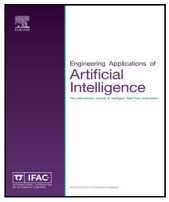
Engineering Applications of Artificial Intelligence 120 (2023) 105848

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



UInDeSI4.0: An efficient Unsupervised Intrusion Detection System for network traffic flow in Industry 4.0 ecosystem

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

[a] Faculty of Information Technology, University of Jyväskylä, Box 35 (Agora), Jyväskylä 40014, Finland
[b] Department of Computer Science, South Asian University, Maidan Garhi, New Delhi 110068, India

ARTICLE INFO

Keywords:
Isolation forest
Industry 4.0
Intrusion detection
ICA
Random forest
Principal component analysis

ABSTRACT

In an Industry 4.0 ecosystem, all the essential components are digitally interconnected, and automation is integrated for higher productivity. However, it invites the risk of increasing cyber-attacks amid the current cyber explosion. The identification and monitoring of these malicious cyber-attacks and intrusions need efficient threat intelligence techniques or intrusion detection systems (IDSs). Reducing the false positive rate in detecting cyber threats is an important step for a safer and reliable environment in any industrial ecosystem. Available approaches for intrusion detection often suffer from high computational costs due to large number of feature instances. Therefore, this paper proposes a novel unsupervised IDS for Industry 4.0 which we term as: Unsupervised Intrusion Detection System for Industry 4.0 (UInDeSI4.0). We have substantiated the proposed UInDeSI4.0 approach through its experimentation on the well-known UNSW-NB15 Industry 4.0 dataset. The proposed UInDeSI4.0 employs feature selection approaches to obtain minimal and optimal features. These features are then used to train isolation forest to detect network traffic threats in an unsupervised manner. Accordingly, the proposed UInDeSI4.0 approach can efficiently differentiate between the normal events and the attacks or intrusions in environments with no label information. Experimental results show that the proposed UInDeSI4.0 provides better accuracy (~63%) and a minimal feature set (nine) compared to traditional IDSs. In contrast to deep learning approaches, UInDeSI4.0 generates faster results with minimum features. In conclusion, we establish the superiority of UInDeSI4.0 approach as an accurate and computationally efficient IDS for Industry 4.0.

1. Introduction

Industry 4.0 (I4.0) conceptualizes factories with processes and organizations to be flexible, customized, efficient, cheaper, safer, and responsible. Powered by digitization and connectivity, the fourth industrial revolution revolves around several technologies that, when used together, are causing a massive paradigm shift. Overall, it is a connected autonomous network that interacts in real-time (Lee et al., 2014; Janmaijaya et al., 2021). The efficiency is attained as the Industry 4.0 systems rely on data-driven approaches for operation and decision-making. Moreover, data transparency and data privacy are two significant issues that are valuable to the industries where people's information is generated and securely accumulated (Onik et al., 2019). The machines in these systems heavily utilize cyber–physical systems (CPSs), internet of things (IoT) devices, and internet connectivity in production processes, which thus open the gateways for increased cyber-attacks. Although the manufacturing in this industry is designed to be highly efficient and secure in the highly volatile network environment, it is still highly vulnerable to intrusion that leads to economic and manufacturing harms with privacy breaches in an organization's assets. These intrusions include distributed denial-of-service, unauthorized access, identity theft, buffer overflow, etc. These attacks could be initiated from any element or module of the overall system, thus, blurring the boundaries between various levels of authentication (Moustafa et al., 2017). Moreover, integrating I4.0 into legacy systems has also enabled attackers to jeopardize their security by exposing crucial information. This is basically due to the integration of different complex technologies. Hence, security in a smart industry is challenging due to more customer/user access in the business and control systems (Yan et al., 2017). The increasing number of modern and different cyber-attacks require efficient intrusion detection systems (IDSs) to secure smart industries.

Therefore, this paper proposes a novel unsupervised intrusion detection approach for Industry 4.0 which we term as: Unsupervised Intrusion Detection System for Industry 4.0 (UInDeSI4.0). The proposed UInDeSI4.0 can efficiently identify intrusions with better accuracy, less

* Corresponding author.
E-mail addresses: amit.k.shukla@jyu.fi (A.K. Shukla), shubham.srv10@gmail.com (S. Srivastav), 2431sandeep@gmail.com (S. Kumar), pranabmuhuri@cs.sau.ac.in (P.K. Muhuri).

https://doi.org/10.1016/j.engappai.2023.105848
Received 7 February 2022; Received in revised form 28 October 2022; Accepted 9 January 2023
Available online xxxx

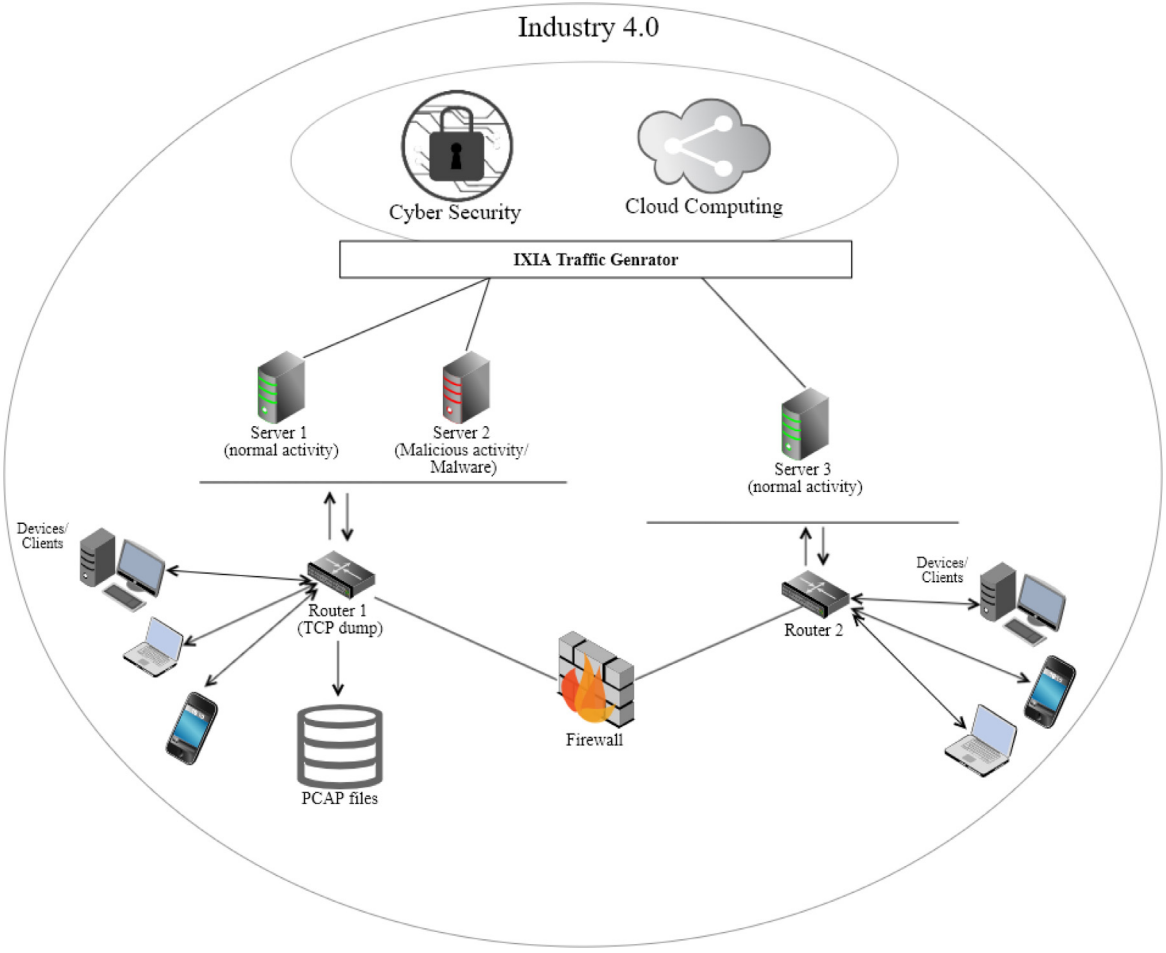


Fig. 1. Traffic flow for generating normal and attack instances in Industry 4.0.

false positive rate, and less computational time than other recently proposed IDSs in the literature. For IDSs, feature selection (FS) is one of the crucial steps (Pérez et al., 2020) since selecting an optimal set of features from high-feature instances is computationally complex in the continuously increasing network traffic data. Accordingly, for FS, our proposed UInDeSI4.0 employs the random forest (RF), which selects the optimal and minimal features based on their importance. It is associated with low overfitting and provides better predictive performance. These RF-selected relevant features reduce the computational cost and also helps in additional analytics. Once the optimal features are extracted, the dataset is subjected to a reliable unsupervised anomaly detection approach which is also a synonym for intrusion detection (ID).

Intrusion detections are majorly of three types: supervised, unsupervised and semi-supervised. In an applicative context, the collected data is in deficiency of any label information. Moreover, most of the ID models need to be trained occasionally as they are unable to detect undefined attacks. UInDeSI4.0 uses the isolation forest (IF) approach to detect anomalies from the selected feature set in an unsupervised manner. IF is an unsupervised tree-based algorithm in which an isolation tree separates the anomalies based on the average path length of the leaf node in a tree by calculating the anomaly score (Liu et al., 2008, 2012). It separates scattered and clustered anomalies more efficiently than the density or distance-based algorithms. Also, IF requires less fine-tuning of the parameters than other unsupervised approaches. Moreover, it has less running time complexity since it isolates anomalies considering a sub-sample of the dataset, which also enables IF to work efficiently with large datasets. It can better deal with the problem of swamping and masking in the dataset. Therefore, to detect intrusions in the network traffic flow of the Industry 4.0 ecosystem, UInDeSI4.0 trains IF by utilizing the RF-extracted optimal features. We have substantiated the proposed UInDeSI4.0 approach through experimentation on the well-known UNSW-NB15 Industry 4.0 dataset. UNSW-NB15 dataset is generated using the IXIA PerfectStorm[1] tool (Moustafa and Slay, 2015), and the complete generation procedure is shown in Fig. 1.

To assess the proficiency and robustness of the proposed UInDeSI4.0, several FS techniques are considered for comparison purposes, including a supervised technique, chi-square (CHI2), and two unsupervised approaches: principal component analysis (PCA) and independent component analysis (ICA). The features extracted from these three FS techniques are then individually trained on the IF and the performance is compared with UInDeSI4.0. Notably, there are also deep learning-based approaches that automatically extract the features and identify the anomalies. However, they are associated with high computational costs and less interpretability. Our proposed UInDeSI4.0, on the other hand, operates on minimal features resulting significant reduction in the computational cost; and importantly, it does not require label information for anomaly detection. Nevertheless, for a fair and thorough comparison, we have further included auto-encoder (AE), deep

[1] The traffic generator tool, PerfectStorm, consists of three virtual servers. Two servers produce normal events and third server produce malicious or attack event in network traffic. The servers are connected to host via routers. A tcpdump tool is installed in the router to capture the PCAP (Packet Capture) in the simulation uptime. These routers relate to the fire wall device that is configured to pass all the traffic either normal or abnormal. The whole process captures normal or attack events in the traffic flow, which shows how network traffic is established between a server and a client. More details on the UNSW-NB-15 Industry 4.0 dataset are provided in the Section 4.

auto-encoder (DAE), and deep variational auto-encoder (DVAE) in our comparative analysis. From the experimental results, it may be seen that the proposed UInDeSI4.0 provides higher accuracy with a minimal feature set than other recently reported approaches. Thus, we establish the superiority of the UInDeSI4.0 approach as a computationally efficient IDS for Industry 4.0.

The major contributions of this paper are as follows:

1. An unsupervised intrusion detection system for Industry 4.0 (UInDeSI4.0) is proposed, which extracts optimal features using RF approach and then identifies anomalies with IF.
2. Experiment results with the well-known UNSW-NB15 Industry 4.0 dataset confirm that the proposed approach achieves efficient performance.
3. A supervised (CHI2) and two unsupervised (PCA and ICA) feature selection approaches are also employed to establish the proficiency and robustness of the UInDeSI4.0.
4. Further, comparisons with state-of-the-art conventional and deep learning-based approaches have been executed to evaluate the suitability of the proposed UInDeSI4.0 approach.

The organization of this paper is as follows: Section 2 discusses the background and related work. Section 3 describes a basic overview and explains the procedure of the proposed UInDeSI4.0 approach. The description of the Industry 4.0 dataset and an explanation of the experiments performed and analysis is discussed in Section 4. Section 5 concludes the paper with a detailed discussion.

2. Related work

IDS has been studied thoroughly in the literature with various supervised and unsupervised approaches. Hassan et al. (2020) have proposed a feature extraction-based semi-supervised deep learning method. It is compatible with multi-level protocols of Industrial-IOT and detect wide range of cyber-attacks effectively. In these cyber-attacks space, Iwendi et al. (2020) introduced *KeySplitWatermark* approach for watermark detection. Wu et al. (2020) proposed the IF-based algorithm for predicting events from low-quality data of synchrophasor measurement. They select features using a hierarchal subspace methodology of two-level scheme. Li et al. (2020) proposed a CNN-GRU based intrusion detection model for a federated CPS. A variational LSTM model for intrusion detection that selects deep learning-based features of UNSW-NB15 dataset was proposed by Zhou et al. in Zhou et al. (2020). Iwendi et al. (2021) proposed a novel approach for IDS in smart healthcare using genetic algorithm and RF.

From the above discussion, we see that several FS techniques have been used in different application areas, such as cyber security, fraud detection, image and voice pattern recognition, geophysics, etc., which are then used to detect anomalous behavior in the dataset. Mostly, IF based approaches with different frameworks as unsupervised methods have been proposed for intrusion detection. Other notable approaches are local outlier factor (LOF), one class support vector machine (OC-SVM), and robust convolution methods.

There are also several neural networks and deep learning-based methods used for IDS (Mittal et al., 2021). However, there is a trade-off between a marginal better accuracy and high computational complexity. For cyber security, we need a more reliable, time-efficient, and better analytical approach that solves vulnerabilities in real time. Our proposed UInDeSI4.0 approach solves intrusion detection problems in the unsupervised big data environments of I4.0 utilizing minimal optimal features extracted through RF. A detailed theoretical comparison of our UInDeSI4.0 with other recent approaches is given in Table 1, which clearly shows that the main motivation behind the proposed UInDeSI4.0 is to have a time-efficient and less complex IDS.

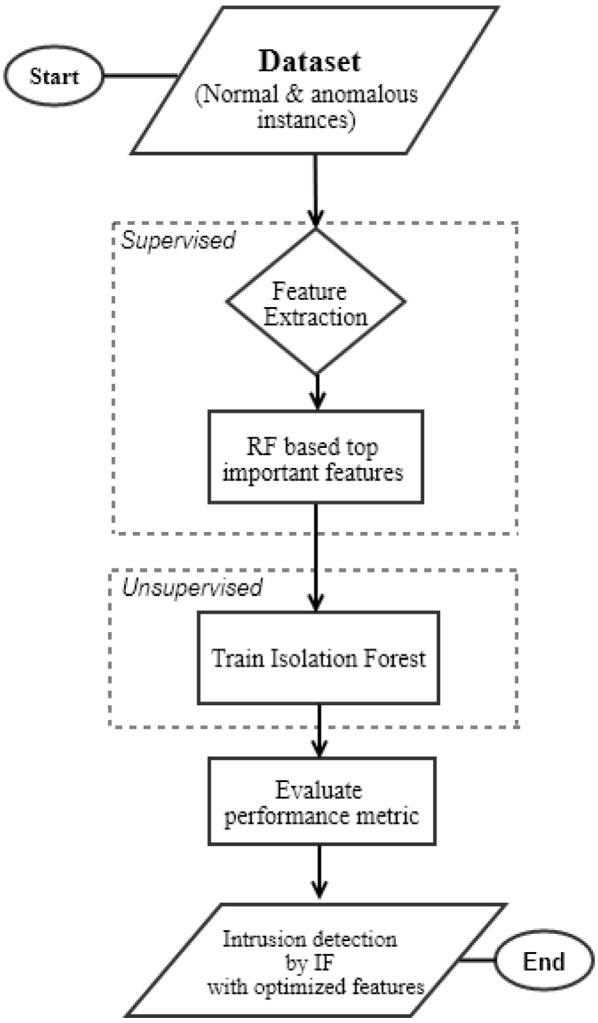


Fig. 2. Flow of the UInDeSI4.0 system.

3. Unsupervised intrusion detection system for Industry 4.0 (UInDeSI4.0)

This section describes the complete procedure for the proposed unsupervised intrusion detection system for industry 4.0 (UInDeSI4.0). It is a generalized feature optimization scheme for intrusion detection. It exploits optimal features from data and utilizes them for training IF to detect anomalies of the selected features. The proposed approach efficiently deals with unbalanced datasets by selecting appropriate data samples. Moreover, it has better decision making for intrusion detection by selecting optimized features of the data. Fig. 2 pictorially depicts the flow of the proposed approach, the step-by-step explanation of which is discussed next.

Step-1: In the first step, optimal features are selected, which is an essential step because irrelevant features in the data make it difficult to detect an intrusion in a network. For making a more reliable and less false positive rate, UInDeSI4.0 utilizes RF-extracted top important features with the highest importance score. The labels in RF are retrieved by first training the model and then selecting the top features according to the highest feature importance score. It returns the relevant optimal features from the data that make a more reliable and generalized framework for intrusion detection. A detailed description of feature sets is given in the experiments and results in Section 4.

Step-2: For each of the optimal features extracted from the above step, we observe its distribution and skewness using the boxplot. This

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

Refs.	FS technique	Methodology	Dataset	Application	Remark
Pérez et al. (2019)	PCA & Auto-Encoder (AE) features	IF, LOF, OC-SVM, & robust-convolution	UNSW-NB15, NSL-KDD, CIC-IDS-2017 and Kyoto dataset.	Intrusion detection	Uses 20 PCA features and 4 layer [230, 120, 60, 20] AE features for UNSW-NB15
Portela et al. (2019)	Sequential FS, main component analysis, features validated in Onik et al. (2019), raw features	SVM, KNN, K-means, IF	UNSW-NB15	Intrusion detection	In unsupervised methods K-means, IF uses raw features of the dataset
Pérez et al. (2020)	High dimensional PCA (Yan et al., 2017), Deep Auto-Encoder (DAE), Deep Variational Auto-Encoder (DVAE)	IF, LOF, Mahalanobis, and HBOS model	UNSW-NB15, NSL-KDD, CIC-IDS-2017 and Kyoto dataset.	Intrusion detection	Uses 100 PCA (HDFS) features, 5-layer [230, 180, 100, 180, 230] DAE and 5-layer [230, 180, 100, 180, 230] DVAE
Yang et al. (2019)	IBFS	IF	Synthetic dataset	Outlier detection	Introduces penalized imbalance score for each feature
Kiran et al. (2020)	Raw features	PCA, AE, IF	i-perf generated traffic	Anomaly detection	PCA, AE are used to detect anomaly cluster, while IF detects anomaly instances.
John and Naaz (2019)	PCA features	LOF, IF	Kaggle credit card transaction dataset	Credit card fraud detection	28 PCA features of the dataset used in IF method.
Liang et al. (2019)	Recursive feature elimination (RFE) technique	RF, OC-SVM, IF, multivariate Gaussian (MG)	Audio recording of telephonic data	Anomaly detection in telephonic data	Selects features with RF& SVM which are used by unsupervised method (IF & MG)
Wang et al. (2020)	Spectral features, Gabor features, EMP features, EMAP features	IF	AVIRIS-I, AVIRIS-II, Cri, PaviaC	Anomaly pixel in image	Spectral features are extracted using spectral feature matrix while other features based on PCA.
Carletti et al. (2019)	DIFFI method	IF	Synthetic dataset, refrigerator industry dataset	Anomaly detection in industry 4.0	An imbalance coefficient λ is introduced for each feature to assign rank to each feature.
Ren et al. (2019)	FS using RF and genetic algorithm (GA)	IF	UNSW-NB15 dataset	Intrusion detection	Feature selected using hybrid technique based on RF and GA
Karczmarek et al. (2020)	Raw features	K-means based IF	Artificial data points, NYC taxi trip, ship, train transportation dataset	Anomaly detection	Each tree is splitted into k branches. Also introduces intuitive result for finding anomaly score.
Ao et al. (2019)	Sequential attribute selection procedure	SCIForest	Channel identification in western bohai sea	Seismic interpretation of channel sand body	Features are selected using sequential attribute selection procedure and SCIForest used to find anomaly clusters.
UInDeSI4.0	**ICA, PCA, CHI2, RF**	**IF**	**UNSW-NB15 dataset**	**Intrusion detection**	**Nine features extracted from RF base FS technique and IF is used to find anomalies**

ALGORITHM-1: IF ANOMALY DETECTION

Output: *Anomalous points*

Input: *Test set x, feature f, no. of iterations T*

1. Let $D(x_i)$ be the depth of the point x_i
2. Let $\underline{D}(x_i)$ be the average of the point x_i
3. Randomly select the feature f
4. While every data point is not in its own leaf do
   - **a.** Randomly select the splitting threshold t from the range $[\min(x_f), \max(x_f)]$
   - **b.** Split the dataset into two subsets based on t
5. Repeat T number of times
6. Compute the anomaly score as follows:
   - **a.** $\text{Score}(x_i)=2^{-\frac{\underline{D}(x_i)}{c(x_i)}}$, Here, $c(x_i)$ is the expected depth
7. Anomaly if Score > threshold.

explanatory data analysis is used to visualize whether there is a linear separation of features between the normal and attack instances.

Step-3: In the third step, the IF is trained with the optimal feature set from RF without giving label information. IF selects a fixed sub-sample S from the data. Data splitting is carried out next by randomly choosing a feature f from optimal features. From a selected feature f, a threshold t is generated, which splits the data point in a tree-based structure. Threshold t is a random value between the minimum and the maximum value of the feature f in each subtree. The data splitting creates a left subtree if the data value is less than the threshold t; else, it creates a right subtree. It continues further recursively until each data point is a leaf itself or some maximum height threshold is attained. IF consists of T decision tree where each decision tree calculates the depth of each data points. An expected depth of each data point is computed from all T decision trees.

Further, an anomaly score for each data point is calculated with the help of the expected depth of each data point as shown in Algorithm 1 (Liu et al., 2012). If the anomaly score is greater than a threshold, the data point is considered anomalous; otherwise, it is a normal point. A detailed numerical experimentation for training IF models is given in Section 4.

Step-4: In step 4, accuracy and ROC curve are computed, which are used as a performance metric for the IF model. The performance evaluation of the selected feature set (from RF) in an IF model is computed in terms of testing accuracy. Here, different contamination values (more detail in Section 4) are used for the IF.

The contamination value is the percentage of expected points set as outliers. ROC (Receiver operating characteristic) curve, which is a plot between the False Positive Rate (FPR) and True Positive Rate (TPR), is used to show the efficacy of the proposed approach.

Step-5: In step 5, we have analyzed the importance of the optimal feature set to be used in an IF model for identifying the anomalies. Algorithm 2 procedurally compiles the proposed UInDeSI4.0. The benefit of the approach lies in a low computational cost due to the optimal FS technique. Moreover, it is an effective and computationally efficient unsupervised technique and works best with any real-world data.

4. Dataset, experiments, and results

This section first details the Industry 4.0 dataset used for experimentation. The discussion of this dataset is of utmost importance due to the scarcity of such realistic datasets. Hence, we first discuss the same, and later, details of the experiments and results are discussed in depth.

4.1. Dataset description

The UNSW-NB15 (Moustafa and Slay, 2015) is an Industry 4.0 dataset that contains more than a million instances. This is one of the rare Industry 4.0 datasets that are publicly available, and readers/practitioners are using it to test their approach in Industry 4.0 domain. As mentioned earlier in Section 1, the UNSW-NB15 dataset is created using IXIA perfect storm tool in the cyber range lab of Australian center for cyber security (ACCS) (Moustafa and Slay, 2015). It consists of modern normal and abnormal network traffic created for industry 4.0 systems. It is a relational dataset where instances consist of attributes of different data types (integer, float, nominal, binary). The label assigned to normal instances is '0', whereas an attack state label is '1'.

The UNSW-NB15 dataset contains a network traffic flow in a cyber–physical system that is generated to understand real-life normal processes and cyber-attack processes in network traffic. This dataset has 47 flow-based and service-based features in network traffic flow. It contains six transactional flow features, six message queue telemetry transport (MQTT) features, 13 DNS features, and 11 http features. The remaining 11 features are basic and time stamp features (Moustafa and Slay, 2015). In all, it contains nine types of attacks: Analysis, Backdoor, DoS, Exploit, Fuzzers, Generic, reconnaissance, Shellcode, and Worms. In the training set, there are 1,75,341 instances, while 82,332 instances in the test set.

Table 2
UNSB-NB15 whole dataset.

Dataset	No of features	No of instances	Normal instances	Attack instances
UNSW-NB15 dataset	42	257 673	93 000	164 673

Table 3
Training accuracy of feature sets using isolation forest.

Isolation forest ($T = 100$ $S = 256$)	BEST training accuracy	Contamination value for training accuracy
PCA features	0.7294	0.25
ICA features	0.6354	0.10
CHI2 features	0.7744	0.25
RF features	0.6926	0.25

4.2. Experiments and results

To evaluate the robustness and the efficacy of the RF-based important features, we have compared it with other widely used FS techniques such as CHI2, PCA, and ICA. UNSW-NB15 dataset is used to analyze the accuracy of different features set in an IF. As mentioned in Table 2, this dataset has a total of 42 features and 257,673 instances, out of which there are 93 000 normal instances and 164,673 attack instances. The features with most of the missing values are removed from the overall feature set resulting in 42 features. Among them, some categorical features are changed into numbers using the label encoder method.

In the first step, features are selected from RF, CHI2, PCA, and ICA techniques. Notably, CHI2 features are extracted by giving label information during feature extraction. Fig. 3(a)–(d) respectively shows the box plot of the distribution of normal instances and attack instances in ICA features, PCA features, CHI2 features, and RF-based important features of the UNSW-NB15 dataset.

The PCA features and CHI2 features are extracted using the highest covariance principle and chi-square test method, respectively. In contrast, RF-based important features are extracted according to the highest importance value. In each box plot, the red line is the distribution of normal instances, whereas the blue line is the distribution of attack events. Each feature value is normalized in the range of 0 to 1.

In the training phase, IF is applied to each of the feature sets selected by the FS techniques without giving label information in an unsupervised manner. To assess the accuracy of the approach, different contamination values are used to analyze the effectiveness of each feature set in the dataset. A contamination value is a parameter in IF, which is set as the percentage of anomalies in the dataset. For experimenting with IF, the parameters settings are: contamination values in the range [0.1:0.05:0.4], sample size (S) is 256, and the number of trees (T) is 100, which are the default values as suggested by the original algorithm (Liu et al., 2012). Fig. 4 shows the training accuracy of each feature set in the considered contamination values.

It can be deduced from this figure that ICA achieved best training accuracy at contamination value of 0.10, whereas PCA, CHI2 and RF achieved it at 0.25. Among all the FS techniques, CHI2 attained the higher training accuracy for every contamination value, while ICA features have the lowest training accuracy. CHI2 achieved its highest training accuracy at the contamination value of 0.25, and ICA achieved it at 0.10. PCA features has higher training accuracy than RF which achieved its higher training accuracy at the contamination value of 0.25. Table 3 compiles training accuracies of each of the FS techniques with their best selected contamination values.

Algorithm 2: UInDeSI4.0

Input: *Dataset (D): UNSW-NB-15* $\{D_{TRAIN} \cup D_{TEST}\}$
Parameters: Sample (S), Number of Trees (T), Contamination Value (C)
Feature set: F_RF

Output: *Intrusion detection, accuracy*

Training phase:

1. *F_RF,* N_f = *extracted features using RF, Number of extracted features*
2. $F_i = (f_j)_{j=1}^{j=N_f}$,
3. for each F_i :
4. for T = 1; T<=100; T++
   $Select\ S\ samples\ from\ D_{TRAIN}$
   $select\ a\ feature\ f_k\ where\ k = rand(1, N_f)$
   $recursion(S, t)$:
   $select\ thresold\ t = rand([min(value(f_k)), max(value(f_k))])$
5. for each instance of *S*:
   $if\ (value(S) < t)$:
   $S = S_{LST} \in left\ sub\ tree$
   $recursion(S_{LST}, t), until\ each\ instance\ is\ in\ the\ leaf\ node$
   $calculate\ anomaly\ score\ of\ each\ leaf\ node\ using\ Algo.1.$
   $else$:
   $S = S_{RST} \in right\ sub\ tree$
   $recursion(S_{RST}, t), until\ each\ instance\ is\ in\ the\ leaf\ node$
   $calculate\ anomaly\ score\ of\ each\ leaf\ node\ using\ Algo.1$
6. for C =0.1 to [0.1:0.05:0.4]:
   find optimal threshold t_{opt}, for fitting D_{TRAIN} with best training accuracy
   If ($anomaly\ score > t_{opt}$):
   leaf node is an anomaly
   else:
   leaf node is a normal
7. Output:
   Trained model M with parameter (S, T, C) using F_{RF} features

Testing phase:

8. for the trained model M :
   Evaluate testing acc(D_{TEST}) using trained model *M*

The testing accuracies are computed on the highest achieved contamination values for each of the above FS techniques. Fig. 5 shows the average testing accuracy bar-plot of the proposed UInDeSI4.0 (with RF feature set) compared with the average testing accuracy on IF when trained with feature sets of CHI2, ICA, and PCA. To further show the effectiveness of the binary classifier among all the above four approaches, Fig. 6 shows the ROC curve. In this curve, the maximum AUC (Area Under the Curve) represents better class separation in the dataset.

The RF features have better area under the curve than the CHI2, PCA, ICA features. Besides RF, PCA, ICA, and CHI2, Fig. 5 also shows the average testing accuracies given by five other recent published approaches viz. PCA with 20 features (PCA20), AE, DAE, DVAE, and Raw Features (RWF) (Pérez et al., 2019; Portela et al., 2019; Pérez et al., 2020). Out of all, UInDeSI4.0 provides better accuracy with faster execution time as compared to other approaches. DAE and DVAE attained marginally higher accuracies than UInDeSI4.0 but at higher computational times. Table 4 shows the optimal number of features, average testing accuracy and average training accuracy (with standard deviation (SD)), maximum testing accuracy, timing complexity, and average execution time of the proposed UInDeSI4.0, along with eight other compared approaches. As can be seen in terms of all the performance factors, i.e., training accuracy (0.6897) and testing accuracy (0.6261), time complexity ($O(fn \log n) + O(n)$), and execution time (14.38 *sec*), UInDeSI4.0 performs efficiently considering its being the unsupervised approach.

CHI2 shows slightly better testing accuracy and average time execution, however, it uses the label information during FS, hence, is categorized as a supervised approach. ICA and PCA have somewhat faster average time as compared to UInDeSI4.0 but at the cost of lower testing accuracy. Other approaches used by Pérez et al. (2019) with 20 feature set using PCA and AE achieved inferior accuracy of 0.5944 and 0.6374, respectively. AE's average execution time is extremely higher than the UInDeSI4.0, i.e., 327.24 s. Recently, raw features (RAWF) of the dataset were experimented using IF (Portela et al., 2019). It also gives less average testing accuracy (0.5678) and high execution

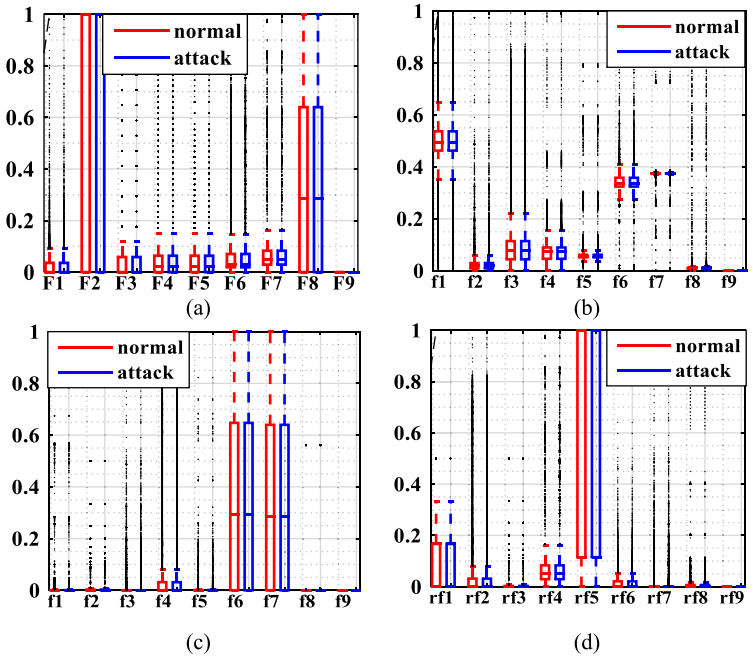


Fig. 3. Box plot of normal and attack instances: (a) of ICA features, (b) PCA features, (c) CHI2 features, and (d) RF features.

Table 4
Comparing proposed approach with other state of the art methods in UNSW-NB15 dataset using IF.

Method	FS technique	No of features/Resulting dimensions	Average testing accuracy (100 iterations)	Average training accuracy (100 iterations)	Maximum testing accuracy	Time complexity	Average time (s) (FS + IF training)
ICA + IF	ICA features (c = 0.10)	9	0.5029 ± 0.0070	0.6327 ± 0.0057	0.5245	$O(2qf(f+1)n) + O(n)$	12.26
PCA + IF	PCA features (c = 0.25)	9	0.6129 ± 0.0091	0.7087 ± 0.0134	0.6391	$O(n^2 f)$ +O(n)	10.98
CHI2 + IF	CHI2 features (c = 0.25)	9	0.6438 ± 0.0111	0.7738 ± 0.0102	0.6735	$O(no\ of\ classes\ *\ f) + 0(n)$	10.14
UInDeSI4.0	RF-features (c = 0.25)	9	0.6261 ± 0.0061	0.6897 ± 0.0079	**0.6410**	$O(fn\log n)$ + O(n)	**14.38**
Pérez et al. (2019)	PCA features (c = 0.10)	20	0.5944 ± 0.0175	0.6533 ± 0.0182	0.6310	$O(n^2 f)$ + O(n)	14.39
	AE features (c = 0.42)	20	0.6374 ± 0.0201	0.6681 ± 0.0139	0.6737	$O(nq*(i*j+j*k+k*l)) + O(n)$	327.24
Pérez et al. (2020)	Deep AE features (c = 0.15)	100	0.6132 ± 0.0117	0.7157 ± 0.0123	0.6437	$O(nq*(i*j+j*k+k*l+l*m))$ + O(n)	358.99
	Deep VAE features (c = 0.05)	100	0.5472 ± 0.0043	0.6596 ± 0.0026	0.5597	$O(nq*(i*j+j*k+k*l+l*m))$+ O(n)	349.90
Portela et al. (2019)	Raw features	42	0.5678 ± 0.0091	0.6729 ± 0.0109	0.5968	O(n)	17.27

*AE = Auto-Encoder, *VAE = Variational Auto-Encoder, *c = contamination value.

time (17.27 s) in comparison to UInDeSI4.0. Neural network-based approaches such as DAE and DVAE took 358.99 and 349.90 s and returned testing accuracy of 0.6132 and 0.5472, respectively. The marginally better maximum testing accuracy of AE and DAE comes at the cost of significant execution time. This trade-off of better accuracy (marginal) versus high computational cost is solely dependent on the context of the application and other factors. Notably, the testing accuracy SD of the proposed UInDeSI4.0 (0.0061) is significantly better than any other approaches, making it more reliable. Only DAE has a better SD of 0.0043, however, it performs poorly on all the other performance indicators.

Overall, our approach produces significantly better accuracies in efficient execution times with manageable computational costs and only nine optimal features. The extracted features from RF according to its feature importance value are shown in Table 5 for the UNSW-NB15 Industry 4.0 dataset.

An interesting aspect of this work is the dataset (in particular) used. The dataset used in this paper has around 67.80% of attack events as compared to the normal events (32.20%). This fact can be verified from Table 6 which classifies the number of normal and attack instances of the UNSW-NB15 dataset. However, related works available in the literature mostly considered only those datasets where the anomalies or irregularities (attack events) in the dataset are less in numbers than the regularities (normal events).

Moreover, the anomaly detection approach employed by UInDeSI4.0, i.e., IF also works on the principle that anomalies (outliers) are less in number, and they may be easily isolated from normal events (inliers), which are in the majority. After the computation of anomaly

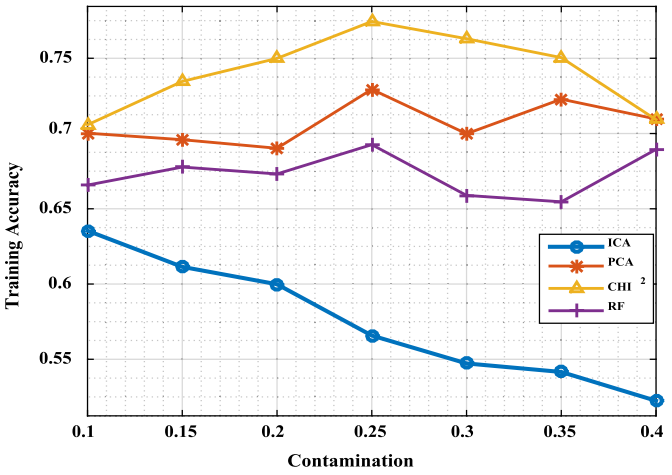


Fig. 4. Training accuracy.

Table 5
RF-based important features.

Features (Abbreviation)	Feature importance index
sttl (rf1)	0.125962
ct_state_ttl (rf2)	0.103160
dload (rf3)	0.077700
rate (rf4)	0.063724
dmean (rf5)	0.061195
dttl (rf6)	0.043587
ackdat (rf7)	0.040899
Sload (rf8)	0.039744
Sbytes (rf9)	0.033084

Table 6
UNSW-NB15 dataset used in isolation forest.

UNSW-NB15 dataset	Normal instances	Attack instances
Training dataset	55 456	119 876
Testing dataset	27 435	54 987

Table 7
Labels allocation for the UNSW-NB15 dataset with IF.

	Original data (label)	IF prediction	Labels change
Normal (32.2%)	0	−1 (1)	0
Attack (67.8%)	1	1 (0)	1

scores, IF will predict '1' for inliers and '−1' for outliers. In general, inliers are considered normal events, while outliers are considered as attack events by IF. However, for UNSW-NB15 dataset, we have considered attack events as inliers and normal events as outliers since they are in the minority. This situation can be visualized in Table 7. Hence, from the aspects of the studied data set also, the current work is a novel contribution.

Based on the above discussion, there are a few but crucial observations that we would like to point out to the readers before they undergo any intrusion detection approach:

1. A careful analysis of the dataset is a prerequisite before the intrusion detection step. In exceptional cases where anomalies are comparatively high compared to normal events, label switching is required in the accuracy computation step after the anomaly detection process is completed.
2. This verification of labels is required only when we already have label information to compute the accuracy. In an unsupervised setting, IF can be used the way it is intended. It is to be kept in mind that, for IF, anomalies are those points which belongs to minority class irrespective of whether they are normal instances or attack instances. Else, IF will be wrongly tagged as a poor performing approach, in such rare scenarios where attack instances dominate.

5. Discussion and conclusion

Due to the advent of industry 4.0 and its rapid, pervasive all-around expansion, the threat of cyber-attacks has touched its zenith. With the continuous emission of data from smart devices in the smart industry, efficient threat intelligence techniques or intrusion detection systems (IDSs) for identifying and monitoring malicious cyber-attacks and intrusions is the need of the hour. This paper has proposed a novel unsupervised IDS for Industry 4.0 which we have termed as: unsupervised intrusion detection system (UInDeSI4.0). In our proposed UInDeSI4.0 approach, after extracting the optimal feature set technique, anomalies are identified using IF in an unsupervised manner.

We have experimented the proposed UInDeSI4.0 approach on the well-known UNSW-NB15 Industry 4.0 dataset. The reliability and efficacy of the proposed UInDeSI4.0 are justified by comparing it with other recently reported approaches. The feature sets selected by various FS techniques are trained with the IF model to analyze the importance of diverse feature sets using the unbalanced and unsupervised UNSW-NB15 dataset. It also shows the impact of all these feature sets in threat detection by IF in the Industry 4.0 domain. Experimental analysis suggests that the proposed UInDeSI4.0, which extracts the optimal feature set using RF, performs better in most cases compared to its other counterparts. When compared with the deep learning-based methods, our proposed UInDeSI4.0 approach is computationally faster and provides comparable accuracy. The proposed UInDeSI4.0 gives a robust solution to real-world smart monitoring systems by enabling intrusion detection in an unsupervised manner. Moreover, our study infers that the RF features perform efficiently with IF in threat detection as compared to other features. Our proposed UInDeSI4.0 may be applied in anomaly detection tasks such as detecting incorrect values in the database, fraud detection, transportation system, and many other application areas as a generalized approach. In the future, new feature extraction techniques with optimized isolation forest models shall be tested. A unified tree-based structure shall be explored for FS and anomaly detection for efficient and much faster execution.

CRediT authorship contribution statement

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

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.

Data availability

Details of the data is already mentioned in the manuscript.

Acknowledgments

We are thankful to the anonymous reviewers and editors for their valuable comments, which helped us improve the paper significantly.

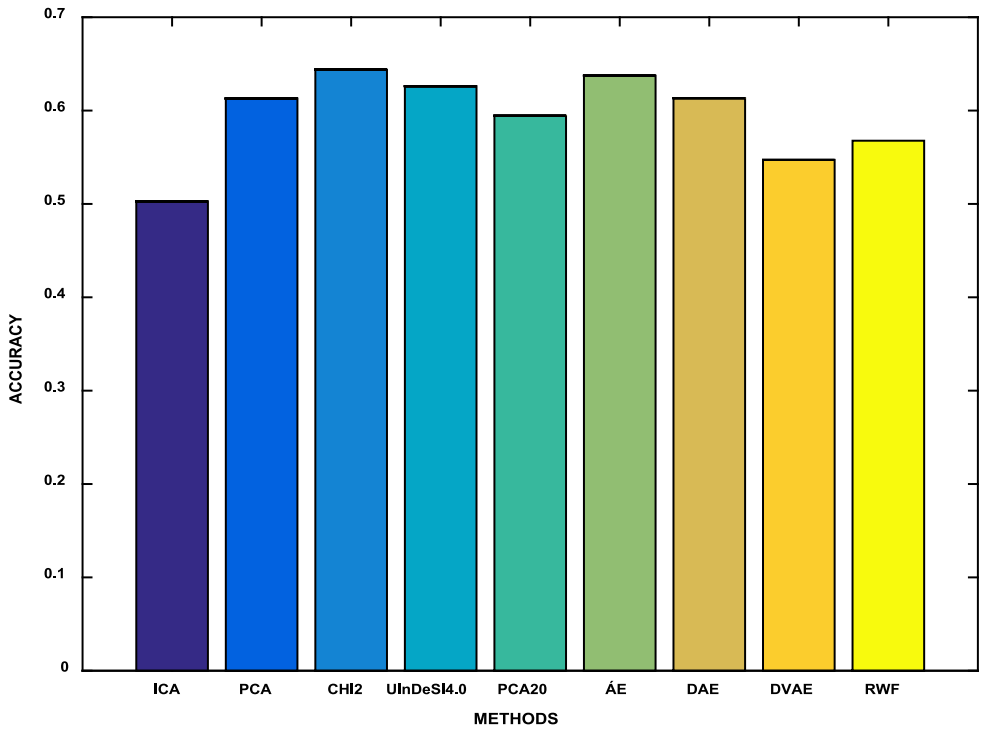


Fig. 5. Bar plot of the average testing accuracy.

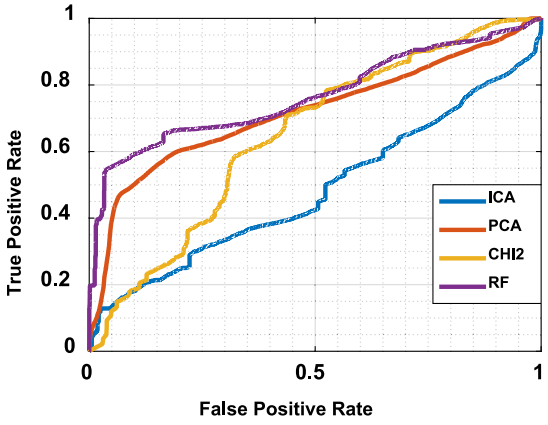


Fig. 6. ROC curve of feature sets using IF.

References

Ao, Y., Li, H., Zhu, L., Yang, Z., 2019. A SCiForest based semi-supervised learning method for the seismic interpretation of channel sand-body. J. Appl. Geophys. 167, 51–62.

Carletti, M., Masiero, C., Beghi, A., Susto, G.A., 2019. Explainable machine learning in industry 4.0: evaluating feature importance in anomaly detection to enable root cause analysis. In: 2019 IEEE International Conference on Systems, Man and Cybernetics. SMC, IEEE, pp. 21–26.

Hassan, M., Huda, S., Sharmeen, S., Abawajy, J., Fortino, G., 2020. An adaptive trust boundary protection for IIoT networks using deep-learning feature extraction based semi-supervised model. IEEE Trans. Ind. Inform..

Iwendi, C., Anajemba, J.H., Biamba, C., Ngabo, D., 2021. Security of things intrusion detection system for smart healthcare. Electronics 10 (12), 1375.

Iwendi, C., Jalil, Z., Javed, A.R., Reddy, T., Kaluri, R., Srivastava, G., Jo, O., 2020. Keysplitwatermark: Zero watermarking algorithm for software protection against cyber-attacks. IEEE Access 8, 72650–72660.

Janmaijaya, M., Shukla, A.K., Muhuri, P.K., Abraham, A., 2021. Industry 4.0: Latent Dirichlet Allocation and clustering based theme identification of bibliography. Eng. Appl. Artif. Intell. 103, 104280.

John, H., Naaz, S., 2019. Credit card fraud detection using local outlier factor and isolation forest. Int. J. Comput. Sci. Eng. 7, 1060–1064.

Karczmarek, P., Kiersztyn, A., Pedrycz, W., Al, E., 2020. K-means-based isolation forest. Knowl.-Based Syst. 105659.

Kiran, M., Wang, C., Papadimitriou, G., Mandal, A., Deelman, E., 2020. Detecting anomalouspackets in network transfers: investigations using PCA, autoencoder and isolation forest in TCP. Mach. Learn. 1–17.

Lee, J., Kao, H.A., Yang, S., 2014. Service innovation and smart analytics for industry 4.0 and big data environment. Procedia Cirp 16 (1), 3–8.

Li, B., Wu, Y., Song, J., Lu, R., Li, T., Zhao, L., 2020. DeepFed: Federated deep learning for intrusion detection in industrial cyber–physical systems. IEEE Trans. Ind. Inform..

Liang, W., Li, K.C., Long, J., Kui, X., Zomaya, A.Y., 2019. An industrial network intrusion detection algorithm based on multifeature data clustering optimization model. IEEE Trans. Ind. Inform. 16 (3), 2063–2071.

Liu, F.T., Ting, K.M., Zhou, Z.H., 2008. Isolation forest. In: 2008 Eighth IEEE International Conference on Data Mining. IEEE, pp. 413–422.

Liu, F.T., Ting, K.M., Zhou, Z.H., 2012. Isolation-based anomaly detection. ACM Trans. Knowl. Discov. Data (TKDD) 6 (1), 1–39.

Mittal, M., Iwendi, C., Khan, S., Rehman Javed, A., 2021. Analysis of security and energy efficiency for shortest route discovery in low-energy adaptive clustering hierarchy protocol using Levenberg–Marquardt neural network and gated recurrent unit for intrusion detection system. Trans. Emerg. Telecommun. Technol. 32 (6), e3997.

Moustafa, N., Creech, G., Slay, J., 2017. Big data analytics for intrusion detection system: Statistical decision-making using finite dirichlet mixture models. In: Data Analytics and Decision Support for Cybersecurity. Springer, Cham, pp. 127–156.

Moustafa, N., Slay, J., 2015. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: 2015 Military Communications and Information Systems Conference (MilCIS). IEEE, pp. 1–6.

Onik, M.M.H., Kim, Chul-Soo, Yang, Jinhong, 2019. Personal data privacy challenges of the fourth industrial revolution. In: 2019 21st International Conference on Advanced Communication Technology. ICACT, IEEE, pp. 635–638.

Pérez, D., Alonso, S., Morán, A., Prada, M.A., Fuertes, J.J., Domínguez, M., 2019. Comparison of network intrusion detection performance using feature representation. In: International Conference on Engineering Applications of Neural Networks. Springer, Cham, pp. 463–475.

Pérez, D., Alonso, S., Morán, A., Prada, M.A., Fuertes, J.J., Domínguez, M., 2020. Evaluation of feature learning for anomaly detection in network traffic. Evol. Syst. 1–12.

Portela, F.G., Mendoza, F.A., Benavides, L.C., 2019. Evaluation of the performance of supervised and unsupervised machine learning techniques for intrusion detection. In: 2019 IEEE International Conference on Applied Science and Advanced Technology. iCASAT, IEEE, pp. 1–8.

Ren, J., Guo, J., Qian, W., Yuan, H., Hao, X., Jingjing, H., 2019. Building an effective intrusion detection system by using hybrid data optimization based on machine learning algorithms. Secur. Commun. Netw. (2019).

Wang, R., Nie, F., Wang, Z., He, F., Li, X., 2020. Multiple features and isolation forest-based fast anomaly detector for hyperspectral imagery. IEEE Trans. Geosci. Remote Sens..

Wu, T., Zhang, Y.J.A., Tang, X., 2020. Online detection of events with low-quality synchrophasor measurements based on *i* forest. IEEE Trans. Ind. Inform. 17 (1), 168–178.

Yan, J., Meng, Y., Lu, L., Li, L., 2017. Industrial big data in an industry 4.0 environment: Challenges, schemes, and applications for predictive maintenance. IEEE Access 5, 23484–23491.

Yang, Q., Singh, J., Lee, J., 2019. Isolation-based feature selection for unsupervised outlier detection. In: Annual Conference of the PHM Society, Vol. 11. No. 1.

Zhou, X., Hu, Y., Liang, W., Ma, J., Jin, Q., 2020. Variational LSTM enhanced anomaly detection for industrial big data. IEEE Trans. Ind. Inform. 17 (5), 3469–3477.