Bulletin of Electrical Engineering and Informatics
Vol. 15, No. 6, December 2026, pp. 5500~5513
ISSN: 2302-9285, DOI: 10.11591/eei.v15i6.14121

Dual-stage autoencoder and density clustering for unsupervised cloud intrusion detection

Sant Kumar Maurya[1], Karuppiah Tamilarasi[2], Sunil Kumar[3], Gaurav Vinchurkar[4], Praveen Kumar Gupta[4], Anubhav Kumar[5]
[1]Department of Computer Science and Engineering, Apex Institute of Technology, Chandigarh University, Mohali, India
[2]Department of Information Technology, Panimalar Engineering College, Chennai, India
[3]Department of Information Technology, Ajay Kumar Garg Engineering College, Ghaziabad, India
[4]Department of AIML, School of Artificial Intelligence, Galgotias University, Greater Noida, India
[5]Department of Computer Science and Engineering, Faculty of Engineering and Technology, HRIT University, Ghaziabad, India

Article Info

Article history:

Received Jul 31, 2026
Revised Sep 7, 2026
Accepted Sep 12, 2026

Keywords:

Anomaly detection
Autoencoder
Cloud intrusion detection
Density-based spatial clustering of applications with noise
Latent representation
Unsupervised learning

ABSTRACT

Cloud intrusion detection remains challenging because modern network traffic is high-dimensional, highly imbalanced, and continuously evolving. Although autoencoders and density-based clustering have been applied independently for intrusion detection, directly clustering the learned latent space often reduces the separation between benign and malicious traffic, particularly for minority attacks. This paper presents a dual-stage unsupervised intrusion detection approach that combines latent representation learning with anomaly-filtered density clustering. A dual-function autoencoder is first trained using benign traffic to learn compact latent representations while identifying anomalous flows through reconstruction error. Instead of clustering the complete latent space, only the latent representations of detected anomalies are forwarded to density-based spatial clustering of applications with noise (DBSCAN) for attack grouping. This anomaly-filtered clustering strategy reduces the influence of dominant benign traffic and improves the discrimination of attack patterns in the latent space. The proposed method was evaluated on the CSE-CIC-IDS2018 benchmark dataset containing diverse cloud network attacks. Experimental results achieved 99.46% anomaly detection accuracy and 98.81% multiclass attack classification accuracy using a 21-dimensional latent representation. Improved clustering quality was further confirmed by a Silhouette score of 0.9859 and a Davies-Bouldin index (DBI) of 0.0093. The proposed pipeline provides an effective unsupervised solution for cloud intrusion detection without requiring attack labels during model training.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Anubhav Kumar
Department of Computer Science and Engineering, Faculty of Engineering and Technology
HRIT University
Ghaziabad, Uttar Pradesh, India
Email: dr.anubhavkumar@gmail.com

1. INTRODUCTION

Cloud computing has become the primary infrastructure for hosting enterprise applications, internet of things (IoT) services, and data-intensive platforms because of its scalability, resource elasticity, and cost efficiency [1], [2]. The increasing reliance on cloud services, however, has also expanded the attack surface exposed to cyber threats [3], [4]. Modern cloud environments generate heterogeneous network traffic with continuously evolving communication patterns, making them vulnerable to attacks such as distributed denial-

of-service (DDoS), brute-force, botnet, infiltration, and web-based exploits [5], [6]. Consequently, intrusion detection systems (IDSs) have become an indispensable component of cloud security [7]–[9]. Unlike conventional enterprise networks, cloud environments produce high-dimensional and highly imbalanced traffic, where benign flows substantially outnumber malicious ones [10], [11]. These characteristics reduce the effectiveness of traditional signature-based detection techniques and motivate the adoption of intelligent data-driven intrusion detection models.

Recent studies from 2024–2026 demonstrate continued progress toward intelligent and data-driven intrusion detection for cloud and networked environments. Aljuaid and Alshamrani [12] employed deep convolutional representation learning for cloud intrusion detection, while Long *et al.* [13] developed a lightweight convolutional neural network (CNN) optimized using particle swarm optimization to improve detection efficiency. Sajid *et al.* [14] further investigated hybrid deep-learning architectures combining convolutional, recurrent, and boosting mechanisms for multiclass intrusion detection. These approaches demonstrate the effectiveness of deep representation learning but remain primarily dependent on labelled attack data for model development.

Recent efforts have increasingly focused on reducing this dependence through semi-supervised, unsupervised, and distributed learning. Megouache *et al.* [15] investigated machine-learning-based anomaly detection for cloud data classification, while Li *et al.* [16] developed a semi-supervised intrusion-detection framework combining representation learning with conventional classifiers. Alharbi *et al.* [17] addressed heterogeneous distributed environments through federated learning, highlighting the importance of decentralized and privacy-aware cybersecurity mechanisms. More recent studies have also explored graph-based and structural representations for intrusion detection. Dugyala *et al.* [18] combined graph neural networks with clustering for cloud intrusion detection, whereas Villegas-Ch *et al.* [19] employed dynamic graph modelling and graph-based neural networks for IoT intrusion detection. Density-based learning has likewise been investigated for security analytics [20], [21], demonstrating its potential for identifying compact behavioural structures without requiring predefined class boundaries.

Despite these advances, three limitations remain evident in the recent literature. First, many deep-learning-based IDS approaches rely on labelled attack samples, limiting their ability to adapt to previously unseen or evolving attacks. Second, although autoencoders and other representation-learning methods can provide compact latent spaces, clustering is commonly performed on the complete feature or latent space, allowing the dominant benign population to influence density estimation. Third, recent intelligent and structural approaches often introduce additional architectural or computational complexity when modelling relationships among traffic flows. These limitations motivate a lightweight sequential design in which benign-only representation learning and reconstruction-error-based anomaly detection are performed before density-based attack categorization. Accordingly, the proposed framework forwards only anomaly-specific latent representations to density-based spatial clustering of applications with noise (DBSCAN), reducing benign-density interference while retaining an unsupervised learning paradigm.

Unlike recent hybrid Autoencoder–DBSCAN approaches [5], [10], [11], [22], which apply clustering to learned or reduced feature representations, the proposed framework introduces an anomaly-filtered latent-space clustering strategy. Specifically, reconstruction-error-based anomaly detection is performed before clustering, and only the latent representations of detected anomalous traffic are forwarded to DBSCAN. This design reduces the influence of the dominant benign population during density estimation and enables clustering to focus on the local structure of anomalous traffic, thereby facilitating the separation of minority attack behaviours. The DBSCAN parameters are further evaluated across different neighbourhood and density settings to identify a stable configuration for the anomaly-filtered latent space. This sequential interaction between anomaly filtering and density-based clustering constitutes the central methodological distinction of the proposed framework from prior Autoencoder–DBSCAN approaches.

Building on this methodological distinction, the proposed framework integrates benign-only representation learning, reconstruction-error-based anomaly detection, and anomaly-focused density clustering into a unified unsupervised pipeline. In addition to the anomaly-filtering mechanism, the study investigates a compact 21-dimensional latent representation and evaluates its effectiveness for multiclass attack categorization. The major contributions of this work are summarized as follows:

a. Anomaly-filtered latent-space clustering: an unsupervised framework is proposed in which a benign-trained autoencoder identifies anomalous traffic through reconstruction error, after which only the corresponding latent representations are clustered using DBSCAN. This sequential design reduces the influence of dominant benign traffic during density estimation and focuses clustering on anomalous traffic structures.
b. Compact latent representation: a 21-dimensional latent representation is investigated against a 29-dimensional configuration, demonstrating stronger attack categorization performance while reducing the original 80-dimensional feature space by 73.75%.
c. Unsupervised attack categorization: the framework performs representation learning, anomaly detection, and density-based clustering without attack labels during model development; ground-truth labels are used only for post-hoc cluster evaluation and performance measurement.

The proposed framework is evaluated on the CSE-CIC-IDS2018 benchmark dataset under highly imbalanced cloud traffic, with experiments examining anomaly detection, attack categorization, latent-dimensionality selection, clustering behaviour, and comparative performance against representative baseline methods.

2. METHOD

2.1. Overall framework

The proposed framework follows a dual-stage unsupervised pipeline, as illustrated in Figure 1. Given a preprocessed network-flow vector x_i, the encoder $f_\theta(\cdot)$ generates a latent representation z_i, while the decoder $g_{\phi(\cdot)}$ reconstructs the input as $\hat{x_i} = g_\phi(z_i)$. Thus,

$$z_i = f_\theta(x_i),\ \hat{x_i} = g_\phi(z_i)$$

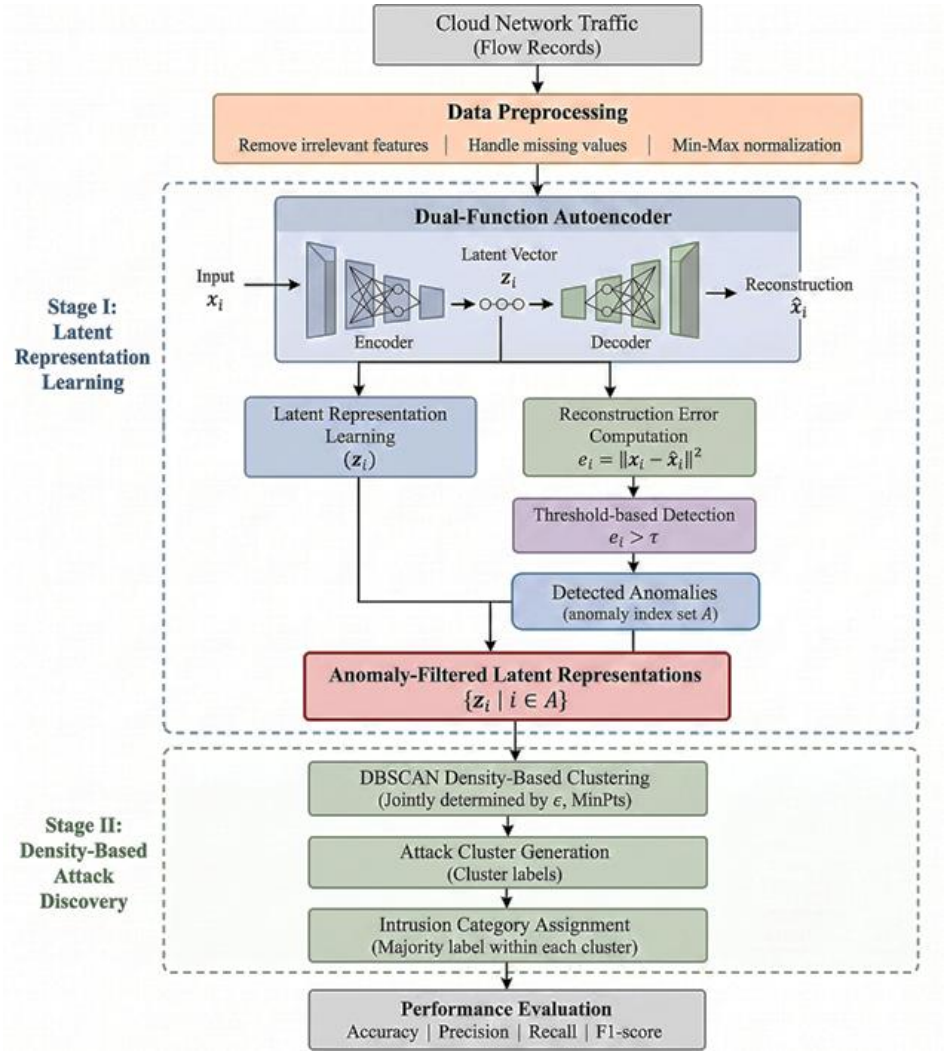


Figure 1. Overview of the dual-stage cloud intrusion detection architecture

The autoencoder is trained exclusively on benign traffic by minimizing the reconstruction loss. During inference, the reconstruction error $e_i = |x_i - \hat{x_i}|^2$ is compared with the threshold τ, and an anomaly indicator is obtained as $a_i = I(e_i > \tau)$. The interaction between the two stages is defined by the anomaly-filtered latent set,

$$Z_A = \{z_i \mid a_i = 1\},$$

which contains only the latent representations of detected anomalous flows. DBSCAN subsequently operates on Z_A rather than on the complete latent space:

$$C = DBSCAN(Z_A; \epsilon, \text{MinPts}),$$

where C denotes the resulting attack clusters. Consequently, anomaly detection determines which latent representations are forwarded to the density-clustering stage, while DBSCAN groups these anomaly-specific representations according to their local density structure. This sequential interaction reduces the influence of dominant benign traffic and enables unsupervised attack categorization.

The raw CSE-CIC-IDS2018 traffic records were consolidated from the available CSV files into a unified dataset. Duplicate and corrupted records were removed during data cleaning. Categorical variables, including protocol and service-related attributes, were converted using one-hot encoding to obtain numerical representations without introducing ordinal relationships. No manual feature-selection or feature-dropping procedure was applied; the retained feature representation was subsequently compressed through the autoencoder. Missing values were handled during preprocessing, and the numerical features were scaled using Min-Max normalization before being supplied to the benign-trained autoencoder. The normalized data are then supplied to the benign-trained autoencoder. During inference, flows exceeding the reconstruction threshold are identified as anomalous, and their latent representations are forwarded to DBSCAN for attack grouping. Cluster assignments are finally mapped to intrusion categories only for post-hoc evaluation.

2.2. Stage I: latent representation learning

The objective of the first stage is to learn a compact representation of normal cloud traffic while preserving its intrinsic behavioural characteristics. Before model training, each numerical feature is normalized using Min-Max scaling to eliminate differences in feature magnitude and accelerate convergence during optimization. The normalized feature vector x_i is computed as (1):

$$x_i' = \frac{x_i - x_{min}}{x_{max} - x_{min}} \quad (1)$$

where x_{min} and x_{max} denote the minimum and maximum values of each feature, respectively.

After preprocessing, the normalized network traffic is supplied to a fully connected autoencoder trained solely on benign flows. The encoder transforms the input into a compact latent representation, whereas the decoder reconstructs the original feature vector from this embedding. Since the model is optimized only on normal traffic, benign samples are reconstructed with relatively small errors, whereas attack traffic generally exhibits larger reconstruction deviations.

The autoencoder parameters are optimized by minimizing the mean squared reconstruction loss,

$$L = \frac{1}{N}\sum_{i=1}^{N} |x_i - \hat{x_i}|^2 \quad (2)$$

where N represents the number of training samples, x_i is the input feature vector, and $\hat{x_i}$ denotes its reconstructed output.

A fully connected autoencoder with two hidden layers was selected to provide nonlinear feature compression while maintaining a lightweight architecture for high-dimensional network-flow data. Two latent configurations, 21 and 29 dimensions, were evaluated to examine the trade-off between information retention and noise reduction in the learned representation. ReLU and Sigmoid activation functions were employed to provide nonlinear transformation and stable reconstruction of the normalized features, respectively. The Adam optimizer with a learning rate of 0.001, 40 training epochs, and a batch size of 10,000 was selected based on preliminary experimentation, dataset characteristics, and consistent optimization across the evaluated configurations.

Rather than using the latent representation directly for attack classification, the proposed framework exploits the reconstruction behaviour of the trained autoencoder to distinguish normal and abnormal traffic before the clustering stage.

2.3. Stage II: anomaly-filtered density clustering

After training, each network flow is reconstructed by the autoencoder, and its reconstruction error is calculated as (3):

$$e_i = |x_i - \hat{x_i}|^2 \quad (3)$$

where e_i represents the reconstruction error of the i^{th} network flow. Samples exhibiting larger reconstruction errors are more likely to deviate from the learned normal behaviour.
A reconstruction threshold is then employed to identify anomalous traffic,

$$a_i = \begin{cases} 1, e_i > \tau \\ 0, e_i \leq \tau \end{cases} \quad (4)$$

where $a_i=1$ indicates an anomalous flow and τ denotes the decision threshold. To preserve the unsupervised nature of the framework, τ is determined solely from the reconstruction-error distribution of the benign training set, without utilizing attack labels. Specifically, let $\mathcal{E}_{\text{bng}} = \{e_j\}_{j=1}^{N_{\text{train}}}$ denote the reconstruction errors computed from the benign training samples. The threshold is determined as the *p-th* percentile of this empirical distribution:

$$\tau = Percentile(\mathcal{E}_{\text{bng}}, p)$$

In this study, p=99.5% is used, yielding an empirical threshold of τ=0.0142. This corresponds to approximately 0.5% of the benign training reconstruction errors exceeding the threshold. During inference, flows satisfying $e_i > \tau$ are identified as anomalous, and their latent representations are forwarded to the anomaly-filtered clustering stage.

Unlike conventional approaches that cluster the complete latent representation, the proposed method extracts only the latent vectors corresponding to anomalous samples. This anomaly-filtering step substantially reduces the influence of dominant benign traffic during density estimation and improves the separability of minority attack behaviours in the latent space.

The extracted anomaly-specific latent representations are subsequently grouped using DBSCAN [22]. For each latent vector z_i, the neighbourhood is defined as:

$$N_\epsilon(z_i) = \{z_j : |z_i - z_j| \leq \epsilon\}$$

where ϵ specifies the neighbourhood radius and $N_\epsilon(z_i)$ denotes the set of neighbouring latent vectors. Clusters are formed by connecting density-reachable samples, whereas isolated points are treated as noise [23]. Since DBSCAN does not require the number of clusters to be predefined, it is well suited for identifying attack groups with arbitrary density distributions [24].

2.4. Intrusion categorization

The clusters generated by DBSCAN represent groups of network flows exhibiting similar anomalous behaviour in the learned latent space. For experimental evaluation, each cluster is associated with an intrusion category using the dominant ground-truth label within that cluster. This mapping is performed only after clustering and is not involved in model training. Consequently, the proposed framework preserves an unsupervised learning paradigm while enabling quantitative evaluation using standard classification metrics. By separating anomaly detection from attack categorization, the framework can identify previously unseen malicious traffic while maintaining interpretable cluster structures for multiclass intrusion analysis.

3. RESULTS AND DISCUSSION

3.1. Experimental setup

All experiments were conducted in Google Colab using an NVIDIA Tesla T4 GPU (16 GB VRAM), Intel Xeon Silver 4210 CPU (20 cores), and 128 GB RAM under Ubuntu 22.04 LTS. The software environment comprised Python 3.10, TensorFlow 2.12.0 with Keras, Scikit-learn 1.3.0, NumPy 1.24.3, and Pandas 2.0.1. The autoencoder was trained for up to 40 epochs with early stopping (patience=7), requiring approximately 25 minutes per run. Random seeds were fixed for reproducibility, and the experiments were repeated five times, with results reported as mean±standard deviation.

The proposed framework was evaluated on the CSE-CIC-IDS2018 dataset [25] using the experimental configuration summarized in Table 1. The autoencoder was trained exclusively with benign traffic to learn the normal network behavior, whereas DBSCAN was subsequently applied to the anomaly-filtered latent representations for attack categorization. All experiments were conducted using the same hyperparameter settings to ensure a fair and consistent evaluation.

Table 1. Experimental dataset and model configuration

Category	Parameter	Value
Dataset	Dataset	CSE-CIC-IDS2018
	Number of features	80
	Benign instances	13,484,708
	Attack instances	2,748,235
Data split	AE training data	Benign
	DBSCAN clustering data	Infiltration, Web, Botnet, Brute Force, DoS, DDoS, and Benign
	Testing data	Infiltration, Web, Botnet, Brute Force, DoS, DDoS, and Benign
	Train–test ratio	70:30
Model	Optimizer	Adam
	Learning rate	0.001
	Epochs	40
	Hidden layers	2
	Hidden nodes	21 and 29
	Batch size	10,000
DBSCAN	MinPts	600
	ε	0.2
	Number of clusters	Automatically determined

3.2. Anomaly detection and classification performance

The anomaly detection and multiclass classification performance of the proposed framework were evaluated using two latent representations with 21 (HN-21) and 29 (HN-29) hidden nodes. The corresponding confusion matrices are presented in Figures 2 and 3, while the quantitative results are summarized in Table 2.

Figure 2(a) presents the confusion matrix obtained using the HN-21 latent representation, which correctly identifies 4,021,295 benign instances and 822,299 attack instances, with 24,117 benign instances misclassified as attacks and 2,171 attack instances misclassified as benign. Figure 2(b) illustrates the results obtained using the HN-29 latent representation, which correctly identifies 3,945,207 benign instances and 769,101 attack instances, with 100,205 benign instances misclassified as attacks and 55,369 attack instances misclassified as benign. A comparison of the two confusion matrices indicates that HN-21 produces substantially fewer misclassifications than HN-29, demonstrating better separation between benign and attack traffic. Furthermore, the multiclass confusion matrix in Figure 3 exhibits strong diagonal dominance, indicating that most attack instances are correctly assigned to their respective categories.

Table 2. Ablation analysis of latent representation learning, anomaly filtering, and DBSCAN clustering

Metrics	AE+DBSCAN		Autoencoder-only		DBSCAN-only	
	HN-29	HN-21	HN-29	HN–21	HN-29	HN-21
Recall	0.9607	0.9853	0.9328	0.9974	0.9073	0.9281
Precision	0.9581	0.9708	0.8847	0.9715	0.6375	0.7038
Specificity	0.9626	0.9837	0.9752	0.9940	0.9478	0.9607
Accuracy	0.9688	0.9881	0.9681	0.9946	0.9445	0.9579
F1-score	0.9685	0.9817	0.9081	0.9843	0.7488	0.8005

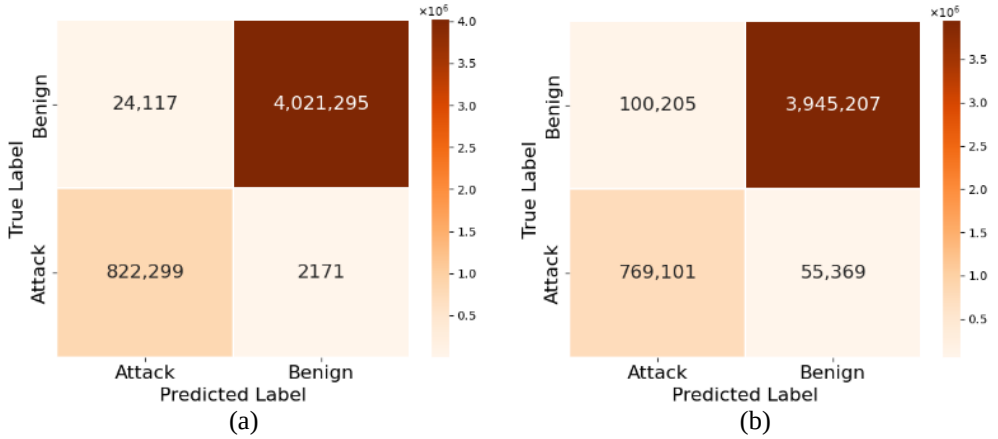


Figure 2. Confusion matrices of the autoencoder-based anomaly detector using; (a) HN-21 and (b) HN-29 latent representations

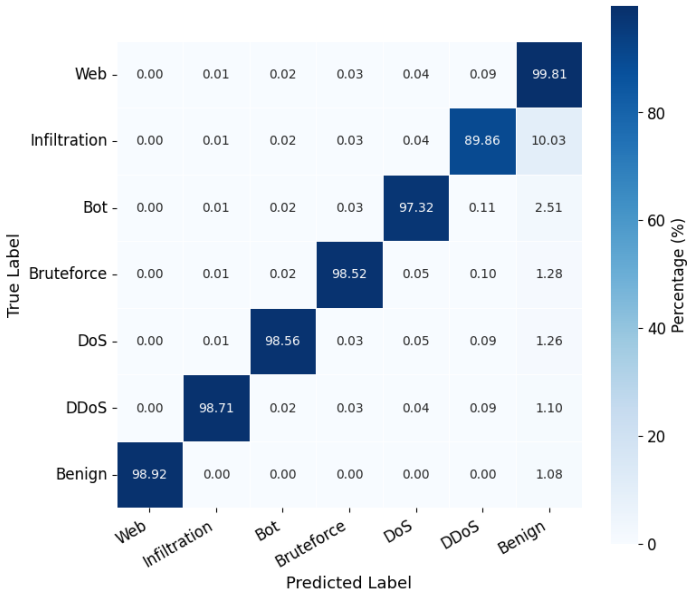


Figure 3. Confusion matrix (normalized) illustrating multiclass attack categorization achieved by the AE+DBSCAN model with the HN-21 latent embedding

The contribution of the individual components was examined through an ablation analysis using the HN-21 and HN-29 latent representations. Three configurations were compared: Autoencoder-only, which evaluates anomaly detection using the learned latent representation; DBSCAN-only, which performs direct clustering without anomaly filtering; and the proposed AE+DBSCAN, which first detects anomalies and then applies DBSCAN to the anomaly-filtered latent representations. This comparison isolates the effect of sequential anomaly filtering and density-based attack categorization.

As reported in Table 2, the proposed AE+DBSCAN framework achieves the highest multiclass classification performance with the HN-21 latent representation, attaining 98.81% accuracy, 98.53% recall, 97.08% precision, 98.37% specificity, and an F1-score of 98.17%. In comparison, direct DBSCAN clustering without anomaly filtering achieves only 95.79% accuracy and 80.05% F1-score, demonstrating the benefit of removing the dominant benign population before density-based clustering. The autoencoder-only configuration achieves higher binary anomaly-detection performance, with 99.46% accuracy and 98.43% F1-score, but cannot categorize different attack types. Thus, latent representation learning provides a compact feature space for anomaly detection, anomaly filtering reduces benign-density interference, and DBSCAN subsequently enables attack categorization. The results demonstrate that the sequential combination of these components provides the intended functionality of the overall framework.

Beyond classification performance, the selection of HN-21 is also supported by its dimensionality reduction and latent-space quality. The original feature space contains 80 dimensions, whereas HN-21 reduces this representation to 21 dimensions, corresponding to a 73.75% reduction in feature dimensionality. In comparison, HN-29 provides a 63.75% reduction. Despite the stronger compression, HN-21 achieves better latent-space clustering quality, with a Silhouette score of 0.9859 compared with 0.9654 for HN-29 and a Davies–Bouldin Index of 0.0093 compared with 0.0125 for HN-29. These results indicate that the 21-dimensional representation retains the discriminative structure required for attack separation while removing greater feature redundancy. Thus, HN-21 provides a more favorable balance between representation compactness and latent-space separability than HN-29.

3.3. Reconstruction-error threshold sensitivity analysis

The reconstruction-error threshold determines which network flows are identified as anomalous and subsequently forwarded to the anomaly-filtered DBSCAN stage. To evaluate the robustness of this decision, a threshold sensitivity analysis was conducted by varying the percentile p used to derive τ from the benign reconstruction-error distribution. Percentiles ranging from 95.0% to 99.9% were evaluated, and the corresponding anomaly-detection performance was measured using accuracy, precision, recall, FPR, and F1-score, as reported in Table 3.

Table 3. Reconstruction-error threshold sensitivity analysis

Percentile (%)	τ	Accuracy	Precision	TPR	FPR	F1
95.0	0.0084	97.41	91.20	99.88	5.00	95.34
98.0	0.0112	98.64	95.12	99.81	2.00	97.41
99.0	0.0129	99.18	97.35	99.78	1.00	98.55
99.5	0.0142	99.46	98.47	99.74	0.50	99.10
99.9	0.0185	99.21	99.30	97.82	0.10	98.55

The results demonstrate that the detection performance remains stable across the central percentile range. At lower thresholds, more benign flows are classified as anomalous, resulting in increased false-positive rates and greater benign contamination of the subsequent clustering stage. Conversely, a very high threshold reduces the number of detected anomalies and can slightly decrease recall because subtle attack flows may remain below the decision boundary. The 99.5th percentile, corresponding to τ=0.0142, provides a favorable balance between high detection capability and low false-positive rate. The threshold is therefore selected using only benign reconstruction errors, while the attack labels are used exclusively for post-hoc evaluation. This procedure preserves the unsupervised nature of the proposed framework and improves the reproducibility of the anomaly-detection stage.

3.4. Additional performance indicators

To further evaluate anomaly-detection performance under class imbalance, receiver operating characteristic–area under the curve (ROC-AUC), precision-recall AUC (PR-AUC), Matthews correlation coefficient (MCC), Cohen's Kappa, and inference latency were evaluated for the HN-21 autoencoder.

As shown in Table 4, HN-21 achieves a ROC-AUC of 0.9984 and PR-AUC of 0.9942, indicating strong discrimination across operating thresholds under class imbalance. The MCC of 0.9811 and Cohen's Kappa of 0.9810 further confirm strong agreement and balanced prediction performance. An average inference latency of 1.84 ms per 10,000-sample batch (18.4 μs/sample) indicates efficient batch processing for cloud-based traffic analysis.

Table 4. Additional performance indicators for autoencoder anomaly detection (HN-21)

Metric	HN-21
ROC-AUC	0.9984
PR-AUC	0.9942
MCC	0.9811
Cohen's Kappa	0.9810
Inference time (ms/batch)	1.84

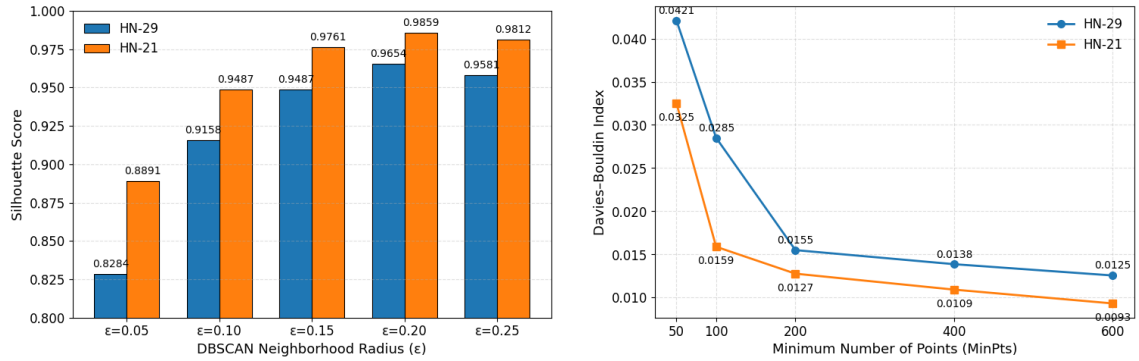
3.5. Sensitivity analysis of DBSCAN hyperparameters

Figure 4 evaluates the influence of DBSCAN parameter selection by examining the clustering behaviour of the two latent representations under different ε and MinPts settings. Clustering performance varies noticeably with changes in the neighborhood radius *(ε)* and the minimum number of neighboring points *(MinPts)*, as reflected by the Silhouette score and Davies-Bouldin index (DBI).

As illustrated in Figure 4(a), the Silhouette score increases steadily as ε increases from 0.05 to 0.20, indicating improved cluster compactness and separation. The highest scores are obtained at ε=0.20, reaching 0.9859 for HN-21 and 0.9654 for HN-29. A slight reduction at ε=0.25 suggests that a larger neighborhood radius begins to merge adjacent clusters. Across all ε values, HN-21 consistently achieves higher Silhouette scores than HN-29.

A similar trend is observed in Figure 4(b), where the DBI decreases as *MinPts* increases, reflecting improved cluster quality. The minimum DBI values of 0.0093 (HN-21) and 0.0125 (HN-29) are obtained at *MinPts*=600. These results confirm that the HN-21 latent representation produces more compact and well-separated clusters.

A parameter-sensitivity-based selection strategy was therefore adopted, in which ε and *MinPts* were systematically varied and the resulting clustering quality was evaluated using the Silhouette score and DBI. The best configuration was obtained at ε=0.20 and *MinPts*=600, which was subsequently used for the reported experiments. This empirical search provides a data-driven alternative to selecting DBSCAN parameters arbitrarily. Automated or adaptive parameter selection methods, such as data-dependent neighborhood estimation or dynamic adjustment of density parameters, could further reduce manual tuning and improve robustness across changing traffic distributions. Such adaptive strategies are considered an important direction for future extension of the proposed framework.



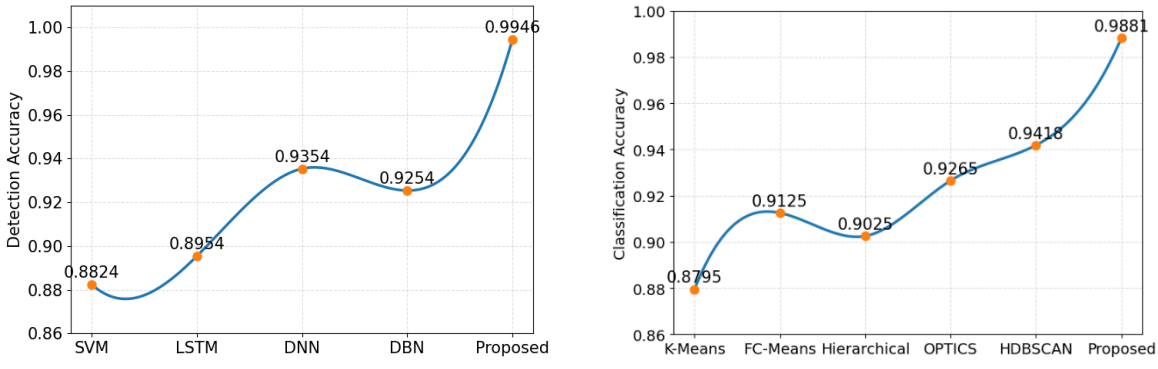
(a) (b)

Figure 4. Sensitivity analysis of DBSCAN hyperparameters; (a) Silhouette score versus neighborhood radius (ε) for the HN-21 and HN-29 latent representations and (b) DBI versus MinPts for the HN-21 and HN-29 latent representations

3.6. Comparison with baseline methods

The proposed framework was compared with representative anomaly detection and clustering methods to evaluate its effectiveness. The comparative results are presented in Figure 5. Figure 5(a) compares the anomaly detection accuracy of the evaluated methods. The proposed autoencoder records the highest accuracy (99.46%), exceeding the performance of support vector machine (SVM) (88.24%), long short-term memory (LSTM) (89.54%), deep belief network (DBN) (92.54%), and deep neural network (DNN) (93.54%). The improvement is attributed to learning latent representations directly from benign traffic, enabling the model to distinguish anomalous behavior without requiring attack labels during training.

The multiclass classification results in Figure 5(b) further demonstrate the effectiveness of the proposed framework. Under identical experimental conditions using the HN-21 anomaly-filtered latent representations, the proposed AE+DBSCAN approach achieves 98.81% accuracy, exceeding K-means (87.95%), Fuzzy C-Means (91.25%), Hierarchical Clustering (90.25%), OPTICS (92.65%), and HDBSCAN (94.18%). The comparison with OPTICS and HDBSCAN provides a direct evaluation against alternative density-based clustering approaches. The higher performance of DBSCAN indicates that its fixed-density neighbourhood formulation is well suited to the compact and well-separated anomaly structures produced by the HN-21 representation. Unlike conventional clustering methods that operate on the complete feature space, the proposed framework clusters only the anomaly-filtered latent representations produced by the autoencoder. This reduces the influence of benign traffic and improves the separation of attack categories.



(a) (b)

Figure 5. Performance comparison of the proposed framework with baseline methods; (a) anomaly detection accuracy and (b) multiclass classification accuracy using HN-21 anomaly-filtered latent representations

In addition to the existing SVM, LSTM, DBN, and DNN baselines, the evaluation includes isolation forest, variational autoencoder (VAE), deep support vector data description (Deep SVDD), and a contrastive-learning-based anomaly-detection method. The results are summarized in Table 5.

Table 5. Comparison of unsupervised anomaly-detection methods

Method	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)	ROC-AUC	PR-AUC	MCC
Isolation forest	88.35	80.14	82.60	81.35	0.8920	0.8245	0.7682
VAE	94.62	90.85	92.40	91.62	0.9570	0.9128	0.8910
Deep SVDD	93.18	88.50	89.75	89.12	0.9415	0.8872	0.8624
Contrastive learning	96.80	93.42	95.10	94.25	0.9785	0.9460	0.9348
Proposed AE (HN-21)	**99.46**	**97.15**	**99.74**	**98.43**	**0.9984**	**0.9942**	**0.9811**

As shown in Table 5, the proposed AE (HN-21) achieves the highest performance across all evaluated metrics. It obtains 99.46% accuracy, 98.43% F1-score, 0.9984 ROC-AUC, 0.9942 PR-AUC, and 0.9811 MCC, outperforming both conventional and recent unsupervised anomaly-detection methods. The contrastive-learning baseline provides the closest performance, with 96.80% accuracy and 94.25% F1-score, while VAE and Deep SVDD achieve 94.62% and 93.18% accuracy, respectively. Isolation Forest records the lowest performance among the evaluated methods. These results indicate that the proposed benign-trained autoencoder provides stronger anomaly discrimination while maintaining a compact latent representation.

To assess whether the observed performance improvements were statistically meaningful, the proposed AE (HN-21) and baseline methods were evaluated over repeated independent runs using consistent experimental settings. The results were summarized as mean ± standard deviation, with F1-score used as the primary metric because of the class imbalance in the dataset. Paired Wilcoxon signed-rank tests were conducted between the proposed method and each baseline. The proposed AE achieved an F1-score of 98.43±0.12%, compared with 81.35±0.45% for Isolation Forest, 91.62±0.28% for VAE, 89.12±0.35% for Deep SVDD, and 94.25±0.22% for contrastive learning. The corresponding *p*-values were 0.0012, 0.0028, 0.0019, and 0.0041, respectively, all below the 0.05 significance level. These results provide statistical evidence that the performance improvements of the proposed method over the evaluated baselines are statistically significant rather than attributable to random variation.

3.7. Latent-space analysis of anomaly filtering

To further examine the effect of anomaly filtering on density-based clustering, the HN-21 latent representations were visualized using t-SNE. Figure 6 compares the complete latent space with the anomaly-filtered latent space. In the complete latent space, the large benign population dominates the local density distribution, which can obscure the neighbourhood structure of minority attack behaviours (Figure 6(a)). Consequently, density estimation must operate in a space containing both the dominant normal population and comparatively sparse anomalous samples (Figure 6(b)).

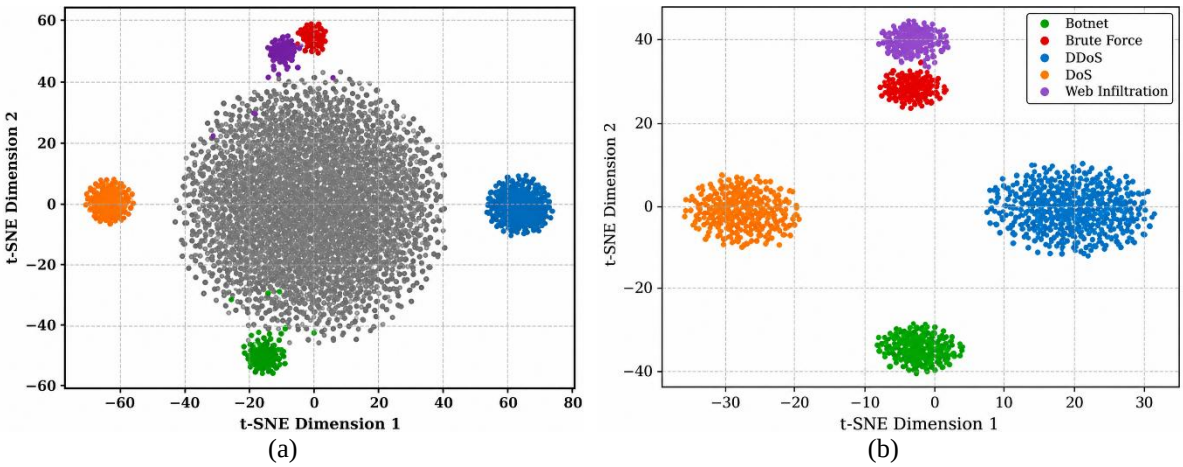


Figure 6. t-SNE visualization of the HN-21 latent space; (a) complete latent space with direct clustering and (b) anomaly-filtered latent space used by the proposed framework

After anomaly filtering, benign samples are removed before DBSCAN is applied, allowing density estimation to focus on the local structure of anomalous traffic. This reduces the influence of the dominant benign region and makes the relative separation among attack behaviours more apparent. The improved latent-space organization is consistent with the clustering results, where HN-21 achieves a high Silhouette score of 0.9859 and a low DBI of 0.0093. These observations support the proposed sequential design, in

which anomaly detection precedes density-based attack categorization rather than clustering the complete latent space directly.

3.8. Discussion

The experimental results demonstrate that filtering anomalous samples before density-based clustering substantially improves intrusion categorization under highly imbalanced cloud traffic. The autoencoder effectively learns the characteristics of benign traffic, allowing anomalous network flows to be identified through reconstruction error. Clustering only the anomaly-filtered latent representations reduces the influence of dominant benign samples, resulting in more compact attack clusters and improved separation of minority intrusion classes. The HN-21 representation also provides a favorable compression–separability trade-off. It reduces the original 80-dimensional feature space by 73.75% while achieving a higher Silhouette score and lower DBI than HN-29. Therefore, the selection of 21 dimensions is not based solely on classification accuracy; it is supported by both the compactness of the learned representation and its ability to preserve well-separated attack structures.

The evaluation considers the highly imbalanced nature of the CSE-CIC-IDS2018 dataset through class-imbalance-aware metrics, including PR-AUC, MCC, and Cohen's Kappa, as well as threshold sensitivity analysis. However, robustness under systematically varied class-imbalance ratios, noisy traffic, and incomplete or missing feature values was not explicitly evaluated in this study. Such conditions can arise in practical cloud environments and may affect reconstruction quality and density-based clustering. Future work will therefore investigate controlled imbalance levels, synthetic and real-world traffic noise, and incomplete-data scenarios to further assess the stability and generalization of the proposed framework.

Compared with supervised IDS approaches such as SVM, LSTM, DBN, and DNN, the proposed framework does not require attack labels during model training, reducing the dependence on costly and continuously updated labeled datasets. Semi-supervised approaches can reduce labeling requirements by combining labeled and unlabeled traffic, but still require a labeled subset for model development. In contrast, the proposed benign-trained autoencoder learns normal traffic characteristics and uses DBSCAN for attack categorization, providing a simpler labeling requirement. However, supervised and semi-supervised methods may offer advantages when representative labeled attack data are available, particularly for known attack categories. The proposed framework is therefore well suited to settings where labeled attack data are limited, while its fixed architecture and DBSCAN parameters may require adaptation when traffic distributions change.

From a practical deployment perspective, the proposed framework has several characteristics that support cloud deployment. The measured inference latency of 1.84 ms per 10,000 samples (18.4 μs/sample) indicates low processing latency for the autoencoder stage. The 21-dimensional latent representation reduces the amount of data passed to the clustering stage, while anomaly filtering prevents the dominant benign traffic from being processed by DBSCAN, thereby reducing the effective clustering workload. As network traffic increases, the autoencoder inference stage can be parallelized across processing nodes, while anomaly-specific latent representations can be forwarded to dedicated clustering resources. The sequential architecture is therefore compatible with distributed cloud deployments in which detection and clustering functions can be placed at different processing layers. However, memory consumption and large-scale distributed throughput were not explicitly benchmarked in this study and require further evaluation under production-scale workloads.

The proposed sequential architecture could be extended to online cloud traffic monitoring by processing network flows in successive batches or sliding windows. The autoencoder could continuously process incoming flows using the learned benign representation, while reconstruction errors could identify anomalous samples for subsequent density-based clustering. To accommodate evolving traffic distributions, the benign reference distribution and reconstruction-error threshold could be periodically updated using recent trusted traffic, while DBSCAN parameters could be dynamically adjusted according to changing local density. Incremental or window-based clustering could further support continuous attack categorization without repeatedly processing the complete historical dataset. Such an online extension could support adaptation to changing traffic patterns and emerging attack behaviours while retaining the separation between anomaly detection and attack categorization. However, streaming implementation and concept-drift adaptation were not evaluated in the present study and remain important directions for future work.

Future work will also investigate explainable AI techniques to improve the interpretability of reconstruction-based anomaly decisions and DBSCAN-derived attack clusters. Feature-level attribution methods could identify the input characteristics that contribute most strongly to reconstruction errors, while cluster-level explanations could characterize the features associated with different attack groups. Such explanations could assist security analysts in validating detected anomalies and understanding attack patterns in operational cloud environments.

4. CONCLUSION

This paper presented a dual-stage unsupervised intrusion detection framework that combines autoencoder-based anomaly detection with DBSCAN-based attack categorization for cloud environments. Unlike conventional approaches that cluster the complete latent space, the proposed framework performs clustering only on anomaly-filtered latent representations, reducing the influence of benign traffic during density estimation. Experimental evaluation on the CSE-CIC-IDS2018 dataset demonstrated 99.46% anomaly detection accuracy and 98.81% multiclass attack classification accuracy. Sensitivity analysis further identified ε=0.20 and MinPts=600 as the selected DBSCAN configuration. Despite these results, the study has several limitations. The evaluation relies on a single benchmark dataset and an offline experimental setting, which may not fully represent heterogeneous and continuously evolving cloud traffic. In addition, the current framework uses fixed DBSCAN hyperparameters and a fixed autoencoder architecture and does not dynamically adapt to changing traffic distributions or emerging attack behaviours. The scalability and memory requirements of distributed, production-scale deployment were also not explicitly benchmarked. Future work will focus on validating the framework across multiple cloud and IoT intrusion datasets, developing adaptive density-clustering mechanisms for changing traffic distributions, and incorporating online learning to address evolving attack behaviours. Further investigation will also consider distributed and real-time streaming deployment, including integration with cloud security management platforms, software-defined networking (SDN) environments, and edge-cloud architectures, together with large-scale resource and memory profiling. Such integration could enable anomaly detection closer to traffic sources while coordinating attack categorization and security responses across distributed cloud infrastructures. Future work will focus on validating the framework across multiple cloud and IoT intrusion datasets, developing adaptive density-clustering mechanisms for changing traffic distributions, incorporating online learning to address evolving attack behaviours, and integrating explainable AI techniques to improve the interpretability of anomaly and attack-clustering decisions.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Sant Kumar Maurya	✓			✓		✓		✓	✓		✓			
Karuppiah Tamilarasi		✓	✓		✓		✓			✓				
Sunil Kumar		✓		✓	✓			✓		✓		✓		
Gaurav Vinchurkar	✓		✓			✓		✓		✓				
Praveen Kumar Gupta		✓	✓			✓	✓			✓		✓		
Anubhav Kumar	✓			✓	✓		✓		✓				✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author, einitials: A.K], upon reasonable request.

REFERENCES

[1] N. Dharini, V. S. Janani, and J. Katiravan, "Efficient detection of intrusions in TON-IoT dataset using hybrid feature selection approach," *Scientific Reports*, vol. 16, no. 1, pp. 1–29, Feb. 2026, doi: 10.1038/s41598-026-37834-y.

[2] M. M. Rahman, S. Al Shakil, and M. R. Mustakim, "A survey on intrusion detection system in IoT networks," *Cyber Security and Applications*, vol. 3, p. 100082, Dec. 2025, doi: 10.1016/j.csa.2024.100082.

[3] A. Elayan and M. Kadoch, "Enhancing IoT Network Security: A BPSO-Optimized Attention-GRU Deep Learning Framework for Intrusion Detection," *Computers*, vol. 15, no. 5, pp. 1–31, Apr. 2026, doi: 10.3390/computers15050266.

[4] A. A. Yılmaz, "A novel deep learning-based framework with particle swarm optimisation for intrusion detection in computer networks," *PLOS ONE*, vol. 20, no. 2, p. e0316253, Feb. 2025, doi: 10.1371/journal.pone.0316253.

[5] N. Ijaz, S. U. Jan, and I. Koo, "A Robust Residual Autoencoder Framework for Anomaly-Based Network Intrusion Detection," in *Proceedings of the 2026 7th International Conference on Advancements in Computational Sciences (ICACS)*, 2026, pp. 1–6, doi: 10.1109/ICACS69208.2026.11433153.

[6] M. Luo *et al.*, "HIoT: heterogeneous information network based compromised IoT device detection," *Cybersecurity*, vol. 9, no. 1, p. 55, Apr. 2026, doi: 10.1186/s42400-025-00465-3.

[7] S. Dardouri, "A Deep Learning Approach Using Optimized LSTM for Anomaly-Based Network Intrusion Detection," *Journal of Engineering (United Kingdom)*, no. 1, pp. 1–13, Jan. 2026, doi: 10.1155/je/2021709.

[8] A. Sahi, D. Lai, Y. Li, and M. Diykh, "An Efficient DDoS TCP Flood Attack Detection and Prevention System in a Cloud Environment," *IEEE Access*, vol. 5, pp. 6036–6048, 2017, doi: 10.1109/ACCESS.2017.2688460.

[9] S. Rameshkumar, R. Ganesan, and A. Merline, "Progressive Transfer Learning-based Deep Q Network for DDOS Defence in WSN," *Computer Systems Science and Engineering*, vol. 44, no. 3, pp. 2379–2394, 2023, doi: 10.32604/csse.2023.027910.

[10] H. V. M.V. and J. T., "A hybrid model for detecting intrusions using stacked autoencoders and extreme gradient boosting," *Computers & Security*, vol. 150, p. 104212, Mar. 2025, doi: 10.1016/j.cose.2024.104212.

[11] P. Kaliyaperumal, S. Periyasamy, M. Thirumalaisamy, B. Balusamy, and F. Benedetto, "A Novel Hybrid Unsupervised Learning Approach for Enhanced Cybersecurity in the IoT," *Future Internet*, vol. 16, no. 7, pp. 1–29, Jul. 2024, doi: 10.3390/fi16070253.

[12] W. H. Aljuaid and S. S. Alshamrani, "A Deep Learning Approach for Intrusion Detection Systems in Cloud Computing Environments," *Applied Sciences*, vol. 14, no. 13, pp. 1–19, Jun. 2024, doi: 10.3390/app14135381.

[13] Z. Long, H. Yan, G. Shen, X. Zhang, H. He, and L. Cheng, "A Transformer-based network intrusion detection approach for cloud security," *Journal of Cloud Computing*, vol. 13, no. 1, p. 5, Jan. 2024, doi: 10.1186/s13677-023-00574-9.

[14] M. Sajid *et al.*, "Enhancing intrusion detection: a hybrid machine and deep learning approach," *Journal of Cloud Computing*, vol. 13, no. 1, pp. 1–24, Jul. 2024, doi: 10.1186/s13677-024-00685-x.

[15] L. Megouache, A. Zitouni, S. Sadouni, and M. Djoudi, "Machine Learning for Cloud Data Classification and Anomaly Intrusion Detection," *Ingenierie des Systemes d'Information*, vol. 29, no. 5, pp. 1809–1819, Oct. 2024, doi: 10.18280/isi.290514.

[16] J. Li, H. Sun, H. Du, L. Li, and Z. Zhang, "Network Intrusion Detection Method Based on Semi-Supervised Learning and Random Forest," *IEICE Transactions on Communications*, vol. E108.B, no. 10, pp. 1152–1163, Oct. 2025, doi: 10.23919/transcom.2024EBP3204.

[17] E. Alharbi, L. S. Marcolino, A. Gouglidis, and Q. Ni, "Robust Federated Learning Method Against Data and Model Poisoning Attacks with Heterogeneous Data Distribution," in *Frontiers in Artificial Intelligence and Applications*, vol. 372, 2023, pp. 85–92, doi: 10.3233/FAIA230257.

[18] R. Dugyala *et al.*, "Secure cloud computing: leveraging GNN and leader K-means for intrusion detection optimization," *Scientific Reports*, vol. 14, no. 1, pp. 1–25, Dec. 2024, doi: 10.1038/s41598-024-81442-7.

[19] W. Villegas-Ch, J. Govea, A. M. Navarro, and P. P. Játiva, "Intrusion Detection in IoT Networks Using Dynamic Graph Modeling and Graph-Based Neural Networks," *IEEE Access*, vol. 13, pp. 65356–65375, 2025, doi: 10.1109/ACCESS.2025.3559325.

[20] W. Guo, B. Tondi, and M. Barni, "Universal Detection of Backdoor Attacks via Density-Based Clustering and Centroids Analysis," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 970–984, 2024, doi: 10.1109/TIFS.2023.3329426.

[21] P. Artioli, A. Maci, and A. Magrì, "A comprehensive investigation of clustering algorithms for User and Entity Behavior Analytics," *Frontiers in Big Data*, vol. 7, pp. 1–24, May 2024, doi: 10.3389/fdata.2024.1375818.

[22] G. Muhammad, M. S. Hossain, and S. Garg, "Stacked Autoencoder-Based Intrusion Detection System to Combat Financial Fraudulent," *IEEE Internet of Things Journal*, vol. 10, no. 3, pp. 2071–2078, Feb. 2023, doi: 10.1109/JIOT.2020.3041184.

[23] K. Garcia-Sanchez, J. L. Perez-Ramos, S. Ramirez-Rosales, A. M. Herrera-Navarro, H. Jiménez-Hernández, and D. Canton-Enriquez, "Causal Interpretation of DBSCAN Algorithm: A Dynamic Modeling for Epsilon Estimation," *Entropy*, vol. 28, no. 4, pp. 1–23, Apr. 2026, doi: 10.3390/e28040452.

[24] H. Yang, "Adaptive Clustering and Machine-Learning-Based DoS Intrusion Detection in MANETs," *Applied Sciences*, vol. 16, no. 6, pp. 1–25, Mar. 2026, doi: 10.3390/app16062723.

[25] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *ICISSP 2018 - Proceedings of the 4th International Conference on Information Systems Security and Privacy*, SCITEPRESS - Science and Technology Publications, 2018, pp. 108–116, doi: 10.5220/0006639801080116.

BIOGRAPHIES OF AUTHORS



Sant Kumar Maurya is an assistant professor in the Department of Computer Science and Engineering (AIT-CSE), Chandigarh University, Mohali, Punjab. His teaching and research interests include artificial intelligence, machine learning, and computer science applications. He has actively mentored winning teams in the Smart India Hackathon (SIH) 2023 and 2024 and serves as an event coordinator. He is committed to academic excellence, research, and innovation in emerging computing technologies. He can be contacted at email: takecr@live.com.



Dr. Karuppiah Tamilarasi is a professor in Department of Information Technology at Panimalar Engineering College, accumulating 25 years of teaching experience. She earned her Ph.D. record with 9 patents, 7 book chapters, and 25 research papers published in esteemed international journals and conferences. Her expertise spans cyber security, networks, cloud computing, and machine learning. She can be contacted at email: thamizhanna@gmail.com.



Dr. Sunil Kumar completed his B.Tech. from BIET Jhansi, M.Tech. from YMCA Institute of Engineering, Fariabad and Ph.D. from I. K. Gujral Punjab Technical University, Kapurthala, all in Computer Science and Engineering. He is currently working as a professor in the Department of Information Technology at the Ajay Kumar Garg Engineering College, Ghaziabad, Uttar Pradesh. His research interests lie in computer networks, wireless networks, cryptography and network security, and machine learning. He can be contacted at email: sunilymca2k5@gmail.com.



Gaurav Vinchurkar is an assistant professor at Galgotias University with over 13 years of teaching experience in Computer Science and Information Technology. He is currently pursuing a Ph.D. and specializes in data structures, design and analysis of algorithms, and theory of computation. His research interests include machine learning, intrusion detection systems, and smart grid technologies. He has qualified GATE multiple times and has published research in reputed international journals and conferences. He can be contacted at email: gbvinchurkar@gmail.com.



Dr. Praveen Kumar Gupta is a professor with Department of AIML, School of Computer Science and Engineering, highest degree is Ph.D. (Computer Science and Engineering), with total experience of more than 33 years in post-graduate academic institutions, including the ICT industry experience. His research interests are AI, ML, DL, quantum computing, and AI ethics in public policy. In recent past, he has been actively focusing interdisciplinary research at the intersection of healthcare, agentic AI and computer vision. He can be contacted at email: onlinepg@gmail.com.



Dr. Anubhav Kumar is a professor and dean of Academic Affairs in the School of Engineering at HRIT University, Ghaziabad. Holding a Ph.D. and bringing 24 years of teaching experience, his core expertise spans machine learning, deep learning, data science, NLP, networks and big data. He has a strong research portfolio featuring 15 patents, 12 book chapters, and 29 research papers published in esteemed international journals and conferences. He can be contacted at email: dr.anubhavkumar@gmail.com.