.0 dataset originated from UNSW university Australia [48]. It contains different types of attacks such as DoS, DDoS, ransomware and Against the web applications, which are collected from heterogeneous sources of IoT and IIoT sensors. They were collected in the packet capture (PCAP) format, log files and CSV files of ZEEK (bro) tool. It has total 461,043 instances with 37 raw features which are divided into train and test dataset. All the categorical values are transformed using a label encoding approach. The training dataset has a total of 368,834 instances with 240,027 normal events and 128,807 attack events, as depicted in Table 2. Testing dataset, however, have total 92,209 instances with 59,973 normal events and 32,236 attack events.

Fig. 2 further depicts the training performance of these algorithms when trained with features from PCA, ICA, CHI2, RF, and AE for different contamination values. The optimal contamination value is selected when the intrusion detection models achieve the best training accuracy. For instance, when trained with LOF, AE features achieves the best training accuracy for contamination value 0.15 by using the optimal number of nearest neighbors as 50. In PCA features, the best training accuracy is found at 0.10 contamination value by using the optimal number of neighbors as 25. In ICA features, the best training accuracy is found at contamination value 0.10 with the optimal number of neighbors as 25. In CHI2 features, the best training accuracy is found at the contamination value of 0.015 when the optimal number of neighbors is set as 20. In RF features, the best training accuracy is at 0.15 contamination value when the optimal number of neighbors is set as 70. Table 3 shows the optimal contamination values and best training accuracies for the LOF, EE, and HBOS models. EE model achieves best training accuracy for, PCA, ICA, CHI2, RF, and AE features at 0.025, 0.025, 0.01, 0.08, and 0.025 contamination values respectively. For HBOS training, all the FS selection approaches achieve best training accuracy at 0.01 contamination value, with optimal number of bins as 60, 80, 30, 20, and 10 for PCA, ICA, CHI2, RF, and AE, respectively.

LOF when trained with the PCA features returned the best training accuracy of 0.6824 with 25 number of neighbors and 0.10 contamination value. With EE and HBOS, better training accuracy is achieved with RF features, which is 0.6622 and 0.6507, respectively. The optimal contamination value with EE is 0.080 and with HBOS, it is 0.01 with optimal number of bins of 20. The average testing accuracy values of 20 different iterations are computed for LOF, EE, and HBOS models at their respective optimal hyperparameters, and are also shown as a bar plot in Fig. 3. Table 4 compiles the average testing accuracy with standard deviation (st. d.) for 20 repetitive runs of all three LOF, EE, and HBOS models, average training accuracy with standard deviation (st.d.), maximum testing accuracy, and average computation time. With the LOF model, PCA and ICA features attained the highest average testing accuracy in comparison with EE and HBOS. PCA and ICA features also attained best average testing accuracy with the EE model as well. With HBOS, RF features have the highest average testing accuracy. Collectively, PCA, ICA and RF features consistently performed well based on average testing accuracy. If we analyze the maximum test accuracy, LOF trained on ICA features, the EE trained on ICA and PCA features, and HBOS trained on RF features returned 0.6535, 0.6518, and 0.6503 values, respectively. In terms of average time execution, the HBOS model with all feature selection techniques consumes the least training time in comparison to LOF and EE models. Among all, AE features are computationally more expensive than PCA, ICA, RF, and CHI2 features, as AE features are deep learning-

Table 2
TON-IoT dataset information.

Dataset	Total features	Total events	Normal events (0)	Attack events (1)
Training	37	368,834	240,027	128,807
Testing	37	92,209	59,973	32,236

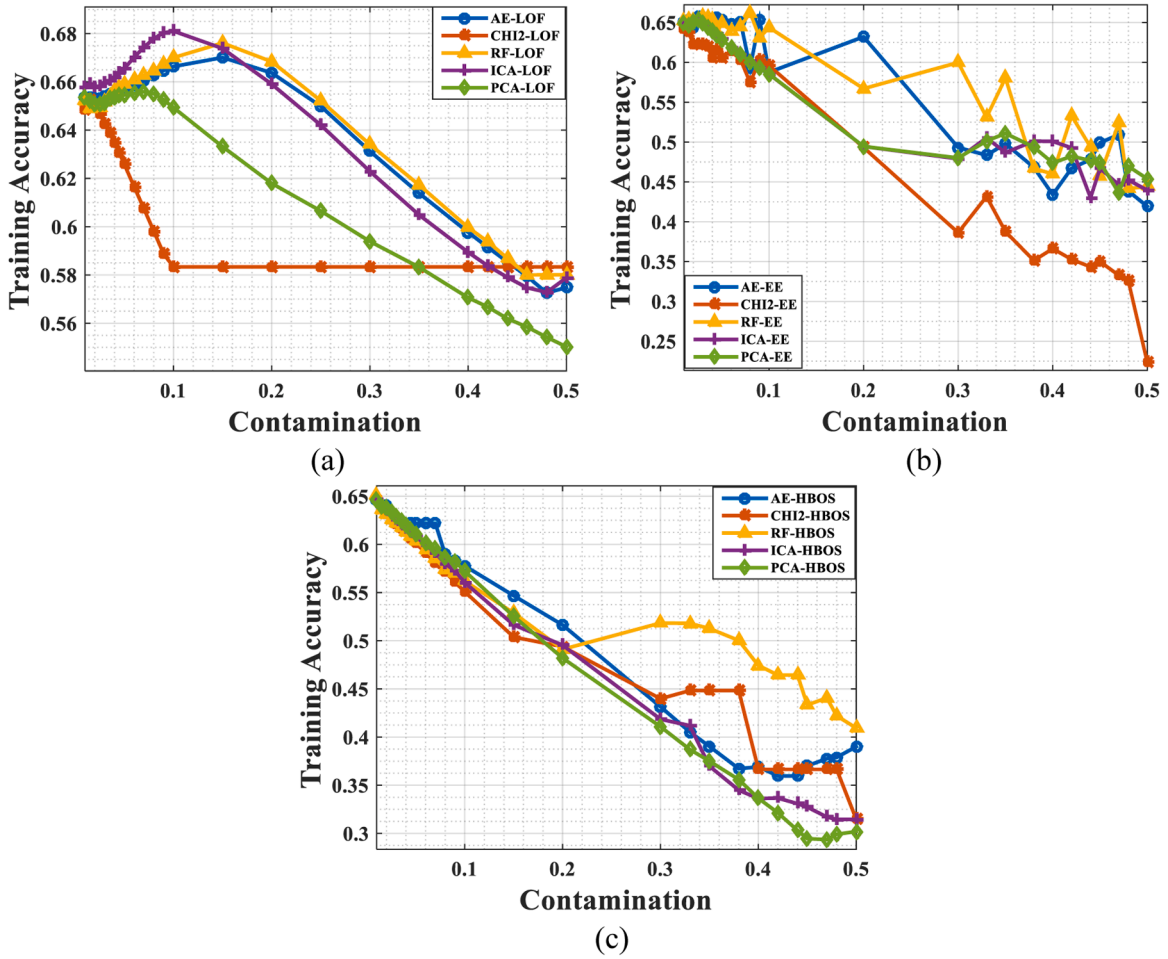


Fig. 2. Training accuracy of PCA, ICA, CHI2, RF, AE with (a) LOF, (b) EE, and (c) HBOS.

Table 3
Best training accuracy and optimal hyperparameters for LOF, EE and HBOS when trained with PCA, ICA, CHI2, RF, AE features.

Methods	LOF		EE		HBOS	
FS	Best Training Accuracy	(Optimal no of neighbors, Optimal Contamination value)	Best Training Accuracy	Optimal Contamination value	Best Training Accuracy	(Optimal no of bins, Optimal contamination value)
PCA	0.682445	(25, 0.10)	0.653091	0.025	0.646112	(60, 0.01)
ICA	0.681035	(25, 0.10)	0.653095	0.025	0.64561	(80, 0.01)
CHI2	0.652445	(20, 0.015)	0.641524	0.01	0.646204	(30, 0.01)
RF	0.676114	(70, 0.15)	0.662184	0.08	0.650696	(20, 0.01)
AE	0.670079	(50, 0.15)	0.657461	0.025	0.644571	(10, 0.01)

based features. Standard deviation is highest for EE model in multiple runs, while it is least for HBOS and LOF. Hence, EE predictions are likely to be varied for different runs.

It can be seen from the experiments that unsupervised feature selection using ICA, PCA results in comparable performance with HBOS, EE and LOF models, when compared with supervised features viz. CHI2 and RF features. Table 4 shows that PCA and ICA features with the LOF model achieves the best average testing accuracy whereas HBOS model gives the lowest average training time. In conclusion, ICA with LOF returned overall highest average test accuracy of 0.6535 with time consumption of 32.64 secs, however, CHI2 with HBOS was the fastest with 0.64 s and test accuracy of 0.6456. ROC curve for LOF, EE and HBOS are plotted in Fig. 4(a) - (c), where the maximum AUC indicates the best class separation in the dataset.

4.2.2. UNSW-NB15 dataset

The UNSW-NB15 dataset has been developed at UNSW university in Australia. Originally, this dataset contained 49 features in

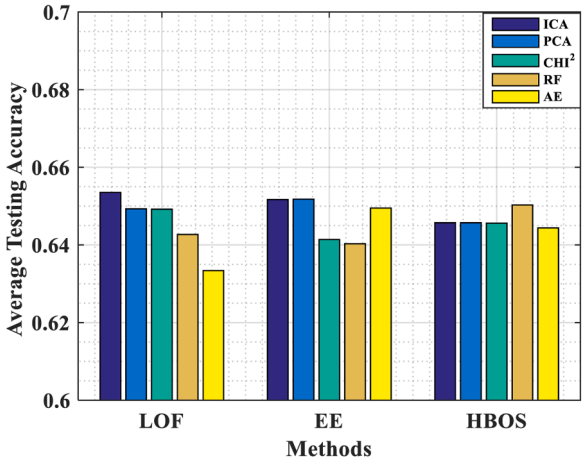


Fig. 3. Average testing accuracy in LOF, EE and HBOS models.

Table 4
Comparison of LOF, EE AND HBOS models for TON_IoT Dataset.

Method	Parameters	Total features/ Resulting dimensions	Average Train Accuracy ± st. d. (20 iterations)	Average Test Accuracy ± st. d. (20 iterations)	Maximum Test Accuracy	AVG. Train Time (second)
ICA + LOF	c= 0.10, no. _neighbor = 25	9	0.6810 ± 0.0	0.6535± 0.0000	0.6535	32.64
PCA + LOF	c= 0.10, no. _neighbor = 25	9	0.6824 ± 0.0	0.6493 ± 0.0000	0.6493	28.58
CHI2 + LOF	c= 0.015, no. _neighbor = 20	9	0.6524± 0.0	0.6492± 0.0000	0.6492	249.73
RF+ LOF	c= 0.15, no. _neighbor = 70	9	0.6761 ± 0.0	0.6427± 0.0000	0.6427	88.36
AE +LOF	c= 0.15 no. _neighbor = 50	100	0.6701 ± 0.0	0.6334± 0.0000	0.6334	1323.53
ICA + EE	c= 0.025	9	0.6530 ± 0.0001	0.6517 ± 0.0001	0.6518	38.21
PCA + EE	c= 0.025	9	0.6531 ± 0.0000	0.6518± 0.0000	0.6518	36.37
CHI2 + EE	c= 0.01	9	0.6417 ± 0.0005	0.6414 ± 0.0006	0.6420	23.57
RF+ EE	c= 0.08	9	0.6413 ± 0.0115	0.6403 ± 0.0114	0.6517	49.18
AE + EE	c= 0.025	100	0.6504 ± 0.0105	0.6495 ± 0.0103	0.6598	1059.93
ICA + HBOS	c= 0.01, bin_size=80	9	0.6456 ± 0.0	0.6457 ± 0.0000	0.6457	3.28
PCA + HBOS	c= 0.01, bin_size = 60	9	0.6461 ± 0.0	0.6457 ± 0.0000	0.6457	1.21
CHI2 + HBOS	c= 0.01, bin_size= 30	9	0.6462 ± 0.0	0.6456± 0.0000	0.6456	0.64
RF+ HBOS	c= 0.01, bin_size = 20	9	0.6507 ± 0.0	0.6503 ± 0.0000	0.6503	27.75
AE + HBOS	c= 0.01, bin_size =10	100	0.6446 ± 0.0	0.6444 ± 0.0000	0.6444	1005.76

which missing values were removed, that led the available version of UNSW-NB15 to contain 42 raw features. Among these features, all those categorical features whose unique value counts are <100, they are processed using one hot encoder method. Otherwise, if the count of the unique values in a categorical variable is >100, then the label encoder method is used to transform categorical variables into numerical values.

As mentioned in Table 5, UNSW-NB15 dataset consist of a training dataset and a testing dataset. The training dataset has 42 features with 175,341 events, of which there are 56,000 normal events labelled as '0′ and 119,341 attack events labeled as '1′. The testing dataset has 42 features with 82,332 events, of which 37,000 are normal events, and 45,332 are attack events. Here, the intrusion attack points are atleast 200 % more than normal data points, thus to align with the utilized anomaly detection approaches, we considered normal data points as anomalies during training, since anomalies are those instances which occurs in little quantity.

Fig. 5(a) depicts the performance of LOF using AE, PCA, ICA, CHI2, and RF with different contamination values at the optimal number of neighbors. Here, the LOF training with AE features achieves the best training accuracy for contamination value 0.33 by using the optimal number of nearest neighbors as 1. In AE features, after the 0.33 contamination value, the training accuracy is the same, so we chose the optimal contamination value as 0.33. Similarly, with PCA features, best training accuracy is found at 0.33 contamination value by using the optimal number of neighbors as 2. Table 6 shows the optimal hyperparameter values for LOF, EE, and

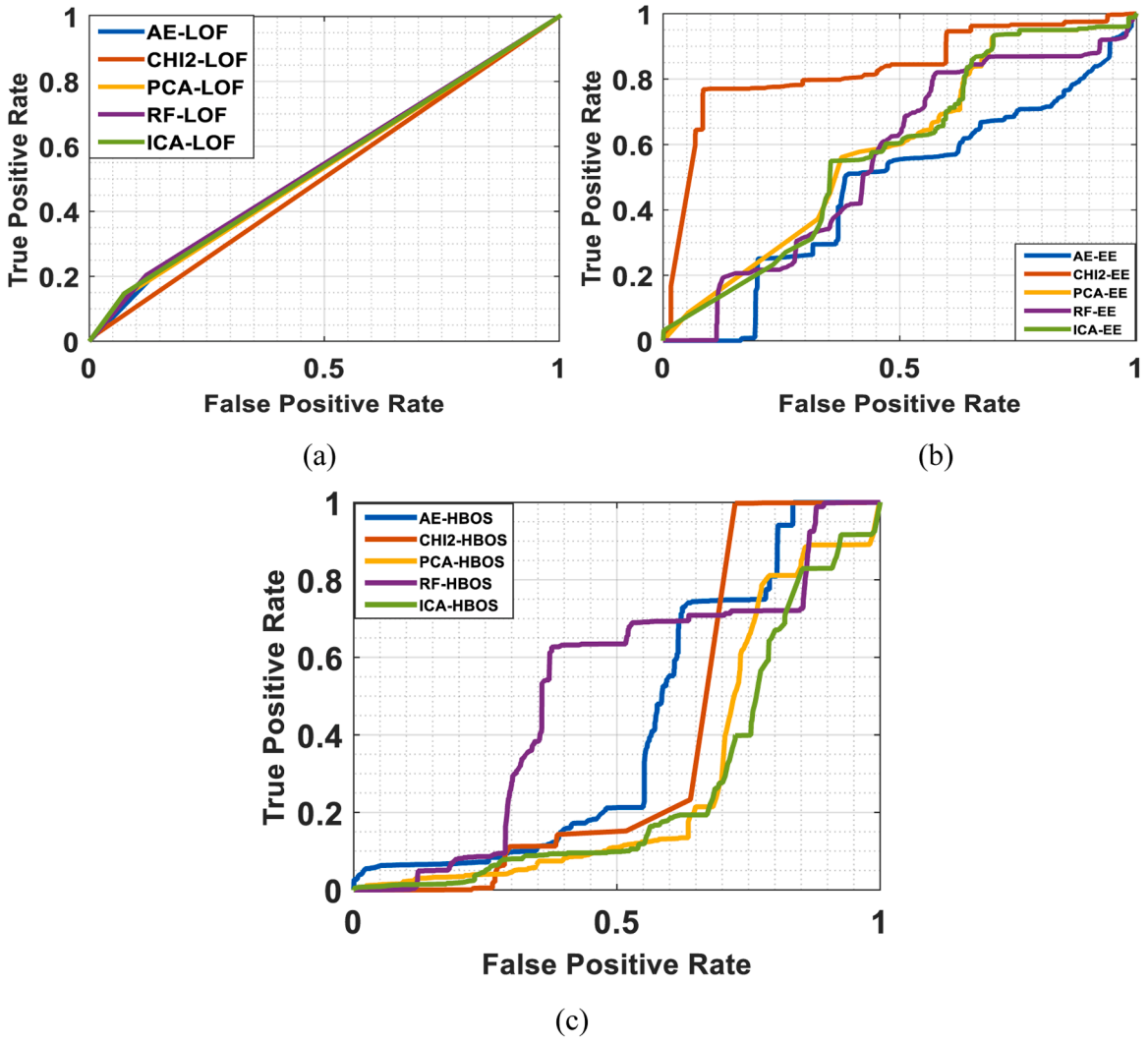


Fig. 4. ROC curve for TON-IoT dataset generated after HYRIDE.

Table 5
UNSW-NB15 dataset information.

UNSW-NB15 DATASET	Total features	Total events	Normal events	Attack events
Training Dataset	42	175,341	56,000	119,341
Testing Dataset	42	82,332	37,000	45,332

HBOS models along with the best training accuracy. In LOF, among all FS methods, the best training accuracy of 0.7346 is obtained with CHI2 features. The EE model achieves the best training accuracy for the PCA, ICA, CHI2, RF, and AE features at 0.35, 0.48, 0.35, 0.08, and 0.30 contamination values, respectively. Among them, EE model trained with CHI2 features returned highest training accuracy of 0.9012. The Same behavior is shown by CHI2 features when trained with HBOS, where the best training accuracy was 0.8365. Therefore, among all FS techniques, CHI2 attained the highest training accuracy for all contamination values in LOF, EE, and HBOS models. Also, EE model has the highest training accuracy using CHI2 features. Other than CHI2, RF features also performed better than the rest three FS approaches of PCA, ICA and AE features in all three LOF, EE, and HBOS models.

The average training and testing accuracies of 20 different iterations are computed for LOF, EE, and HBOS models at their respective optimal hyperparameters, and shown in Table 7. This table also enlists the maximum testing accuracy and average computation time for all the considered approaches. In the HBOS model, across all FS approaches, CHI2 features attained the highest average testing accuracy, even in comparison with EE and LOF models. In EE and LOF model, PCA attained a highest average testing accuracy of 0.7106 and 0.6754, respectively. Similar to the behavior shown in TON-IoT datasets, the HBOS model with all FS

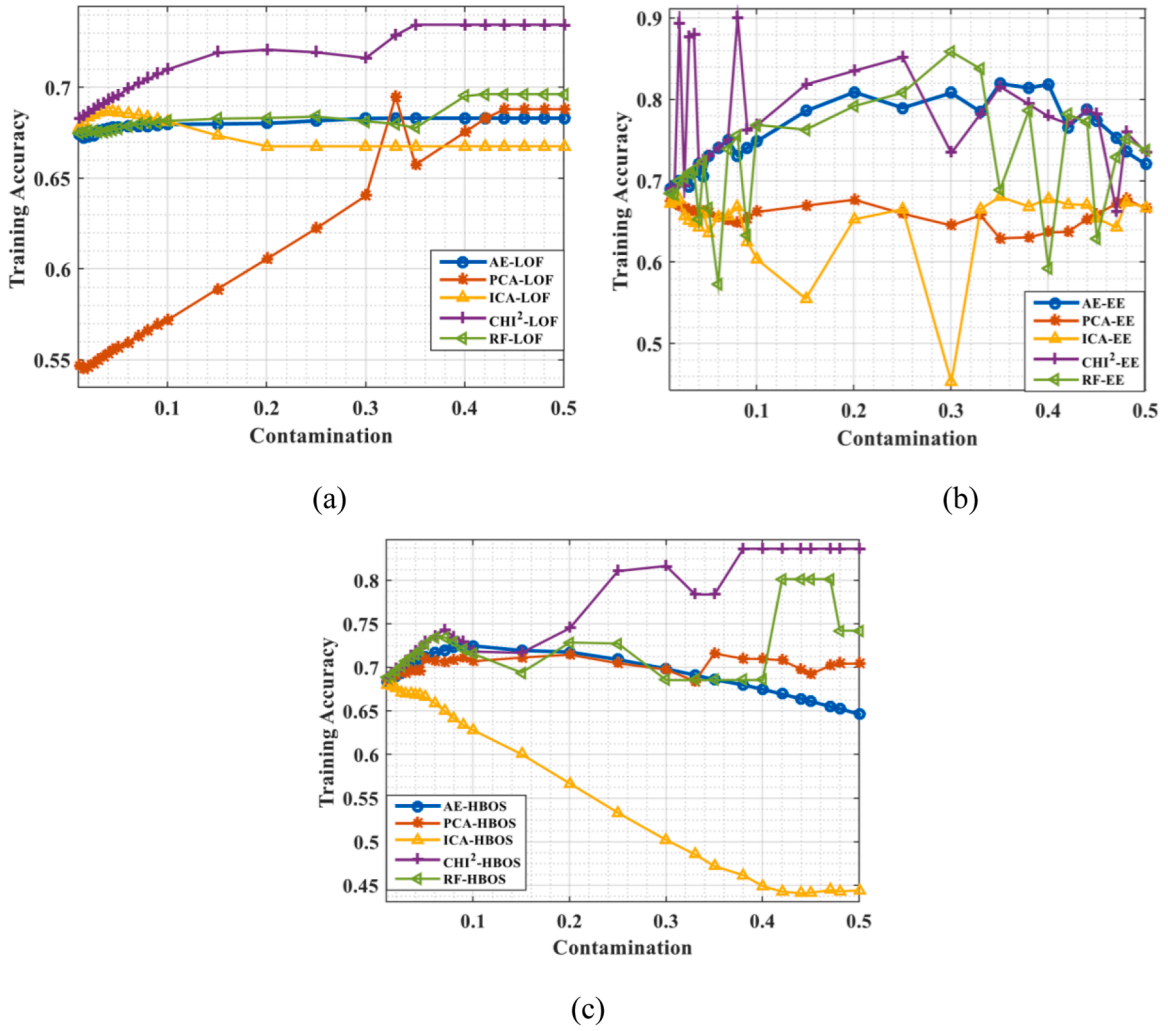


Fig. 5. Training accuracy with different FS approaches in (a) LOF, (b) EE, and (c) HBOS.

Table 6
Best training accuracy and optimal hyperparameters for LOF, EE and HBOS methods when trained on PCA, ICA, CHI2, RF, and AE features.

Methods	LOF		EE		HBOS	
FS	Best Training Accuracy	(Optimal no of neighbors, Optimal Contamination value)	Best Training Accuracy	Optimal Contamination value	Best Training Accuracy	(Optimal no of bins, Optimal contamination value)
PCA	0.6951	(2, 0.33)	0.6816	0.48	0.7160	(10, 0.35)
ICA	0.6871	(1,0.04)	0.6804	0.35	0.6797	(70, 0.01)
CHI2	**0.7346**	(70, 0.35)	**0.9012**	0.08	**0.8365**	(10, 0.38)
RF	0.6964	(60, 0.42)	0.8644	0.30	0.8016	(10, 0.42)
AE	0.6830	(1, 0.33)	0.8197	0.35	0.7249	(80, 0.10)

techniques again produced the least training time in comparison to LOF and EE models. Among all feature selection techniques, AE features are naturally computationally more expensive than PCA, ICA, RF, and CHI2 features. Standard deviation is highest for EE model in multiple runs, while its least for HBOS and LOF. Fig. 6 shows the average testing accuracy bar plot with the proposed HYRIDE approach.

The ROC curves corresponding to the considered approaches are plotted in Fig. 7(a)-(c). As depicted in Fig. 7(a), PCA and CHI2 features have large and almost similar AUC for the LOF model. Fig. 7(c), depicts the large AUC in HBOS model for using CHI2 features. In conclusion, the results show that CHI2 features with the HBOS model achieves the best average testing accuracy with comparatively lowest average training time (1.38 s).

Table 7
Comparison of LOF, EE, and HBOS models with different feature sets on UNSW-NB15 Dataset.

Method	Parameters	Total features/ Resulting dimensions	Average Train Accuracy ± st. d. (20 iterations)	Average Test Accuracy ± st. d. (20 iterations)	Maximum Test Accuracy	Avg. Train Time (Sec)
ICA + LOF	c = 0.04, no. _neighbor= 1	9	0.6871 ± 0.0	0.5636 ± 0.0	0.5636	9.48
PCA + LOF	c= 0.33, no. _neighbor= 2	9	0.6951 ± 0.0	0.6754 ± 0.0	0.6754	5.46
CHI2 + LOF	c = 0.35, no. _neighbor= 70	9	0.7346 ± 0.0	0.6730 ± 0.0	0.673	193.84
RF+LOF	c = 0.42, no. _neighbor= 60	9	0.6964 ± 0.0	0.6637 ± 0.0	0.6637	28.7
AE +LOF	c = 0.33, no. _neighbor= 1	100	0.6830 ± 0.0	0.6370 ± 0.0	0.637	1454.86
ICA +EE	c = 0.35	9	0.6118 ± 0.1037	0.5671 ± 0.0788	0.6275	38.75
PCA + EE	c= 0.48	9	0.6790 ± 0.00582	0.7106 ± 0.00317	0.716	49.33
CHI2 + EE	c = 0.08	9	0.7563 ± 0.0315	0.6242 ± 0.0258	0.65	12.87
RF+EE	c = 0.30	9	0.7448 ± 0.1242	0.6769 ± 0.0690	0.7459	19.57
AE + EE	c = 0.35	100	0.7540 ± 0.0268	0.6958 ± 0.01535	0.7112	1123.11
ICA + HBOS	c = 0.01, bin_size=70	9	0.6797 ± 0.0	0.5368 ± 0.0	0.5368	**1.01**
PCA + HBOS	c= 0.35, bin_size=10	9	0.7160 ± 0.0	0.5863 ± 0.0	0.5863	3.21
CHI2 + HBOS	c = 0.38, bin_size=10	9	0.8365 ± 0.0	**0.7527**± 0.0	0.7527	1.38
RF+ HBOS	c = 0.42, bin_size=10	9	0.8016 ± 0.0	0.7385 ± 0.0	0.7385	4.42
AE + HBOS	c = 0.10, bin_size=80	100	0.7249 ± 0.0	0.6006 ± 0.0	0.6006	310.35

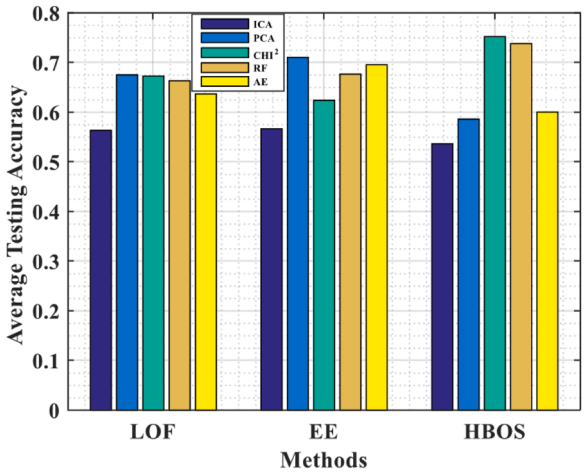


Fig. 6. Average testing accuracy in LOF, EE and HBOS models.

4.2.3. CICIDS 2017

The CICIDS2017 dataset originates from the Canadian Institute for Cybersecurity (CIC), developed as a comprehensive benchmark dataset for cybersecurity research and analysis. It comprises a diverse collection of network traffic data, encompassing both normal network behaviour and various types of cyberattacks, including but not limited to DoS, DDoS, brute force, and infiltration attempts. This dataset was carefully crafted to simulate real-world network scenarios and is instrumental in evaluating the efficacy of intrusion detection and prevention systems.

As mentioned in Table 8, CICIDS2017 dataset consists of a training and a testing dataset. Originally, this dataset had 79 features and after pre-processing, the training dataset returned 66 features with 2016,638 events, of which there are 340,339 normal events labelled as '0′ and 16,766,299 attack events labelled as '1′. The testing dataset also has 66 features with 504,160 events, of which 85,402 are normal events, and 418,758 are attack events. In CICIDS 2017 training dataset, the intrusion attack points are around 500 % more than normal data points as depicted in Table-8. There is clearly imbalance in this dataset since attack events are significantly higher than the normal events. To handle this situation, the used unsupervised intrusion detection approaches such as LOF, EE and HBOS consider normal data points as anomalies during training. This will ensure robustness considering that the anomalies are those instances which occur in little quantity. In preprocessing step of CICIDS 2017 dataset, we removed zero variance columns, dropped null

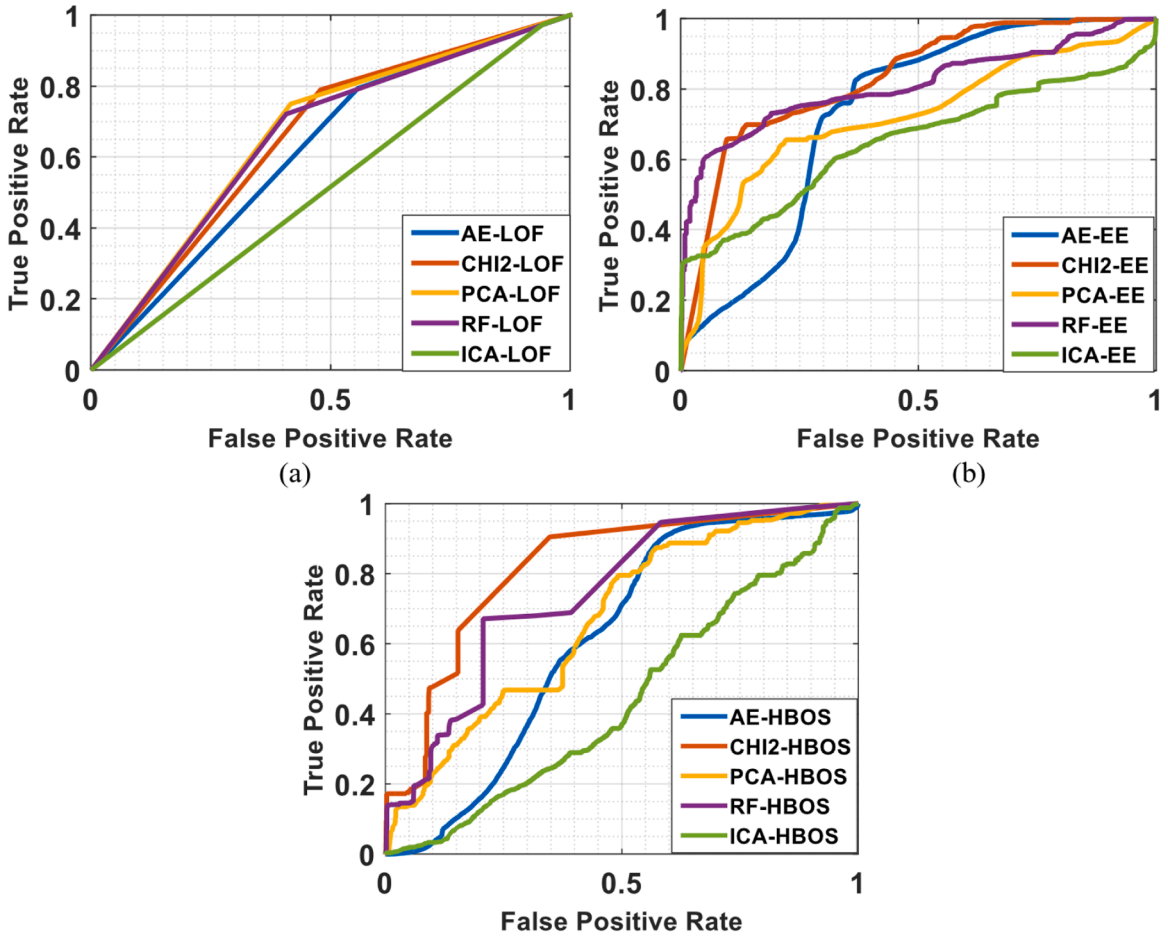


Fig. 7. ROC curve of five different feature sets in (a): LOF, (b): EE, and (c): HBOS.

Table 8
CICIDS 2017 dataset information.

CICIDS 2017 DATASET	Total features	Total events	Normal events (0)	Attack events (1)
Training Dataset	66	2016,638	340,339	1676,299
Testing Dataset	66	504,160	85,402	418,758

values rows and removed identical columns in the CICIDS2017 dataset.

Fig. 8 depicts the performance in terms of training accuracy of LOF, EE, and HBOS using AE, PCA, ICA, CHI2, and RF with different contamination values. In Fig. 8(b), EE model achieves best training accuracy for AE, PCA, ICA, CHI2, and RF features at 0.01, 0.07, 0.07, 0.06, and 0.10 contamination values, respectively. Similarly, Fig. 8(c) shows the training accuracy plot for HBOS model. For HBOS, among all feature selection techniques, RF attained the highest training accuracy of 0.9087 for all contamination values in LOF and HBOS models. EE model also has the highest training accuracy of 0.9074 using RF features for the contamination values of 0.10. For LOF, CHI2 returned a highest training time of 0.8288 with optimal no of neighbors and contamination values as 1 and 0.01, respectively. Table 9 compiles the training accuracies using different feature selection techniques in each of the three intrusion detection models with best selected hyperparameters.

Further, Table 10 compiles the hyperparameters, average training and testing accuracies, and average training time for CICIDS dataset. The average values are computed over 20 independent runs. The table shows that the RF features with the HBOS model achieves the best average testing accuracy. The CHI2 on the other hand gave the least training time of 40.41 secs, however, it produced less average testing accuracy of 0. 8812. Overall, the HBOS model with all feature selection techniques consumes the least training time, as is the behavior with other two datasets discussed in above sections.

Fig. 9 depicts the average testing accuracy bar plot for CICIDS with the proposed HYRIDE approach. In the HBOS model, across all FS approaches, RF features attained the highest average testing accuracy. Same behaviour of RF is seen with EE model as well. In LOF, almost all FSs attained approximately similar average test accuracy.

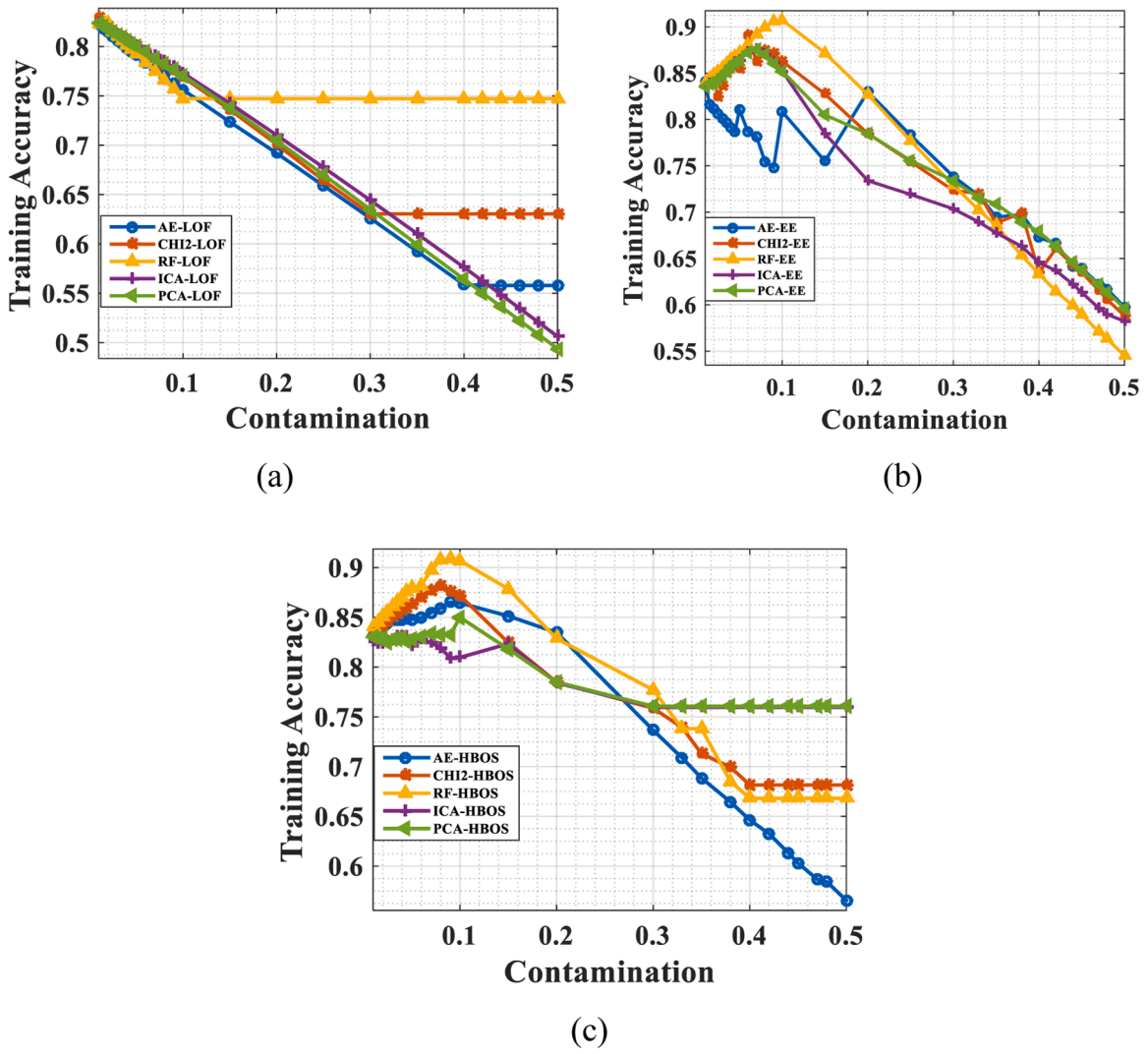


Fig. 8. Training accuracy with different contamination values in (a) LOF, (b) EE, and (c) HBOS.

Table 9
Best training accuracy and hyperparameters of PCA, ICA, CHI2, RF, and AE feature sets using LOF, EE and HBOS.

Methods	LOF		EE		HBOS	
FS	Best Training Accuracy	(Optimal no of neighbors, Optimal Contamination value)	Best Training Accuracy	Optimal Contamination value	Best Training Accuracy	(Optimal no of bins, Optimal contamination value)
PCA	0.8239	(5, 0.01)	0.8756	0.07	0.8500	(10, 0.1)
ICA	0.8238	(35,0.02)	0.8761	0.07	0.8311	(10, 0.035)
CHI2	0.8288	(1, 0.01)	0.8912	0.06	0.8818	(40, 0.08)
RF	0.8237	(1, 0.02)	0.9074	0.10	0.9087	(10, 0.09)
AE	0.8229	(1, 0.01)	0.8412	0.01	0.8653	(10, 0.09)

5. Conclusion

5.1. Discussion

I4.0-led smart industries produce large amounts of data from different sensors, and the modern cyber threat increases due to the leakage of information from these sensors. This necessitates the ever-increasing need to rigorously analyze network traffic in a cyber-physical system for detecting newly designed intrusions without any label information. Thus, this paper studied and proposed a novel

Table 10
Comparison of LOF, EE and HBOS results.

Method	Parameters	Total features/ Resulting dimensions	Average Train Accuracy ± st. d. (20 iterations)	Average Test Accuracy ± st. d. (20 iterations)	Maximum Test Accuracy	AVG. Train Time
ICA + LOF	c= 0.02, no. _neighbor = 35	9	0.8238 ± 0.0	0.8224 ± 0.0	0.8224	574.90
PCA + LOF	c= 0.01, no. _neighbor = 5	9	0.8239 ± 0.0	0.8231 ± 0.0	0.8231	659.02
CHI2 + LOF	c= 0.01, no. _neighbor = 1	9	0.8288 ± 0.0	0.8250 ± 0.0	0.8250	122.80
RF + LOF	c= 0.0.02, no. _neighbor = 2	9	0.8237 ± 0.0	0.8175 ± 0.0	0.8175	4356.73
AE + LOF	c= 0.01, no. _neighbor = 1	100	0.8229 ± 0.0	0.8216 ± 0.0	0.8216	8765.21
ICA + EE	c= 0.07	9	0.8759 ± 0.0002	0.8751 ± 0.0001	0.8752	780.26
PCA + EE	c= 0.07	9	0.8759 ± 0.0001	0.8751± 0.0001	0.716	523.67
CHI2 + EE	c= 0.06	9	0.8684 ± 0.0144	0.8679 ± 0.0146	0.8825	145.85
RF + EE	c= 0.10	9	0.9012 ± 0.0058	0.9013 ± 0.0058	0.9071	4095.38
AE + EE	c= 0.01	100	0.8239± 0.0059	0.8233 ± 0.0058	0.8298	6778.72
ICA + HBOS	c = 0.035, bin_size=10	9	0.8311 ± 0.0	0.8307 ± 0.0	0.8307	730.95
PCA + HBOS	c= 0.1, bin_size=10	9	0.8500 ± 0.0	0.8493 ± 0.0	0.8493	176.56
CHI2 + HBOS	c = 0.08, bin_size=40	9	0.8818 ± 0.0	0.8812 ± 0.0	0.8812	40.41
RF + HBOS	c = 0.09, bin_size=10	9	0.9087 ± 0.0	**0.9067 ± 0.0**	0.9067	4157.78
AE + HBOS	c = 0.09, bin_size=10	100	0.8653 ± 0.0	0.8643 ± 0.0	0.8643	6366.78

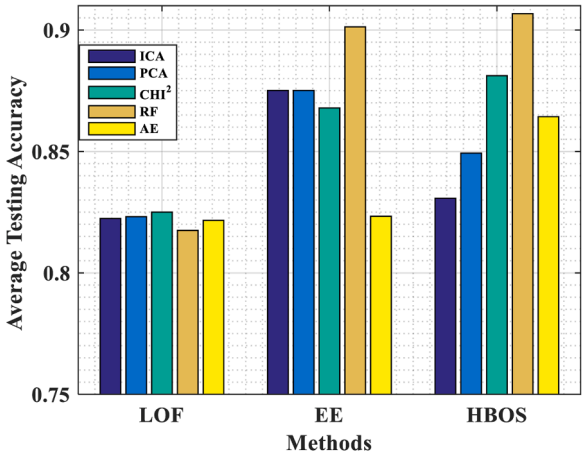


Fig. 9. Average testing accuracy in LOF, EE and HBOS model.

approach to enhance cybersecurity in I4.0 with a hybrid and robust intrusion detection algorithm, termed as HYRIDE.

HYRIDE approach uses sophisticated unsupervised intrusion detection algorithms with optimum features extracted from wide range of feature selection approaches to detect ever growing cyberattacks in I4.0 system. The most relevant extracted features help in significantly reducing the dimensionality and computational complexity of the data. Different characteristics of real-world data has inherent complexities which makes it difficult to extract efficient features, thus, a hybrid features selection approach is utilized where we used five widely used techniques of ICA, PCA, RF, CHI2, and AE. The outcome of these FSs approaches (optimal features) are utilized with unsupervised anomaly detection approaches, implying that the HYRIDE does not require labelled data for training. Specifically, the proposed approach analyzes the effectiveness of feature selection approaches in extracting optimal features for LOF, EE, and HBOS based unsupervised intrusion detection models. Three different datasets of TON-IoT, UNSW-NB15, and CICIDS 2017, with different sizes and attributes are experimented with the proposed HYRIDE. The TON-IoT dataset, all the approaches returned almost similar testing accuracy with minor percentage improvements. The highest among them is achieved for AE extracted features and EE trained intrusion detections, which is 3 % better than the accuracy when RF features are used. In the case of UNSW-NB15 dataset, features extracted with CHI2 and trained with HBOS resulted in highest texting accuracy of 0.7527 which is 40.22 % better than the accuracy with ICA extracted features. Moreover, they are 11.44 % and 6 % more than the highest text accuracy attained

with LOF and EE. For the CICIDS 2017 dataset, HBOS with RF resulted in the highest testing accuracy of 0.9013, which is 9.14 % better than the lowest testing accuracy of 0.8307 achieved when ICA features are trained with HBOS.

Table 11 compiles the two important evaluation criteria of average testing accuracy and training time for all the considered hybrid combinations of FS and intrusion detection approaches. The analysis and conclusion are as follows:

- For TON-IoT, LOF trained with ICA features resulted in slightly improved average testing accuracy of 1.22 % as compared to the HBOS trained with CHI2. However, HBOS with CHI2 provided significant improvement of ~98 % in the training time.
- For UNSW-NB15, there is ~40 % improvement for CHI2 combined with HBOS in comparison to HBOS trained with ICA, which in turn has marginally better training time of 1.01 s than the earlier (1.38 s).
- For CICIDS 2017 dataset, HBOS trained with RF features gave minor improvement of 2.89 % in average testing accuracy as compared to, when HBOS is trained with CHI2. In training time, however, CHI2 with HBOS resulted in substantial improvement as compared to the later.
- This trade-off is crucial to clearly analyze the applicability of the results as the selection may depend on the priority of accuracy versus resource efficiency.
- With HBOS model, when trained with CHI2 features demonstrated better performance in terms of average testing accuracy along with faster training times.
- Overall, EE performs optimally when paired with unsupervised FS approaches. In contrast, HBOS is the better choice when supervised FS is employed.

In conclusion, better feature selection is a crucial step for the efficient identification of intrusions, as it helps in reducing the dimensionality of the data along with choosing the most relevant features. Our proposed HYRIDE approach has introduced a robust and hybrid approach to consider both unsupervised and supervised FS methods each with their advantages which are then trained with three unsupervised intrusion detection approaches to efficiently identify intrusions.

5.2. Limitations and future works

The proposed work presents a static feature selection and intrusion detection approach, as has been widely used in literature [49–51], thus its capacity is limited to continually evolving cyber threats. Another limitation of the study is the consideration of fix number of features for all FS approaches, therefore, future work will explore adaptive feature selection tuned specifically for each dataset. Additionally, while the present work emphasizes methodology and comparative analysis, future work will address online intrusion detection scenarios. It will involve handling related issues such as real-time processing and deployment in resource-constrained environments typical of many IoT settings. Furthermore, future work shall explore the optimal deep learning-based approaches in terms of faster execution and accuracy. Additionally, future work shall explore the development of an autonomous cybersecurity AI system [52], so as to extend the practical reach in complex environments. By advancing relevant feature selection and efficient anomaly detection in transparent manner, such work may also set precedent for ethical AI standards in security. Collectively, future work shall focus on designing a robust and autonomous systems, while also ensuring accountability and transparency.

CRediT authorship contribution statement

Shubham Srivastav: Writing – original draft, Visualization, Validation, Investigation, Formal analysis, Data curation. **Amit K. Shukla:** Writing – review & editing, Writing – original draft, Methodology, Conceptualization. **Sandeep Kumar:** Validation, Methodology, Conceptualization. **Pranab K. Muhuri:** Writing – review & editing, Supervision, Methodology, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Table 11
Comparison of average testing accuracy and training time for considered datasets.

Datasets	Best approaches	Avg. testing accuracy	Avg. training time (sec)
TON-IOT	ICA + LOF	0.6535	32.64
	CHI2 + HBOS	0.6456	0.64
UNSW-NB15	CHI2 + HBOS	0.7527	1.38
	ICA + HBOS	0.5368	1.01
CICIDS 2017	RF + HBOS	0.9067	4157.78
	CHI2 + HBOS	0.8812	40.41

Data availability

Data used in the paper is mentioned within the manuscript.

References

[1] M. Janmaijaya, A.K. Shukla, P.K. Muhuri, A. Abraham, Industry 4.0: latent Dirichlet Allocation and clustering based theme identification of bibliography, Eng. Appl. Artif. Intell. 103 (2021) 104280.
[2] A. Ustundag, E. Cevikcan, B.C. Ervural, B. Ervural, Overview of cyber security in the industry 4.0 era. Industry 4.0: Managing the Digital Transformation, 2018, pp. 267–284.
[3] M. Elsisi, M. Altius, S.F. Su, C.L. Su, Robust Kalman filter for position estimation of automated guided vehicles under cyberattacks, IEEE Trans. Instrum. Meas. 72 (2023) 1–12.
[4] M. Elsisi, J.T. Yu, C.C. Lai, C.L. Su, A Drone-assisted deep learning based IoT system for monitoring ship emissions in ports considering adversarial attacks, in: IEEE Transactions on Instrumentation and Measurement, 2024.
[5] I.H. Sarker, Y.B. Abushark, F. Alsolami, A.I. Khan, Intrudtree: a machine learning based cyber security intrusion detection model, Symmetry 12 (5) (2020) 754.
[6] M. Azeem, D. Khan, S. Iftikhar, S. Bawazeer, M. Alzahrani, Analyzing and comparing the effectiveness of malware detection: a study of machine learning approaches, Heliyon 10 (1) (2024).
[7] A. Binbusayyis, T. Vaiyapuri, Comprehensive analysis and recommendation of feature evaluation measures for intrusion detection, Heliyon 6 (7) (2020).
[8] M.A. Mohammed, A. Lakhan, K.H. Abdulkareem, M.K. Abd Ghani, H.A. Marhoon, J. Nedoma, R Martinek, Multi-objectives reinforcement federated learning blockchain enabled Internet of things and Fog-Cloud infrastructure for transport data, Heliyon 9 (11) (2023).
[9] M. Elsisi, M. Amer, C.L. Su, A comprehensive review of machine learning and IoT solutions for demand side energy management, conservation, and resilient operation, Energy (2023) 128256.
[10] M. Verkerken, L. D'hooge, T. Wauters, B. Volckaert, F De Turck, Towards model generalization for intrusion detection: unsupervised machine learning techniques, J. Netw. Syst. Manag. 30 (1) (2022) 1–25.
[11] M.N. Ali, M. Amer, M. Elsisi, Reliable IoT paradigm with ensemble machine learning for faults diagnosis of power transformers considering adversarial attacks, IEEE Trans. Instrum. Meas. (2023).
[12] M. Elsisi, A.L. Rusidi, M.Q. Tran, C.L. Su, M.N. Ali, Robust indoor positioning of automated guided vehicles in internet of things networks with deep convolution neural network considering adversarial attacks, IEEE Trans. Veh. Technol. (2024).
[13] S. Bergies, T.M. Aljohani, S.F. Su, M. Elsisi, An IoT-based deep-learning architecture to secure automated electric vehicles against cyberattacks and data loss, IEEE Trans. Syst. Man Cybern. (2024).
[14] S. Kumar, A.K. Shukla, P.K. Muhuri, Anomaly based novel multi-source unsupervised transfer learning approach for carbon emission centric GDP prediction, Comput. Ind. 126 (2021) 103396.
[15] A.K. Shukla, S. Srivastav, S. Kumar, P.K. Muhuri, UInDeSI4.0: an efficient unsupervised intrusion detection system for network traffic flow in Industry 4.0 ecosystem, Eng. Appl. AI 120 (2023) 1–9.
[16] M.Q. Tran, M. Amer, A.Y. Abdelaziz, H.J. Dai, M.K. Liu, M. Elsisi, Robust fault recognition and correction scheme for induction motors using an effective IoT with deep learning approach, Measurement 207 (2023) 112398.
[17] M. Elsisi, C.L. Su, M.N. Ali, Design of reliable IoT systems with deep learning to support resilient demand side management in smart grids against adversarial attacks, IEEE Trans. Ind. Appl. (2023).
[18] M. Elsisi, C.L. Su, C.H. Lin, T.T. Ku, Enhancing resilient operation of distributed energy resources using reliable machine learning-based IoT connectivity, in: 2024 IEEE/IAS 60th *Industrial and Commercial Power Systems Technical Conference* (I&CPS), IEEE, 2024, pp. 1–6.
[19] O. Alghushairy, R. Alsini, T. Soule, X. Ma, A review of local outlier factor algorithms for outlier detection in big data streams, Big Data Cognit. Comput. 5 (1) (2020) 1.
[20] M. Vishwakarma, N. Kesswani, A new two-phase intrusion detection system with Naïve Bayes machine learning for data classification and elliptic envelop method for anomaly detection, Decis. Anal. J. 7 (2023) 100233.
[21] I. Aguilera-Martos, M. García-Barzana, D. García-Gil, J. Carrasco, D. López, J. Luengo, F. Herrera, Multi-step histogram based outlier scores for unsupervised anomaly detection: ArcelorMittal engineering dataset case of study, Neurocomputing 544 (2023) 126228.
[22] N. Moustafa, A new distributed architecture for evaluating AI-based security systems at the edge: network TON_IoT datasets, Sustain. Cities Soc. 72 (2021) 102994.
[23] H. Xu, L. Zhang, P. Li, F. Zhu, Outlier detection algorithm based on k-nearest neighbors-local outlier factor, J. Algor. Comput. Technol. 16 (2022) 17483026221078111.
[24] M. Omar, Malware anomaly detection using local outlier factor technique. Machine Learning for Cybersecurity, Springer, Cham, 2022, pp. 37–48.
[25] M. Esna-Ashari, F. Khamesian, F. Khanizadeh, Using local outlier factor to detect fraudulent claims in auto insurance, J. Math. Model. Finance 2 (1) (2022) 167–182.
[26] S. Luan, Z. Gu, L.B. Freidovich, L. Jiang, Q. Zhao, Out-of-distribution detection for deep neural networks with isolation forest and local outlier factor, IEEE Access 9 (2021) 132980–132989.
[27] Y. Peng, Y. Yang, Y. Xu, Y. Xue, R. Song, J. Kang, H. Zhao, Electricity theft detection in AMI based on clustering and local outlier factor, IEEE Access 9 (2021) 107250–107259.
[28] Q. Gu, A. Fallah, P. Ashok, D. Chen, E. Van Oort, Real-time multi-event anomaly detection using elliptic envelope and a deep neural network for enhanced MPD robustness, in: IADC/SPE Managed Pressure Drilling & Underbalanced Operations Conference & Exhibition, OnePetro, 2021.
[29] M. Ashrafuzzaman, S. Das, A.A. Jillepalli, Y. Chakhchoukh, F.T. Sheldon, Elliptic envelope-based detection of stealthy false data injection attacks in smart grid control systems, in: 2020 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, 2020, pp. 1131–1137.
[30] D.L. Aguilar, M.A.M. Perez, O. Loyola-Gonzalez, K.K.R. Choo, E. Bucheli-Susarrey, Towards an interpretable autoencoder: a decision tree-based autoencoder and its application in anomaly detection, IEEE Trans. Depend. Secure Comput. (2022).
[31] M.B. Pranto, M.H.A. Ratul, M.M. Rahman, I.J. Diya, Z.B. Zahir, Performance of machine learning techniques in anomaly detection with basic feature selection strategy- network intrusion detection system, J. Adv. Inf. Tech. 13 (1) (2022).
[32] N. Paulauskas, A. Baskys, Application of histogram-based outlier scores to detect computer network anomalies, Electronics 8 (11) (2019) 1251.
[33] Á. Fernández, J. Bella, J.R. Dorronsoro, Supervised outlier detection for classification and regression, Neurocomputing 486 (2022) 77–92.
[34] O.A. Alghanam, W. Almobaideen, M. Saadeh, O. Adwan, An improved PIO feature selection algorithm for IoT network intrusion detection system based on ensemble learning, Expert. Syst. Appl. 213 (2023) 118745.
[35] A.K. Dey, G.P. Gupta, S.P. Sahu, Hybrid meta-heuristic based feature selection mechanism for cyber-attack detection in IoT-enabled networks, Procedia Comput. Sci. 218 (2023) 318–327.
[36] M.A. Talukder, K.F. Hasan, M.M. Islam, M.A. Uddin, A. Akhter, M.A. Yousuf, M.A. Moni, A dependable hybrid machine learning model for network intrusion detection, J. Inf. Secur. Appl. 72 (2023) 103405.
[37] M. Sarhan, S. Layeghy, N. Moustafa, M. Gallagher, M. Portmann, Feature extraction for machine learning-based intrusion detection in IoT networks, Digit. Commun. Netw. (2022).
[38] G. Sah, S. Banerjee, S. Singh, Intrusion detection system over real-time data traffic using machine learning methods with feature selection approaches, Int. J. Inf. Secur. 22 (1) (2023) 1–27.
[39] M.A. Umar, C. Zhanfang, Effects of feature selection and normalization on network intrusion detection, Authorea Preprints (2023).

[40] J.B. Awotunde, S.O. Folorunso, A.L. Imoize, J.O. Odunuga, C.C. Lee, C.T. Li, D.T. Do, An ensemble tree-based model for intrusion detection in industrial internet of things networks, Appl. Sci. 13 (4) (2023) 2479.
[41] S. Amaouche, A. Guezzaz, S. Benkirane, M. Azrour, S.B.A. Khattak, H. Farman, M.M. Nasralla, FSCB-IDS: feature selection and minority class balancing for attacks detection in VANETS, Appl. Sci. 13 (13) (2023) 7488.
[42] H. Zouhri, A. Idri, A. Ratnani, Evaluating the impact of filter-based feature selection in intrusion detection systems, Int. J. Inf. Secur. (2023) 1–27.
[43] M. Sarhan, S. Layeghy, M. Portmann, Towards a standard feature set for network intrusion detection system datasets, Mob. Netw. Appl. (2022) 1–14.
[44] A. Nazir, Z. Memon, T. Sadiq, H. Rahman, I.U. Khan, A novel feature-selection algorithm in IoT networks for intrusion detection, Sensors 23 (19) (2023) 8153.
[45] M. Alhanaya, K.H Ateyeh Al-Shqeerat, Performance analysis of intrusion detection system in the IoT environment using feature selection technique, Intell. Autom. Soft Comput. 36 (3) (2023).
[46] M.M. Breunig, H.P. Kriegel, R.T. Ng, J. Sander, LOF: identifying density-based local outliers, in: Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, 2000, pp. 93–104.
[47] Zhao, Y., Nasrullah, Z., & Li, Z. (2019). Pyod: a python toolbox for scalable outlier detection. arXiv preprint arXiv:1901.01588.
[48] M.M. Alani, A. Miri, Towards an explainable universal feature set for IoT intrusion detection, Sensors 22 (15) (2022) 5690.
[49] B. Kaushik, R. Sharma, K. Dhama, A. Chadha, S. Sharma, Performance evaluation of learning models for intrusion detection system using feature selection, J. Comput. Virol. Hack. Tech. 19 (4) (2023) 529–548.
[50] E.U.H. Qazi, M.H. Faheem, T. Zia, HDLNIDS: hybrid deep-learning-based network intrusion detection system, Appl. Sci. 13 (8) (2023) 4921.
[51] A. Pathak, U. Barman, T.S. Kumar, Machine learning approach to detect android malware using feature-selection based on feature importance score, J. Eng. Res. (Ponta Grossa) (2024).
[52] A.K. Shukla, V. Terziyan, T. Tiihonen, AI as a user of AI: towards responsible autonomy, Heliyon (2024).