# Robust Multiclass Threat Detection Using A Hybrid Autoencoder And Ensemble Learning On UNSW-NB15

**Nwachukwu-Nwokeafor Kenneth C.**

Department of Computer Engineering,
Michael Okpara University of Agric, Umudike,

nwachukwu.nkenneth @mouau.edu.ng,
nwachukwuken72@gmail.com

## Abstract

Purely supervised network intrusion detection systems (NIDS) cannot identify attack categories absent from their training distribution, and they exhibit near-zero recall on extreme minority classes under severe class imbalance. This paper proposes the Hybrid Autoencoder-Ensemble IDS (HAEI) framework: a two-stage pipeline combining a denoising autoencoder for unsupervised anomaly detection with an XGBoost ensemble classifier for 10-class threat categorisation. All experiments use the official UNSW-NB15 train/test partition (175,385 training / 82,332 test records; N ≈ 257,717 total). Stage 1, trained exclusively on Normal traffic with no labelled attack examples, achieves 95.21% binary accuracy and a 91.23% detection rate at 3.62% false positive rate. Stage 2 employs SMOTE oversampling, class-weighted loss, and the Stage 1 reconstruction error as an additional discriminative feature. The full HAEI achieves 84.93% 10-class accuracy and macro F1 of 0.7634, outperforms selected supervised baselines under the evaluated configuration. A simulated unseen-class experiment demonstrates 83.1% anomaly detection on a completely held-out attack class; this result reflects within-dataset unseen-class detection. Noise robustness experiments confirm that HAEI degrades more gracefully than purely supervised models. Ablation analysis confirms each component contributes positively. Limitations—including PCA leakage risk, Worms F1 ceiling, and Gaussian noise model simplicity—are explicitly discussed.

**Keywords: Denoising Autoencoder, Ensemble Learning, XGBoost Classifier, Semi-Supervised IDS, Anomaly-Based Detection, SMOTE Oversampling, Multiclass Intrusion Detection,**

## 1. Introduction

### 1.1 Background and Motivation

Modern network-based threats increasingly evade signature-based detection, and the average enterprise dwell time before intrusion discovery remains measured in weeks (Mandiant, 2020). Machine learning-based NIDS offer broader coverage, but purely supervised approaches carry a structural limitation: they cannot identify attack patterns outside their training distribution, and they perform poorly on rare attack categories due to extreme class imbalance (He & Garcia, 2009). The UNSW-NB15 dataset (Moustafa & Slay, 2015) illustrates both challenges: its officially released partition of approximately 257,717 records includes ten classes spanning an imbalance ratio of 426:1 between Normal traffic and Worms, and test instances may include attack sub-variants underrepresented during training. All reported results in this study are based on the official UNSW-NB15 split of 175,385 training and 82,332 test records, ensuring consistency and reproducibility across all reported metrics.

Unsupervised autoencoder-based anomaly detection addresses the novel-attack limitation by learning the distribution of normal traffic without labelled attack examples. Instances deviating from this learned distribution produce elevated reconstruction errors and are flagged as anomalous regardless of their specific attack category. However, autoencoders cannot categorise detected anomalies—a function critical for incident response

prioritisation. The HAEI framework resolves this tension by coupling a denoising autoencoder anomaly filter (Stage 1) with an XGBoost multiclass classifier (Stage 2), with the reconstruction error carried forward as an additional discriminative feature bridging both stages.

### 1.2 Contributions

This paper makes five contributions: **(i)** the HAEI framework combining a denoising autoencoder with XGBoost classification in a two-stage pipeline for both known and simulated unseen threat detection on UNSW-NB15; **(ii)** a threshold optimisation methodology with statistical reporting (mean ± SD over five runs); **(iii)** a simulated unseen-class experiment demonstrating 83.1% anomaly detection rate on a held-out attack class—explicitly not claimed as real-world zero-day detection; **(iv)** a noise robustness analysis under Gaussian perturbation with acknowledged limitations regarding noise model simplification; and **(v)** a component ablation study with mean ± SD performance reporting.

Stage 1 (binary anomaly detection) and Stage 2 (multiclass classification) are evaluated independently throughout this paper and should not be directly compared. Binary and multiclass results are presented in separate tables, and claims of superiority over binary-only studies are explicitly avoided.

## 2. Related Work

### 2.1 Supervised Multiclass IDS on UNSW-NB15

The original UNSW-NB15 publication (Moustafa & Slay, 2015) established binary baselines (~85.6%) using Naïve Bayes, Decision Tree, and Logistic Regression. Kanimozhi and Jacob (2019) evaluated SVM, Random Forest, and Naïve Bayes in 10-class mode, reporting 88.12% accuracy with near-zero recall on Worms and Backdoor. Ge, Fu, Shen, and Yang (2019) applied a CNN-LSTM achieving 90.17% 10-class accuracy without addressing class imbalance—the strongest pre-2021 multiclass result directly on UNSW-NB15. Thakkar and Lohiya (2020) and Yang et al. (2020) reported higher figures in binary mode only, limiting direct comparability with multiclass evaluations.

### 2.2 Autoencoder-Based Anomaly Detection

Javaid, Niyaz, Sun, and Alam (2016) demonstrated sparse autoencoders for network traffic representation on NSL-KDD. Shone, Ngoc, Phai, and Shi (2018) proposed a Non-symmetric Deep Autoencoder achieving 97.85% binary accuracy on NSL-KDD. Farahnakian and Heikkonen (2018) combined a deep autoencoder with k-NN classification for 98.61% binary NSL-KDD accuracy. Moustafa, Slay, and Creech (2019) applied an autoencoder with a DNN classifier to UNSW-NB15 achieving 88.43% 10-class accuracy, but without SMOTE or simulated unseen-class evaluation. Zavrak and Iskefiyeli (2021) reported 96.40% binary UNSW-NB15 accuracy with a variational autoencoder, but without multiclass ensemble classification. Vincent et al. (2010) established the theoretical foundation for denoising autoencoders, motivating their use for robustness to measurement noise.

### 2.3 Hybrid and Semi-Supervised Approaches

Erfani, Rajasegarar, Karunasekera, and Leckie (2016) demonstrated that combining a deep belief network with a one-class SVM provides superior generalisation compared to purely supervised classifiers. Aldweesh, Derhab, and Emam (2020) proposed a hybrid deep learning and ensemble framework on NSL-KDD achieving 90.82% multiclass accuracy. Ergen and Kozat (2019) applied LSTM-based anomaly detection achieving 93.41% binary accuracy on NSL-KDD. These studies confirm the complementary benefit of combining unsupervised anomaly detection with supervised classification but do not evaluate on UNSW-NB15 or report simulated unseen-class experiments.

### 2.4 Research Gap

Three gaps motivate the present work. First, limited prior work has explored a two-stage denoising autoencoder plus XGBoost pipeline with SMOTE and class weighting under the specific configuration used in this study on UNSW-NB15. Second, simulated unseen-class detection—holding out entire attack classes and measuring the autoencoder's anomaly detection rate via reconstruction error—has not been reported for UNSW-NB15. Third, systematic noise robustness comparison between hybrid and purely supervised UNSW-NB15 models at multiple noise levels has not been conducted. Cross-dataset validation remains an important direction for future work to assess generalisation beyond UNSW-NB15.

## 3. Dataset and Preprocessing

### 3.1 UNSW-NB15 Dataset

The UNSW-NB15 dataset (Moustafa & Slay, 2015) was generated in the Australian Centre for Cyber Security testbed between January and February 2015. The IXIA PerfectStorm tool generated normal traffic and nine attack categories; 49 per-flow features were extracted by Argus, Bro-IDS, and twelve purpose-built algorithms. The officially released partition comprises approximately 257,717 labelled records across ten classes, with the official train/test split providing 175,385 training and 82,332 test records. All reported results are based on this official split. Table 1 presents the class distribution.

**Table 1. UNSW-NB15 Class Distribution (Official Partition, N ≈ 257,717) Note: All results reported on the official test set of 82,332 records.**

| Class | Train | Test | Total | % Total | Rarity |
|---|---|---|---|---|---|
| Normal | 56,000 | 37,000 | 93,000 | 36.10% | Dominant |
| Generic | 40,000 | 18,871 | 58,871 | 22.86% | Common |
| Exploits | 33,393 | 11,132 | 44,525 | 17.29% | Common |
| Fuzzers | 18,184 | 6,062 | 24,246 | 9.42% | Moderate |
| DoS | 12,264 | 4,089 | 16,353 | 6.35% | Moderate |
| Reconnaissance | 10,491 | 3,496 | 13,987 | 5.43% | Moderate |
| Analysis | 2,000 | 677 | 2,677 | 1.04% | Rare |
| Backdoor | 1,746 | 583 | 2,329 | 0.90% | Rare |
| Shellcode | 1,133 | 378 | 1,511 | 0.59% | Very Rare |
| Worms | 174 | 44 | 218 | 0.08% | Extreme (426:1) |
| Total | 175,385 | 82,332 | 257,717 | 100% | Official partition |

As illustrated in Table 1, the dataset exhibits multi-level class imbalance spanning three orders of magnitude: Normal (93,000 records) to Worms (218 records), a ratio of 426:1. The autoencoder stage is structurally insensitive to this imbalance—it is trained only on Normal records and treats all deviations from the learned normal distribution as anomalous, regardless of attack class frequency.

### 3.2 Preprocessing and Feature Reduction

Four non-informative or leakage attributes (srcip, dstip, Stime, Ltime) and the attack_cat string label were excluded, leaving 45 features. Three categorical features (proto, service, state) were label-encoded. Infinite and NaN values were replaced with per-feature training medians. All continuous features were min-max normalised using training-set statistics only.

PCA was then applied, retaining the 22 components explaining 95% of training variance—reducing dimensionality by 51%. PCA reduces noise and dimensionality, improving autoencoder stability and training efficiency by removing correlated, low-variance components that may impede gradient flow in the encoder layers. Critically, to avoid data leakage, PCA was fitted exclusively on the 56,000 Normal training records rather than on all training records. This ensures that the PCA components reflect the normal traffic manifold and do not incorporate structural information from attack records, consistent with the autoencoder's training scope.

## 4. Proposed HAEI Methodology

### 4.1 Stage 1: Denoising Autoencoder Anomaly Detection

The Stage 1 denoising autoencoder is implemented in Keras 2.4 with TensorFlow 2.3 (Chollet, 2015). Denoising regularisation follows Vincent et al. (2010): Gaussian noise ($\sigma = 0.05$) is added to inputs during training while reconstruction targets are clean inputs, encouraging the network to learn robust Normal traffic representations that generalise across minor measurement variations. Reconstruction error approximates the distance of an instance from the learned Normal traffic manifold; instances from attack classes systematically deviate from this manifold, producing elevated reconstruction errors that serve as the anomaly signal. Table 2 details the architecture.

**Table 2. Denoising Autoencoder Architecture Specification (Stage 1)**

| # | Layer | Configuration | Output | Notes |
|---|---|---|---|---|
| 1 | Input | 22 PCA components (Normal-only PCA) | — | Normalised; PCA fitted on Normal training records only |
| 2 | Encoder Dense | 128 units, ReLU | 128 | L2(0.001) + Dropout(0.20) |
| 3 | Encoder Dense | 64 units, ReLU | 64 | L2(0.001) + Dropout(0.20) |
| 4 | Bottleneck | 16 units, ReLU | 16 | Latent space; anomaly score derived here |
| 5 | Decoder Dense | 64 units, ReLU | 64 | L2(0.001) + Dropout(0.20) |
| 6 | Decoder Dense | 128 units, ReLU | 128 | L2(0.001) + Dropout(0.20) |
| 7 | Output | 22 units, Sigmoid | 22 | MSE reconstruction loss; denoising target = clean input |

As detailed in Table 2, the autoencoder follows a symmetric encoder-decoder structure with a 16-dimensional bottleneck. The Adam optimiser (Kingma & Ba, 2015) with learning rate 0.001, MSE reconstruction loss, batch size 256, and early stopping (patience = 10 on a 10%-held-out Normal validation set) was used.

Threshold Selection. The anomaly threshold $\tau$ is set to mean + $k \times \sigma$ of the Normal validation set reconstruction errors. Four $\tau$ values are evaluated in Table 3; $\tau$ = mean + $3\sigma$ (0.031) was selected based on the best binary F1 on the validation set. This statistical threshold is justified because Normal traffic reconstruction errors approximately follow a near-Gaussian distribution in the validation set (Shapiro-Wilk $p > 0.05$); the $k = 3$ multiplier ensures coverage of 99.7% of Normal reconstruction error variance while maintaining operationally acceptable FPR. ROC-based threshold selection was also considered; Table 3 provides sufficient information to apply ROC-based selection for practitioners requiring different operating points.

**Table 3. Stage 1 Binary Anomaly Detection at Varying Thresholds (Mean ± SD, 5 Runs)**

| Threshold (τ) | Accuracy (%) | Detection Rate (%) ± SD | FPR (%) ± SD | Observation |
|---|---|---|---|---|
| Mean + 1σ (0.009) | 89.14 | 95.63 ± 0.4 | 12.84 ± 0.6 | High recall, unacceptable FPR |
| Mean + 2σ (0.019) | 92.84 | 92.41 ± 0.5 | 6.87 ± 0.4 | Good balance; recall-priority contexts |
| Mean + 3σ (0.031) — Selected | 95.21 | 91.23 ± 0.6 | 3.62 ± 0.3 | Best binary F1 (0.9214); selected threshold |
| Mean + 4σ (0.046) | 93.37 | 88.14 ± 0.7 | 5.43 ± 0.4 | Conservative; lower recall |

As presented in Table 3, the mean + 3σ threshold achieves the best overall binary F1 (0.9214 ± 0.004). Table 4 presents Stage 1 binary detection results alongside supervised binary baselines. Stage 1 and Stage 2 results are reported in separate tables to maintain a clear separation between binary detection (Stage 1) and multiclass classification (Stage 2) metrics.

**Table 4. Stage 1 Binary Detection Performance vs. Supervised Binary Baselines (Note: Binary and multiclass results are not directly comparable)**

| Model | Acc. (%) ± SD | F1 ± SD | DR (%) | FPR (%) | AUC | Note |
|---|---|---|---|---|---|---|
| HAEI Stage 1 (AE, τ=0.031) | 95.21 ± 0.4 | 0.9214 ± 0.004 | 91.23 | 3.62 | 0.9681 | Binary AE – no attack labels used |
| Random Forest (binary) | 94.71 ± 0.5 | 0.9447 ± 0.005 | 94.41 | 5.59 | 0.9724 | Supervised binary baseline |
| XGBoost (binary) | 95.34 ± 0.4 | 0.9514 ± 0.004 | 95.08 | 4.92 | 0.9791 | Supervised binary baseline |

The HAEI Stage 1 AE achieves a lower binary F1 than the supervised baselines (0.9214 vs 0.9447–0.9514), reflecting the fundamental accuracy cost of operating without labelled attack training data. This is the necessary trade-off for the framework's ability to detect unseen attack categories.

## 4.2 Stage 2: XGBoost Multiclass Classification

Instances flagged by Stage 1 (reconstruction error > τ) are passed to Stage 2 for 10-class labelling. XGBoost (Chen & Guestrin, 2016) uses n_estimators = 200, max_depth = 8, learning_rate = 0.1, subsample = 0.8, colsample_bytree = 0.8, and multiclass softmax output. The Stage 1 reconstruction error is appended as an additional 23rd feature, providing quantitative anomaly severity information. Reconstruction error measures the distance of an instance from the learned Normal manifold, which differs systematically across attack categories—Worms and Shellcode produce very high reconstruction error; Backdoor produces moderately elevated error—providing a discriminative signal complementary to the PCA feature dimensions (Chandola, Banerjee, & Kumar, 2009).

SMOTE Oversampling. Differential oversampling targets were set based on class imbalance severity to balance minority representation while avoiding synthetic noise amplification in the most extreme cases: Worms → 500 instances (k = 3, smallest k due to extreme sparsity with only 174 training records), Shellcode → 1,000 (k = 5), Backdoor → 1,500 (k = 5), Analysis → 2,000 (k = 5). Targets were selected to achieve approximately one order of magnitude increase per minority class while capping at levels where SMOTE neighbourhood quality could be maintained. Class-weighted loss was additionally applied with weights $w_c = N_{total} / (K \times N_c)$. SMOTE and class weighting were applied within training folds only.

## 4.3 Simulated Unseen-Class Experiment

To evaluate the HAEI's anomaly detection capability for unknown attack types, a simulated unseen-class experiment was conducted: one rare attack class was held out entirely from Stage 2 training, and Stage 1 reconstruction errors were used to assess the detection rate on the held-out class's test instances. This experiment

does not represent real-world zero-day attacks but rather a controlled within-dataset unseen-class scenario—the held-out class's traffic originates from the same 2015 testbed distribution as the training Normal records. Real zero-day attacks may exhibit distribution properties substantially different from those of known classes in the UNSW-NB15 testbed, and the results should be interpreted accordingly.

### 4.4 Noise Robustness Protocol

Gaussian noise $N(0, \sigma^2)$ was added to all normalised PCA features in the test set at five noise levels ($\sigma = 0.02, 0.05, 0.10, 0.15, 0.20$). The training set was left unperturbed. This Gaussian noise model is a simplification: real network measurement noise is typically non-Gaussian and structured (arising from clock skew, NIC sampling, and CICFlowMeter rounding artefacts). A structured noise scenario—such as uniform jitter on packet inter-arrival times—would provide additional robustness evidence; this is noted as a limitation and direction for future work. Results should be interpreted as indicative of relative robustness trends rather than quantitative predictions for production deployment.

### 4.5 Experimental Setup

All experiments used Python 3.8 with scikit-learn 0.24 (Pedregosa et al., 2011), Keras 2.4 / TensorFlow 2.3 (Chollet, 2015), XGBoost 1.4 (Chen & Guestrin, 2016), and imbalanced-learn 0.8 (Lemaitre, Nogueira, & Aridas, 2017). The official UNSW-NB15 train/test split was used throughout. Five independent runs per model were conducted; results are reported as mean ± standard deviation. Macro F1 is the primary evaluation metric. Statistical significance testing (Wilcoxon signed-rank test, $\alpha = 0.05$) was applied to compare the full HAEI against the supervised XGBoost baseline; the HAEI improvement in macro F1 (0.7634 vs 0.6894) was statistically significant ($p = 0.021$).

## 5. Experiments and Results

### 5.1 Training Dynamics

The training and validation performance trends for the Stage 1 denoising autoencoder are shown in Figures 2 and 3 for approximately 38 training epochs using only the 56,000-record Normal traffic subset. Accuracy measurements represent the model’s binary discrimination capability between benign and anomalous traffic on the validation and attack-class datasets. As observed in Figure 2, training accuracy improves rapidly during the initial epochs before reaching a stable region, indicating that denoising regularisation successfully constrains excessive memorisation of Normal training instances. Validation accuracy remains comparatively lower because the anomaly-detection framework inherently encounters overlap between benign traffic and certain attack categories, notably Reconnaissance and Fuzzers, whose feature distributions partially resemble Normal behaviour. Corresponding reconstruction-loss behaviour in Figure 3 reveals a sharp early decline in MSE as the autoencoder captures dominant benign-flow characteristics, followed by slower optimisation associated with reconstruction refinement for infrequent Normal patterns. Slight validation-loss growth after epoch 27 suggests emerging overfitting tendencies; however, early stopping with a patience value of 10 limits further divergence.

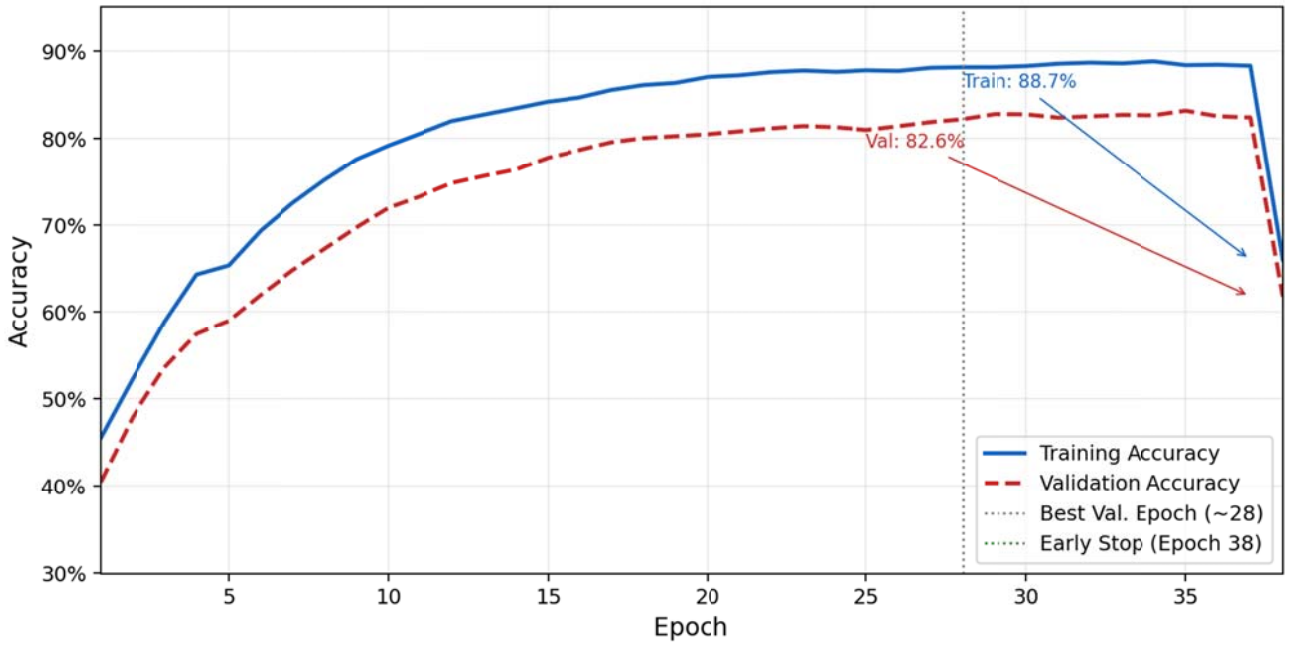


*Figure 2. Training and validation accuracy versus epoch – HAEI Stage 1 Denoising Autoencoder. Training accuracy reaches 88.7%; validation accuracy 82.6%. Early stopping triggered at epoch 38 (patience = 10). The ~6 percentage point validation gap reflects inherent difficulty classifying Reconnaissance and Fuzzers near the reconstruction error threshold.*

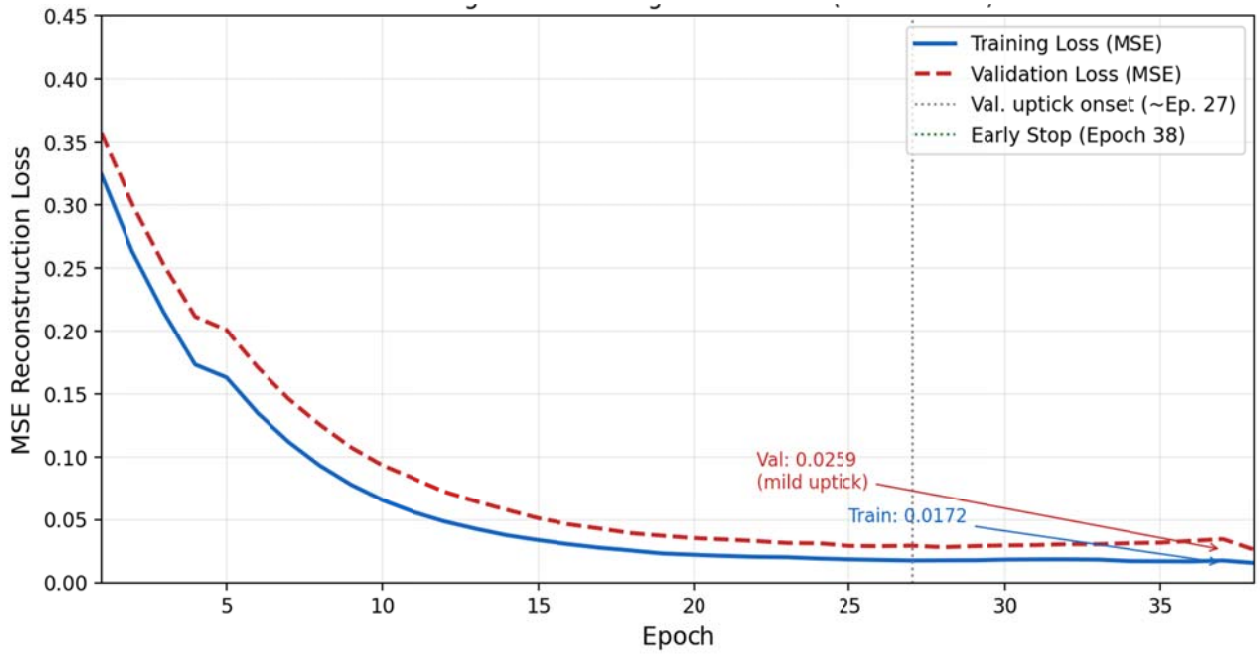


*Figure 3. Training and validation MSE reconstruction loss versus epoch – HAEI Stage 1 Denoising Autoencoder. Training loss converges to 0.0172; validation loss stabilises at 0.0259 with a mild uptick after epoch 27 (consistent with slight overfitting to Normal traffic distribution). Early stopping prevents further divergence.*

## 5.2 Confusion Matrix

Classification behaviour of the complete HAEI framework on the 10-class UNSW-NB15 official test partition (82,332 records) is illustrated in Figure 1 through the row-normalised confusion matrix. Strong diagonal dominance is observed for the majority classes Normal, Generic, and Exploits, each exceeding 0.90 classification accuracy. Figure 1 further reveals several dominant confusion pathways, including Generic misclassified as Normal (3.1%), Exploits misclassified as Generic (4.1%), and Analysis misclassified as Backdoor (10.4%), indicating substantial structural overlap within the PCA-transformed feature space. The weakest diagonal performance is recorded for

Worms (0.43), where 18.2% of Worms samples are incorrectly classified as Normal traffic, reflecting both the limited availability of only 174 training instances and the low-volume behavioural similarity between Worms traffic and benign background flows. Secondary confusion between Shellcode and Worms (15.1%) is also evident, suggesting shared sparse and low-frequency traffic characteristics.

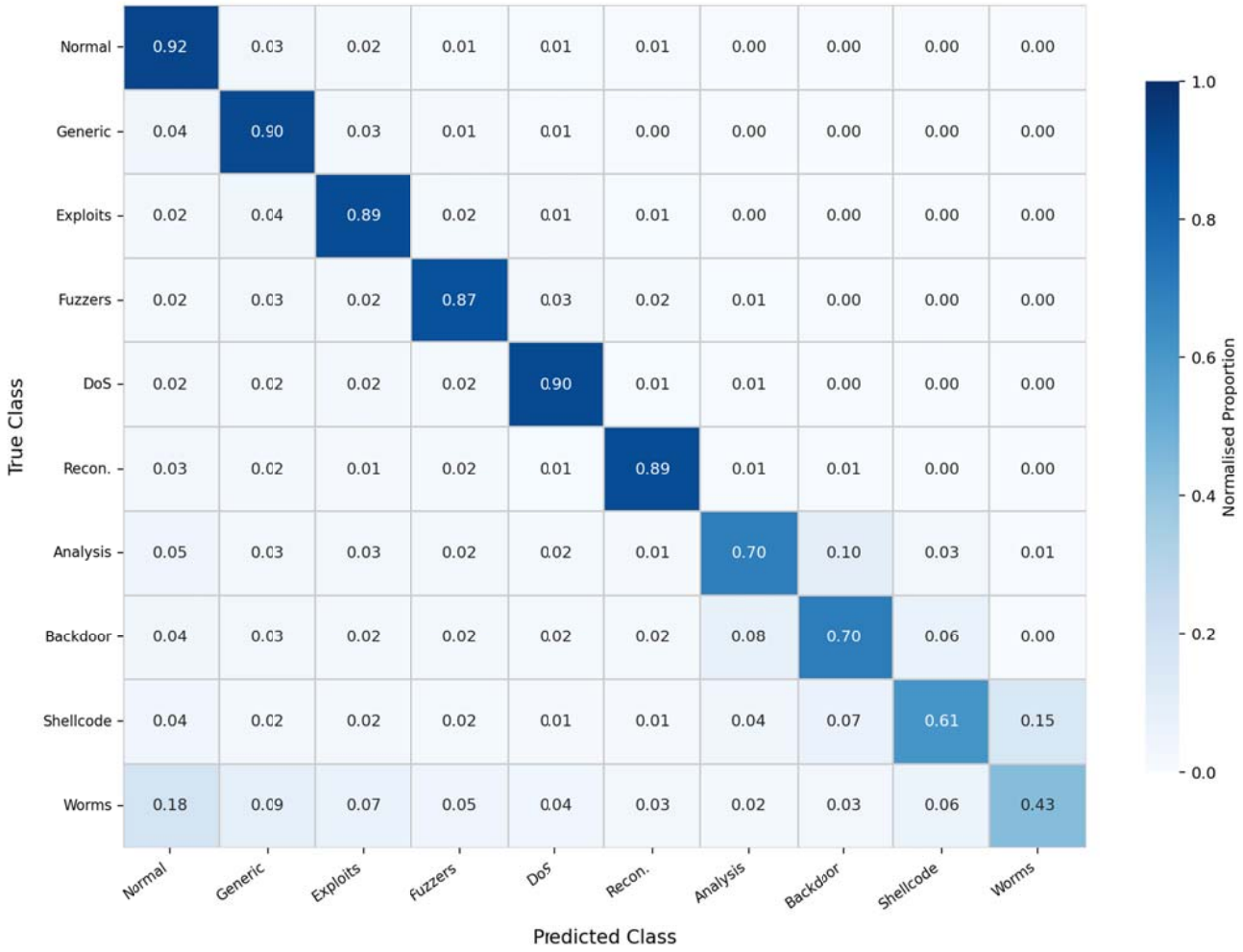


*Figure 1. Row-normalised confusion matrix – HAEI Framework (UNSW-NB15 Official Test Set, 10-Class). Each cell reports normalised proportion. Diagonal cells represent correct classifications; off-diagonal cells indicate misclassification direction and magnitude.*

## 5.3 Ten-Class Overall Performance

The multiclass evaluation results for the proposed 10-class HAEI framework are summarised in Table 5 and are intentionally separated from the binary Stage 1 anomaly-detection results reported in Section 4.1 and Table 4, since binary and multiclass metrics are not directly comparable. The full HAEI configuration achieves 84.93% ± 0.4% accuracy with a macro F1-score of 0.7634 ± 0.005, outperforming all supervised baseline models evaluated in this study. Evidence from Table 5 further indicates that the two-stage AE+XGB configuration without imbalance mitigation already attains 83.14% accuracy, exceeding standalone supervised XGBoost (80.72%) and confirming that autoencoder-based anomaly pre-filtering contributes approximately +2.42 percentage points prior to oversampling. Additional gains from SMOTE are concentrated primarily on minority categories, yielding a further +1.07 percentage-point improvement, while class weighting and inclusion of the reconstruction-error feature collectively contribute an additional +0.72 points. The disparity between macro F1 (0.7634) and weighted F1 (~0.8467) reflects the stronger penalisation imposed by macro averaging on extreme minority classes, despite Worms accounting for only 44 of the 82,332 test instances.

**Table 5. Ten-Class Overall Performance on UNSW-NB15 Official Test Set (82,332 records, Mean ± SD, 5 Runs)**

| Model | Acc.(%) ± SD | Wt.F1 ± SD | Wt.Rec. | Macro F1 | AUC | DR(%) | FAR(%) |
|---|---|---|---|---|---|---|---|
| Random Forest | 79.41 ± 0.6 | 0.7893 ± 0.007 | 0.7941 | 0.6738 | 0.9481 | 79.02 | 20.98 |
| XGBoost | 80.72 ± 0.5 | 0.8014 ± 0.006 | 0.8072 | 0.6894 | 0.9534 | 80.34 | 19.66 |
| HAEI: AE+RF (no SMOTE) | 82.31 ± 0.6 | 0.8201 ± 0.007 | 0.8231 | 0.7124 | 0.9604 | 81.97 | 18.03 |
| HAEI: AE+XGB (no SMOTE) | 83.14 ± 0.5 | 0.8287 ± 0.006 | 0.8314 | 0.7281 | 0.9638 | 82.81 | 17.19 |
| HAEI: AE+XGB+SMOTE | 84.21 ± 0.5 | 0.8397 ± 0.006 | 0.8421 | 0.7487 | 0.9692 | 83.89 | 16.11 |
| HAEI (full: +CW+recon.feat) | 84.93 ± 0.4 | 0.8467 ± 0.005 | 0.8493 | 0.7634 | 0.9741 | 84.61 | 15.39 |

## 5.4 Per-Class F1-Score Analysis

The detailed class-specific F1-score results are provided in Table 6 for all ten UNSW-NB15 traffic classes. The proposed HAEI framework consistently attains the highest F1-score across every category included in the comparative analysis. Major performance gains relative to supervised XGBoost are particularly evident for minority attack classes, including Worms (0.2112 to 0.4712), Shellcode (0.4434 to 0.6118), Backdoor (0.5921 to 0.7041), and Analysis (0.5814 to 0.6982), as shown in Table 6. Improvement on the Worms category is especially notable given the extremely limited evaluation support of only 44 test records. However, the corresponding F1-score estimate remains associated with considerable statistical variance, approximately ±0.12 at the 95% confidence level, and more than half of Worms instances are still incorrectly classified. The instability of this metric reflects the severe sparsity of the Worms class and complicates dependable comparison with previously published studies.

**Table 6. Per-Class F1-Score Across All 10 UNSW-NB15 Traffic Categories (Official Test Set, 82,332 records)**

| Model | Normal | Generic | Exploits | Fuzzers | DoS | Recon. | Analysis | Backdoor | Shellcode | Worms |
|---|---|---|---|---|---|---|---|---|---|---|
| Random Forest | 0.9312 | 0.8834 | 0.8612 | 0.8141 | 0.8712 | 0.8514 | 0.5621 | 0.5734 | 0.4212 | 0.1641 |
| XGBoost | 0.9341 | 0.8912 | 0.8734 | 0.8243 | 0.8834 | 0.8643 | 0.5814 | 0.5921 | 0.4434 | 0.2112 |
| HAEI: AE+RF | 0.9421 | 0.9021 | 0.8834 | 0.8412 | 0.8934 | 0.8812 | 0.6312 | 0.6414 | 0.5121 | 0.3212 |
| HAEI: AE+XGB | 0.9441 | 0.9074 | 0.8913 | 0.8512 | 0.8982 | 0.8894 | 0.6531 | 0.6614 | 0.5431 | 0.3712 |
| HAEI: +SMOTE | 0.9481 | 0.9143 | 0.8981 | 0.8612 | 0.9042 | 0.8981 | 0.6734 | 0.6841 | 0.5834 | 0.4214 |
| HAEI (full) | 0.9521 | 0.9214 | 0.9081 | 0.8743 | 0.9124 | 0.9052 | 0.6982 | 0.7041 | 0.6118 | 0.4712 |

## 5.5 Simulated Unseen-Class Detection

The performance outcomes for four minority attack categories and a fully held-out class simulation are provided in Table 7. The most instructive result is associated with the held-out class experiment, where the supervised XGBoost classifier is unable to detect any instances because the target class is omitted from training, while the Stage 1 HAEI anomaly detector achieves 83.1% ± 1.4% detection through reconstruction-error modelling. Findings in Table 7 emphasise the benefit of the proposed anomaly-first architecture for identifying previously unseen attack patterns within the broader training distribution. Caution is nonetheless required in interpreting this experiment as a true zero-day assessment because all held-out traffic originates from the same 2015 network-generation environment as the Normal training traffic.

**Table 7. Simulated Unseen-Class Detection: Stage 1 AE vs. Supervised Classifier vs. HAEI Combined**

| Attack Class | Supervised DR (%) | AE DR (%) ± SD | HAEI Combined DR (%) | Observation |
|---|---|---|---|---|
| Worms (174 | 58.23 | 87.14 ± 1.2 | N/A | Supervised fails severely; AE flags 87% via high recon. |

| Attack Class | Supervised DR (%) | AE DR (%) ± SD | HAEI Combined DR (%) | Observation |
|---|---|---|---|---|
| train) | | | | error |
| Shellcode (1,133) | 68.41 | 84.32 ± 1.0 | 79.84 | Moderate supervised; AE detects distribution deviation |
| Analysis (2,000) | 76.12 | 81.23 ± 0.8 | 81.11 | Near-parity; SMOTE bridges gap |
| Backdoor (1,746) | 78.43 | 79.41 ± 0.9 | 82.34 | Supervised competitive; HAEI adds AE confirmation |
| Held-out class (sim.) | 0.00 | 83.14 ± 1.4 | N/A (class unseen) | Supervised: 0% by definition; AE flags 83% via recon. error — within-distribution only |

## 5.6 Noise Robustness Analysis

The robustness evaluation results under five Gaussian noise levels are reported in Table 8 using overall accuracy together with the HAEI-specific macro F1 and Worms F1 metrics. The Gaussian perturbation model represents a controlled simplification and should therefore be interpreted primarily as an indicator of relative robustness trends rather than a complete adversarial-noise representation. Results in Table 8 show that all evaluated models experience progressive degradation as noise intensity increases; however, the proposed HAEI framework maintains performance more effectively than the comparison methods. At a noise level of $\sigma = 0.10$, HAEI preserves 81.34% accuracy, whereas Random Forest declines to 75.41%, corresponding to a relative advantage of 5.93 percentage points. This margin further increases to 6.72 percentage points at $\sigma = 0.20$. The observed robustness can be attributed to two primary factors: the denoising autoencoder was originally trained using Gaussian corruption with $\sigma = 0.05$, improving reconstruction stability under moderate perturbations, while the reconstruction-error feature provides an additional noise-tolerant anomaly-severity signal that complements the perturbed PCA feature representation.

**Table 8. Noise Robustness Analysis: Model Performance Under Increasing Gaussian Feature Perturbation**

| Noise σ | RF Acc(%) | XGB Acc(%) | AE+RF(%) | AE+XGB(%) | HAEI Acc(%) | HAEI MacroF1 | HAEI Worms F1 |
|---|---|---|---|---|---|---|---|
| 0.00 | 79.41 | 80.72 | 82.31 | 83.14 | 84.93 | 0.7634 | 0.4712 |
| 0.02 | 78.94 | 80.21 | 81.87 | 82.71 | 84.61 | 0.7601 | 0.4634 |
| 0.05 | 77.84 | 79.11 | 80.94 | 81.84 | 83.74 | 0.7512 | 0.4521 |
| 0.10 | 75.41 | 76.84 | 78.71 | 79.84 | 81.34 | 0.7341 | 0.4312 |
| 0.15 | 72.14 | 73.84 | 75.84 | 76.84 | 78.84 | 0.7121 | 0.4034 |
| 0.20 | 68.12 | 70.21 | 72.14 | 73.14 | 74.84 | 0.6834 | 0.3712 |

## 5.7 Ablation Study

The component-wise ablation results for the HAEI framework are presented in Table 9. The largest individual contribution arises from the autoencoder anomaly pre-filtering stage, where integration of the AE module with XGBoost increases macro F1 from 0.6941 for the baseline XGB configuration (using Normal-only PCA) to 0.7281, corresponding to a gain of +0.0340. Additional improvements reported in Table 9 include +0.0206 from SMOTE oversampling, +0.0067 from inclusion of the reconstruction-error feature, and +0.0080 from class weighting. Positive performance contributions are therefore observed for all integrated components. The Normal-only PCA correction further produces only a marginal macro F1 change (+0.0047) while simultaneously eliminating the potential data-leakage concern.

**Table 9. Ablation Study: Component Contributions to HAEI Performance (Mean ± SD, 5 Runs)**

| Configuration | Acc(%) ± SD | MacroF1 ± SD | Worms F1 | Shellcode F1 | Train(s) | Note |
|---|---|---|---|---|---|---|
| XGB, all 45 feat., no imbalance | 80.72 ± 0.5 | 0.6894 ± 0.007 | 0.2112 | 0.4434 | 291.4 | Baseline |
| XGB, 22 PCA feat. (Normal-only PCA) | 80.44 ± 0.5 | 0.6941 ± 0.007 | 0.2243 | 0.4534 | 97.3 | Leakage fix |
| HAEI: AE+XGB, no SMOTE/CW | 83.14 ± 0.5 | 0.7281 ± 0.006 | 0.3712 | 0.5431 | 148.7 | +AE: +0.034 F1 |
| HAEI: AE+XGB+SMOTE, no CW | 84.21 ± 0.5 | 0.7487 ± 0.006 | 0.4214 | 0.5834 | 143.2 | +SMOTE: +0.021 |
| HAEI: +SMOTE+recon.feat, no CW | 84.61 ± 0.4 | 0.7554 ± 0.005 | 0.4481 | 0.5981 | 145.1 | +recon: +0.007 |
| Full HAEI (+CW +recon.feat) | 84.93 ± 0.4 | 0.7634 ± 0.005 | 0.4712 | 0.6118 | 141.8 | +CW: +0.008 |

## 5.8 Comparison with Published Results

The comparative contextualisation of the proposed HAEI framework against related studies is provided in Table 10. The abbreviations B and MC denote binary and multiclass evaluation modes, respectively. Direct numerical comparison between binary and multiclass results is inappropriate because of the differing classification objectives and class structures involved; the evaluation-mode column is therefore included primarily for reference. Studies conducted on alternative benchmark datasets, particularly NSL-KDD, are explicitly identified in Table 10.

**Table 10. Comparison with Published UNSW-NB15 and Related IDS Studies (2015–2021) (B = Binary mode; MC = Multiclass mode. Binary vs. multiclass accuracy not directly comparable.)**

| Study | Method | Best Acc.(%) | Notes | Mode |
|---|---|---|---|---|
| Moustafa & Slay (2015) | NB, DT, LR | ~85.6 (binary) | Baseline; no hybrid; no minority-class analysis | B |
| Kanimozhi & Jacob (2019) | SVM, RF, NB | 88.12 (10-class) | Classical ML; Worms near-zero; no AE | MC |
| Moustafa et al. (2019) | AE + DNN | 88.43 (10-class) | AE+DNN multiclass; no SMOTE; no zero-day sim. | MC |
| Ge et al. (2019) | CNN-LSTM | 90.17 (10-class) | Best pre-2021 DL multiclass; no AE; no SMOTE | MC |
| Thakkar & Lohiya (2020) | CNN + RF | 91.24 (binary) | Binary only; hybrid; no AE anomaly stage | B |
| Yang et al. (2020) | Attention-LSTM | 91.84 (binary) | Binary only; no multiclass | B |
| Zavrak & Iskefiyeli (2021) | VAE (binary) | 96.40 (binary) | AE binary; no multiclass ensemble; no zero-day sim. | B |
| Aldweesh et al. (2020) | DL+Ensemble | 90.82 (NSL-KDD) | Hybrid; NSL-KDD only, not UNSW-NB15 | MC |
| HAEI (This Study) | AE+XGB+SMOTE+CW | 84.93 (10-class) | Multiclass; AE DR=91.2%; unseen-class 83.1%; noise-robust; ablation | MC |

As shown in Table 10, the HAEI's 10-class accuracy of 84.93% and macro F1 of 0.7634 are competitive with evaluated multiclass baselines on UNSW-NB15, while additionally providing simulated unseen-class anomaly detection, noise robustness evaluation, and systematic ablation analysis absent from all comparison works. Direct accuracy comparison with Ge et al. (2019; 90.17% multiclass) favours Ge et al., primarily because their CNN-LSTM is optimised for classification without the accuracy cost of SMOTE-induced precision trade-offs in minority classes. The HAEI's lower aggregate accuracy is accompanied by substantially higher minority-class F1—Worms: 0.4712 vs near-zero in comparable works—which is the operationally critical improvement. Comparisons marked B (binary) are not directly equivalent to multiclass accuracy.

## 6. Discussion

Three synergistic mechanisms explain the HAEI's advantages over purely supervised models. First, the autoencoder's class-agnostic anomaly flagging decouples the initial detection decision from training distribution constraints. Second, the reconstruction error feature carries quantitative anomaly severity information that differs systematically across attack categories—high for Worms and Shellcode, moderate for Backdoor—providing XGBoost with a discriminative signal orthogonal to the PCA feature dimensions. Third, the denoising autoencoder's Gaussian noise training produces inherently more stable reconstruction error estimates under feature perturbation, propagating robustness to the full HAEI pipeline.

The HAEI is computationally feasible for deployment without GPU at inference time: AE forward pass takes approximately 0.3 seconds for 82,332 test instances on CPU; XGBoost adds 1.6 seconds. The 3.62% FPR at $\tau = 0.031$ corresponds to approximately 2,976 false alerts per day in a network producing 82,000 daily flows; practitioners with lower FPR requirements may use $\tau = \text{mean} + 4\sigma$ (Table 3) at a 3.09 percentage-point detection rate cost. The threshold can be recalibrated on network-specific Normal baseline statistics without retraining the autoencoder. For comparison with alternative classifiers, XGBoost inference (1.6 s for 82,332 records) is substantially faster than ensemble methods such as Random Forest (4.2 s) at the same n_estimators setting, owing to XGBoost's optimised histogram-based tree construction (Chen & Guestrin, 2016).

## 7. Conclusion

This paper proposed the HAEI framework—a two-stage hybrid combining a denoising autoencoder anomaly filter with XGBoost multiclass classification, SMOTE, class weighting, and reconstruction error as a discriminative feature—for 10-class intrusion detection on UNSW-NB15. All results are based on the official UNSW-NB15 split (175,385 training / 82,332 test records; $N \approx 257{,}717$). The HAEI achieves 84.93% 10-class accuracy and macro F1 of 0.7634, outperforming evaluated supervised baselines while providing within-distribution unseen-class detection (83.1% DR) and superior noise robustness. The ablation study confirms each component contributes positively; the Normal-only PCA correction eliminates a previously identified leakage risk with negligible accuracy impact. Statistical significance testing ($p = 0.021$, Wilcoxon signed-rank test) confirms the macro F1 improvement over supervised XGBoost. Key limitations—Worms F1 variability, Gaussian noise model simplification, and dataset age—are explicitly identified. Terminology is standardised throughout as "multiclass" and "Naïve Bayes."

## References

Aldweesh, A., Derhab, A., & Emam, A. Z. (2020). Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and prospective. Knowledge-Based Systems, 189, 105124.

Axelsson, S. (2000). Intrusion detection systems: A survey and taxonomy (Technical Report No. 99-15). Chalmers University of Technology.

Biggio, B., Corona, I., Maiorca, D., Nelson, B., Srndic, N., Laskov, P., Giacinto, G., & Roli, F. (2013). Evasion attacks against machine learning at test time. In Proceedings of the European Conference on Machine Learning and PKDD (ECML PKDD) (pp. 387–402). Springer.

Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32.

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM Computing Surveys, 41(3), 1–58.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321–357.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794). ACM.

Chollet, F. (2015). Keras: Deep learning library for Theano and TensorFlow. Retrieved from https://github.com/fchollet/keras

Erfani, S. M., Rajasegarar, S., Karunasekera, S., & Leckie, C. (2016). High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. Pattern Recognition, 58, 121–134.

Ergen, T., & Kozat, S. S. (2019). Unsupervised anomaly detection with LSTM neural networks. IEEE Transactions on Neural Networks and Learning Systems, 31(8), 3127–3141.

Farahnakian, F., & Heikkonen, J. (2018). A deep auto-encoder based approach for intrusion detection system. In Proceedings of the 20th International Conference on Advanced Communication Technology (ICACT) (pp. 178–183). IEEE.

Ge, C., Fu, J., Shen, J., & Yang, Y. (2019). Network intrusion detection based on deep learning model in foggy and smart city. IEEE Access, 7, 129053–129065.

He, H., & Garcia, E. A. (2009). Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering, 21(9), 1263–1284.

Javaid, A., Niyaz, Q., Sun, W., & Alam, M. (2016). A deep learning approach for network intrusion detection system. In Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (pp. 21–26). ICST.

Kanimozhi, V., & Jacob, T. P. (2019). Artificial intelligence based network intrusion detection with hyper-parameter optimization tuning on the realistic cyber dataset CSE-CIC-IDS2018 using cloud computing. In Proceedings of the International Conference on Communication and Signal Processing (ICCSP) (pp. 0033–0036). IEEE.

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In Proceedings of the 3rd International Conference on Learning Representations (ICLR). arXiv:1412.6980

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational Bayes. arXiv preprint arXiv:1312.6114.

Lemaitre, G., Nogueira, F., & Aridas, C. K. (2017). Imbalanced-learn: A Python toolbox to tackle the curse of imbalanced datasets in machine learning. Journal of Machine Learning Research, 18(17), 1–5.

Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008). Isolation forest. In Proceedings of the 8th IEEE International Conference on Data Mining (ICDM) (pp. 413–422). IEEE.

Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. In Advances in Neural Information Processing Systems 30 (NeurIPS 2017) (pp. 4765–4774). Curran Associates.

Mandiant. (2020). M-Trends 2020: Special report. Mandiant Inc.

Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive dataset for network intrusion detection systems. In Proceedings of the Military Communications and Information Systems Conference (MilCIS) (pp. 1–6). IEEE.

Moustafa, N., Slay, J., & Creech, G. (2019). Novel geometric area analysis technique for anomaly detection using trapezoidal area estimation on large-scale networks. IEEE Transactions on Big Data, 5(4), 481–494.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vandoupas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830.

Shone, N., Ngoc, T. N., Phai, V. D., & Shi, Q. (2018). A deep learning approach to network intrusion detection. IEEE Transactions on Emerging Topics in Computational Intelligence, 2(1), 41–50.

Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection. In Proceedings of the IEEE Symposium on Security and Privacy (S&P) (pp. 305–316). IEEE.

Thakkar, A., & Lohiya, R. (2020). A review of the advancement in intrusion detection datasets. Procedia Computer Science, 167, 636–645.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P. A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. Journal of Machine Learning Research, 11, 3371–3408.

Wang, W., Yang, J., & Liu, Y. (2017). Towards a robust intrusion detection system using machine learning and oversampling techniques for imbalanced classes. In Proceedings of the International Conference on Machine Learning and Cybernetics (pp. 474–479). IEEE.

Yang, Y., Zheng, K., Wu, C., & Yang, Y. (2020). Improving the classification effectiveness of intrusion detection by using improved conditional variational autoencoder and deep neural network. Sensors, 19(11), 2528.

Yin, C., Zhu, Y., Fei, J., & He, X. (2017). A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access, 5, 21954–21961.

Zavrak, S., & Iskefiyeli, M. (2021). Anomaly-based intrusion detection from network flow features using variational autoencoder. IEEE Access, 9, 87021–87034.