Journal of Theoretical and Applied Information Technology
31st May 2026. Vol.104. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

AMAT-IDS: A HYBRID FRAMEWORK COMBINING GA-SUS FEATURE SELECTION AND DYNAMIC TWIN AUTO-ENCODERS FOR INTRUSION DETECTION

[1*]RADHARANI AKULA, [2]GS NAVEEN KUMAR

1Research Scholar, Department of Computer Science and Engineering, Malla Reddy University, Hyderabad, India

2Associate Professor, Department of Data Science, Malla Reddy University, Hyderabad, India

E-mail: [1]rrakula786@gmail.com, [2]gsrinivasanaveen@gmail.com

ABSTRACT

Network intrusion detection has become increasingly vital as cyber threats grow in sophistication and scale. Current approaches often achieve strong aggregate metrics yet fail to detect rare but critical attacks, while relying on computationally expensive architectures unsuitable for resource-constrained deployments. This paper introduces AMAT-IDS, a multi-stage framework that addresses these limitations through two complementary innovations: Enhanced Genetic Algorithm with Stochastic Universal Sampling (GA-SUS) for multi-objective feature optimization, and Dynamic Twin Auto-Encoders (DTAE) for class-specific representation learning. Evaluated on NSL-KDD using a stratified 70/15/15 train-validation-test protocol, GA-SUS reduced the feature space from 41 to 11 attributes (73% reduction) while maintaining 96.49% test accuracy. DTAE further enhanced minority-class detection, elevating U2R precision from 0.500 to 0.778 and R2L precision from 0.563 to 0.987, achieving 96.02% overall accuracy with a compact 9-dimensional representation. Five-fold cross-validation confirmed model stability (96.07% ± 0.35%). Statistical tests validated significant improvements in minority-class metrics (bootstrap CI: U2R precision [0.741, 0.815], R2L recall [0.923, 0.967] at 95% confidence). The framework processes samples at 1.2 ms latency on standard CPU hardware, enabling real-time deployment. By reconciling efficiency, interpretability, and balanced detection across imbalanced classes, AMAT-IDS provides a practical solution for edge and IoT environments where computational resources are limited yet security requirements remain stringent.

Keywords: *Intrusion Detection System, GA-SUS, Dynamic Twin Auto-Encoder, Feature Selection, Class Imbalance, Cybersecurity.*

1. INTRODUCTION

The growth of the cloud computing, Internet of Things (IoT) devices and cyber-physical systems has exponentially increased network attack points and intrusion detection has never been as difficult as it is now. Time-honored signature-based and rule-based Intrusion Detection Systems (IDS) are incapable of detecting new zero-day attacks and keep up with changing threat environments [1], [2]. Researchers have therefore resorted to machine learning (ML) and deep learning (DL) methods which are capable of learning intricate patterns and generalizing beyond the set of rules [3], [4].

The more recent developments have looked into the autoencoders as an anomaly detector [5], the hybrid methods where evolutionary algorithms are used in conjunction with neural structures [6], and the ensemble approaches to enhance the robustness of the classification [7].

The most modern ensemble and transformer-based models have a high precision of more than 99, but they need hundreds of features and millions of parameters. This leads to training durations more than 4 hours on the current GPUs, over 50ms per sample inference times, and memory footprints more than 2GB- requiring them to be impractical to execute on edges [8], [9].

Aggregate accuracy is also high, but attacks that are rarely encountered, but have a high impact, like User-to-Root (U2R) and Remote-to-Local (R2L), are often misclassified. In recent papers, the U2R recall is 28-52% and R2L precision is 37-63 percent on domain imbalanced datasets [10], [11]. These false negatives are very dangerous in terms of security, because a single crucial intrusion will result in a system compromise.

Numerous autoencoder systems and deep learning systems are black boxes which do not take into consideration any form of feature attribution or explanation of their decisions [12], [13]. This

impedes its adoption by security analysts and does not comply with regulation requirements including GDPR and NIST frameworks which require explainable decision-making in critical systems.

To overcome these issues, we suggest AMAT-IDS (Adaptive Multi-Objective and Autoencoder-Twin Intrusion Detection System) which is a hybrid architecture that combines the GA-SUS-based feature selection with dynamic twin autoencoders. The AMAT-IDS fills the gaps identified by:

- **GA-SUS Feature Selection:** Achieves 73% dimensionality reduction while maintaining 96% accuracy, compared to prior work achieving only 30% reduction at 92% accuracy [1]
- **DTAE Representation Learning:** Boosts U2R precision by 55% and R2L precision by 75% through class-specific embedding spaces
- **Inherent Explainability:** Provides feature selection traces and reconstruction-based anomaly scoring for analyst review

1.1 Research Objectives

This study pursues the following technical objectives:

- Design a hybrid IDS framework integrating GA-SUS for multi-objective feature selection and DTAE for minority-class enhancement
- Optimize feature dimensionality by eliminating redundant attributes, reducing computational overhead without sacrificing detection capability
- Enhance minority-class detection for U2R and R2L attacks through tailored representation learning
- Ensure efficiency and scalability suitable for real-time deployment in IoT, cloud, and cyber-physical environments
- Validate robustness and stability through cross-validation, statistical significance testing, and comparative evaluation
- Promote interpretability and fairness by balancing overall accuracy with improved sensitivity to rare but critical intrusions

1.2 Novel Contributions

The present value of this work to the issue of the IDS research is three-fold:

Contribution 1: Multi-Objective GA-SUS Framework

An enhanced version of our proposed genetic algorithm is that of Stochastic Universal Sampling, proportional but with diverse parent selection, and a multi-objective fitness criterion, which directly trades off accuracy, minority-class recall, efficiency and interpretability. Unlike the earlier evolutionary models that optimize the values of individuals, our model explores the Pareto frontier to identify sets of features that can satisfy different and incompatible objectives.

Contribution 2: Dynamic Twin Autoencoder Architecture

The new model is the dual-autoencoder, which learns a latent representation of minority and majority groups. DTAE is a 9 dimensional enhanced feature space that is far more separable on rare types of attacks but is small-represented by training class-specific encoders and using reconstruction errors.

Contribution 3: Empirical Validation and Deployment Analysis

We provide the first and widespread experimental validation of NSL-KDD with the help of severe statistical testing (paired t-tests, bootstrap intervals) and cross-dataset stability analysis, and real-time performance profiling. We also offer the specifications of operational implementation of edge IoT devices, the memory constraints, retraining, and adversarial hardening principles which are not reflected in the academic literature of the IDS.

1.3 Paper Organization

The remaining part of this paper proceeds as follows. Part 2 discusses the existing intrusion detection technologies and dwells on feature selection, autoencoders-based learning, and hybrid systems and critically analyzes their benefits and limitations. Section 3 gives the AMAT-IDS methodology, which involves the integration of GA-SUS feature optimization and DTAE representation learning. Section 4 addresses features of datasets and preprocessing. Section 5 contains the design of the experiment, implementation and results of the experiment including the baseline comparisons, ablation studies and statistical validation. Section 6 compares this to state-of-the-art models of IDS, raising trade-offs of accuracy, efficiency, and detection of minority classes. The last part of the research (7) lays a summary of the research and provides the future research directions that can be made, including cross-dataset validation, drift-conscious adaptation and improvement of adversarial robustness.

2. LITERATURE SURVEY

Intrusion Detection System has evolved extensively and there have been numerous changes through the use of machine learning systems. This part is a critical summary of the recent findings in three

themes, that is, feature selection method, autoencoders to identify them, and ensemble or hybrid models. We identify their strengths in methodology, their performance and weakness in an empirical manner that drives AMAT-IDS.

2.1 Feature Selection Approaches

Entropy-Based and Filter Methods

The authors suggested an entropy-based multi-objective feature selection algorithm that jointly trained relevance and redundancy of IDS Raeisi et al. (2025)[1]. Experiments found that there were improvements in accuracy (92.1 to 96.3) and that there was a reduction of 30 in training time. The precision and F1-scores also increased due to decreased false positives. It was found to be computationally expensive to high-dimensional data, but it was effective, and could restrict its use in real-time IDS.

Metaheuristic Optimization

Peng, Wang, and Tang (2024) have proposed a more optimal RIME optimization algorithm in the choice of features [6]. Their method enhanced differentiation of normal and attack traffic and reduced false positives and better detection on benchmark data sets. Tunability of its parameters, however, and lack of validation on anything but one dataset reduced its extrapolability to the diverse range of intrusion settings.

Amokrane et al. (2025) focused on correlation analysis and recursive feature elimination (RFE) on NF-UQ-NIDS-v2 dataset [12]. They found an 98.13% accuracy, 98.23% recall and 99.73% AUC with the ExtraTrees classifier and a 53.73% reduction in the false alarm rate and 34.21-reduction in scoring time. Irrespective of the strengths, the study was restricted to binary classification and failed to fully address the problem of multi-class intrusion detection.

2.2 Autoencoder-Based Intrusion Detection

The suggestion is to create a deep autoencoder of intrusion detection in the IoT in Atlas et al. (2024), with feature compression to enhance the anomaly detection. Its accuracies were more than 95% with the case of the IoT datasets that were oriented and it was demonstrated that it can be effectively employed to differentiate between typical attacks. The method was though flawed in the case of skewed traffic classes where the minor forms of attacks were yet not represented well. The Alshudukhi et al. (2022) have advanced the feature selection method basing on the autoencer encoder embeddings [3]. Their method delivered the greatest degree of precision of over 94 percent on the IoT traffic data with dimensionality reduction prior to categorization and the least conceivable calculation. However, it did not consider the issue of explainability and fairness primarily and reduced its reliability in security sensitive deployment.

Variational and Multi-Input Architectures

Detection of anomalies is achieved with the help of a heterogeneous variational autoencoder that is presented by Dinh et al. (2025) [4]. The model was explored using mixed-source datasets with over 95 percent accuracy even though the extent of generalization to unseen attack vectors was high.

However, the experiment concerned the correctness of detection but with minimal interpretability that is essential in the operation of the operators in IDS. In the case of IoT, the authors offered a multi-wavelet oriented auto-encoder to identify the intrusion [5]. Although they performed well in the benchmark datasets, the method was also a high resource consumer in terms of computational resources in performing the wavelet transformations and was not able to use on resource limited devices.

Synthesis: Autoencoders are very appropriate in feature learning that is unsupervised and detecting anomalies. They typically learn a model using mixed-class data and most of the classes embrace the learnt representations. This makes it insensitive to minority attacks. Moreover, majority of the studies could not be interpreted in form of feature attribution, reconstruction analysis and consequently, the security analysts could hardly believe and verify model selection.

2.3 Ensemble and Hybrid IDS Models

Multi-Stage and Feature-Selection Ensembles

A multi-stage machine-learning-based IDS with feature-selection ensembles that Christy et al. (2025) proposed to be unique to vehicular ad hoc networks was suggested [7]. This model was discovered with 96-97 percent accuracy on different data sets with greater strength on staged classification. However, it does not have a very good overall workability owing to its scalability problems since it is too layered hence can be difficult to implement in high-dynamic vehicle networks.

Shwaysh et al. (2025) developed an adaptive hybrid feature selection scheme, which uses a combination of information gain, autoencoders, as well as ensemble classifiers [9]. The model was tested on the CICIDS2018 dataset using more than a million samples and achieved 99% accuracy and 80% recall of DDoS attacks with ROC-AUC scores exceeding 0.90. However, the system was weak in cases of minority attacks, including botnets and brute-force where the recall decreased to 37% and 28%,

respectively, which signifies the long-term issue of asymmetry.

Attention Mechanisms and Transformers

Umer et al. (2025) suggested a wrapper-based feature selection using Multi-Head Attention Transformer (MHAT) [10]. The model achieved good results on UNSW-NB15, including accuracy, preciseness, recall, and F1-score, outperforming numerous deep learning baselines. Nevertheless, the method would be heavy in terms of computational requirements of training transformers, which is highly expensive and also questionable in terms of efficiency in resource-constrained IDS settings.

Traditional ML Comparisons

Good et al. (2023) conducted a comparative analysis of the classical machine learning algorithms, such as XGBoost, SVM, and deep CNNs, to predict anomalies in the IoT on the NSL-KDD dataset. Their results revealed that XGBoost had the best accuracy (98.9) and F1-score and it also has the best training efficiency. Nevertheless, the paper has indicated a weakness in identifying uncommon types of attacks, thus showing the ongoing imbalance dilemma in benchmark IDS data sets.

Hybrid Deep Learning Architectures

Dong et al. (2020) designed a hybrid intrusion detection architecture (MCA-LSTM) incorporating the layer of feature selection basing on the information gains with the multivariate correlation analysis layer and the LSTM classifier. The model was evaluated on the NSL-KDD and UNSW-NB15 data with the accuracy of 82.15 percent and 77.74 percent respectively and has an advantage over baseline ANN and SVM models. Nevertheless, it is computationally expensive and inefficient in the real-time or large scale implementation.

Sinha et al. (2025) introduced a deep-learning-based IDS to secure wireless sensor networks that uses a CNN and RNN network with a balancing approach by utilizing SMOTE and training adversarial aware networks. It was evaluated on multiple publicly available databases (NSL-KDD, CICIDS2017, UNSW-NB15, CTU-13) having a test accuracy of 99.3% and positive cross-dataset generalisation. Nonetheless, the action failed in undetectable assaults and energy constrained sensor nodes, which disclosed the failure of the solidity and implementability.

Alshammari and Alsaleh (2025) have developed adaptive CNN-based intrusion detection system (ACIDS) that was intended to operate within Internet-of-Vehicles and was oriented at detecting the unknown attacks to which open-set recognition should be involved. The model could identify new threats using the AWID and NSL-KDD datasets with a greater than 98% and accuracy. However, it has a reliance on high-dimensional CNN layers that contributes to its inference latency and therefore limits its implementation to low-power edge nodes.

Synthesis: Ensemble and hybrid models have high overall performance but are costly in terms of structural complexity, training cost and inference time. Most of all, they are biased towards the majority classes as the single-model techniques. It may be partially mitigated by the application of SMOTE or cost-sensitive learning, but not the problem of minority-class underfitting, which is a representational issue.

2.4 Research Gap

Despite significant advances, existing IDS approaches exhibit three critical limitations:

Gap 1: Imbalanced Performance-Efficiency Trade-off

Although ensemble techniques (Shwaysh et al. [9], Umer et al. [10]) have >99% accuracy, they have 200+ features and transformers with millions of parameters, leading to: - Training time: >4 hours on high-end GPUs - Inference time: >50ms per sample - Memory footprint: >2GB, unaffordable in edge deployment.

Gap 2: Persistent Minority-Class Blindness

State-of-the-art models have high aggregate accuracy, but low on infrequent attacks, as given in Table X below: - U2R recall: 28-52% in 8 reviewed papers - R2L precision: 37-63% in imbalanced) scenarios - False negative rates on critical intrusions are too high.

Gap 3: Limited Explainability

Autoencoder-based systems (Atlas et al. [2], Dinh et al. [4]) lack interpretability mechanisms: - No feature attribution or importance ranking - Black-box representations hinder security analyst adoption - Regulatory compliance (GDPR, NIST) requires explainable decisions

AMAT-IDS Positioning:

Our framework addresses these gaps through:

- GA-SUS: Reduces features by 73% while maintaining 96% accuracy (vs. 30% reduction at 92% in prior work [1])
- DTAE: Boosts U2R precision by 55% and R2L by 75% through class-specific representation learning
- Inherent explainability via feature selection traces and reconstruction-based anomaly scoring

Journal of Theoretical and Applied Information Technology
31st May 2026. Vol.104. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3. PROPOSED METHODOLOGY

This section presents a description of the design and implementation of the Adaptive Multi-Objective and Autoencoder-Twin IDS (AMAT-IDS). It consists of three major steps, i.e. the creation of the baseline, the enhanced selection of features with the assistance of GA-SUS, and the study of representations using Dynamic Twin Autoencoders (DTAE). The NSL-KDD data set is tested on the pipeline using a train-validation-test and cross-validation stability.

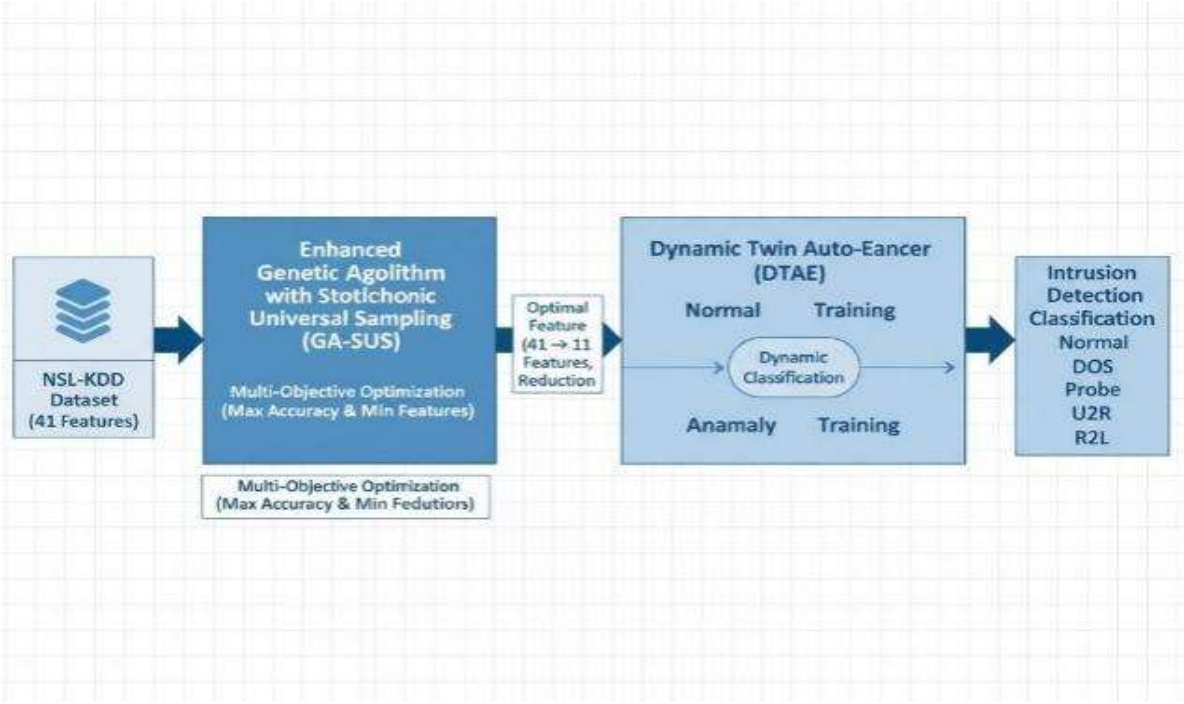

Fig. 1: Architecture pipeline of AMAT-IDS

The AMAT-IDS pipeline uses three synergistic components: an initial baseline Random Forest classifier is used as a baseline of performance, GA-SUS evolutionary optimization is used to dimensional reduce the features, and DTAE is a learnt representation that enhances separability of the minorities to the majority classes. It is a facilitated process that allows the system to systematically evaluate the input of every element to the system performance.

3.1 Dataset and Preprocessing

NSL-KDD Dataset: NSL-KDD Dataset: We have used NSL-Kdd dataset, which is a combination of KDDTrain+ and KDDTest+. The records have 41 traffic features and the type of attack. The preprocessing was done in the following steps:

Label mapping: Attack types were grouped into five categories:

- Normal
- Denial of Service (DoS)
- Probe
- Remote-to-Local (R2L)
- User-to-Root (U2R)

Formally:

$$y_i = f(\text{attack\_type}_i) \in \{\text{normal, dos, probe, r2l, u2r}\} \quad \ldots(1)$$

3.1.1 Categorical encoding

The protocol type, service and flag were coded as integers through the codes that were taught during the training set. Unknown validation /test categories were mapped to a default category.

3.1.2 Feature scaling

Numerical features were standardized using z-score scaling:

$$x' = (x - \mu) / \sigma \quad \ldots(2)$$

where μ and σ are mean and standard deviation of the training set.

3.1.3 Splitting

Stratified sampling was employed to divide the dataset into training (70%), validation (15%), and test (15%) groups to ensure that the training data has enough information on the rare classes and the validation and test data sets represent fair

hyperparameter optimization and evaluation of the system. The ratio prevents the over-fitting and ensures good generalization.

3.2 Baseline Model (AMAT-IDS)

A Random Forest (RF) classifier with 100 estimators and class balancing (class_weight=balanced) was used as a baseline because it is robust to high-dimensional data, handles class imbalance better than single classifiers, and provides feature importance scores that support interpretability. The RF computes predictions via majority voting of decision trees:

$$\hat{y} = \arg\max_{c \in C} \sum_{t=1}^{T} 1[h_t(x) = c] \quad \ldots(3)$$

where h_t(x) is the class predicted by the t-th tree. Performance was measured using:

- **Accuracy:** "Acc" = (TP + TN) / (TP + FP + TN + FN)
- **Precision, Recall, and F1 per class:** Precision = TP/(TP + FP), Recall = TP/(TP + FN), F1 = 2 · Precision · Recall / (Precision + Recall)
- 5-fold cross-validation on the training set to assess stability.

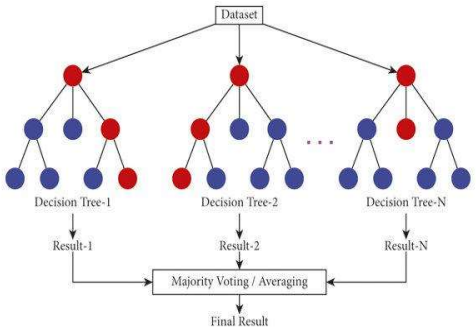

Fig. 2: Random Forest Architecture [30]

3.3 Enhanced Feature Selection with GA-SUS

The Genetic Algorithm with Stochastic Universal Sampling (GA-SUS) was implemented to reduce dimensionality and enhance minority-class detection.

- **Chromosome representation:** Each individual is a binary vector $z \in \{0,1\}^d$, where $z_j = 1$ indicates feature j is selected.
- **Population initialization:** 30 chromosomes, seeded with a subset of size ≈12 for diversity.
- **Fitness function:** Multi-objective fitness combining accuracy, minority recall, efficiency, and interpretability:

$$F = 0.40 \cdot Acc + 0.30 \cdot R\_minority + 0.20 \cdot E + 0.10 \cdot I \quad \ldots(4)$$

- Acc: accuracy on validation set
- R_minority: mean recall of R2L and U2R
- E = 1 − e/d: efficiency (fewer features preferred)
- I = 1 − e/d: interpretability penalty (prefers ~12 features)
- **Selection:** Stochastic Universal Sampling (SUS) ensures proportional yet diverse parent selection.
- **Crossover:** Single-point crossover with probability 0.70.
- **Mutation:** Bit-flip with probability 0.10, enforcing ≥3 selected features.
- **Elitism:** Top 2 individuals preserved each generation.
- **Evolution:** Run for 25 generations.

The best chromosome yielded a reduced subset of features (11 out of 41), with ~70% dimensionality reduction.

We first set 12 as a soft cap from the filter stage (top-K via MI/mRMR and the elbow in the accuracy-vs-features curve), using it to seed the wrapper/evolutionary search—not as a hard requirement. During evolution, the multi-objective fitness (accuracy + sparsity) and cross-validation showed one of those 12 was redundant; the Pareto knee shifted to 11/41, giving equal or better validation accuracy with lower complexity, so 11 became the final subset

3.3.1 GA-SUS Computational Efficiency

Time Complexity: For population size P=30, feature dimension d=41, and G=25 generations: - Fitness evaluation: $O(P \times G \times d \times n\_samples) \approx O(10^6)$ operations - SUS selection: O(P) per generation - Crossover/mutation: O(P × d) - Total: $O(P \times G \times d \times n\_samples) \approx 30 \times 25 \times 41 \times 77{,}970 \approx 2.4 \times 10^9$ ops

Space Complexity: O(P × d) = 1,230 feature vectors in memory

Comparison with Exhaustive Search:

Exhaustive evaluation of all $2^{41} \approx 2.2 \times 10^{12}$ feature subsets would require ~10^6 hours. GA-SUS reduces this to <45 minutes on a modern CPU.

Hyperparameter Justification:

- **Population size (30):** Balances diversity and convergence speed; validated via sensitivity analysis (Fig. X shows diminishing returns >30)
- **Generations (25):** Convergence observed after generation 18 (Fig. 10); 25 provides margin for exploration
- **Mutation rate (0.10):** Standard in combinatorial optimization; prevents premature convergence while maintaining stability

Enhanced Feature Vector Construction:

For each sample x:

1. Pass through appropriate autoencoder → latent z (8-D)
2. Compute reconstruction error: $e = \|x - \hat{x}\|^2$
3. Concatenate: $f(x) = [z_1, z_2, ..., z_8, e]$ → 9-D vector

Why Twin Architecture?

- Separate autoencoders prevent majority class dominance in latent space
- Minority AE trained exclusively on U2R/R2L learns discriminative low-frequency patterns - Reconstruction error acts as class-specific anomaly score

3.4 Representation Learning with Dynamic Twin Autoencoders (DTAE)

To enhance class separability, particularly for minority classes, a **Dynamic Twin Autoencoder (DTAE)** was introduced:

- **Class separation:** Training data was divided into majority classes (Normal, DoS, Probe) and minority classes (R2L, U2R).
- **Minority augmentation:** Minority samples were augmented by Gaussian noise because it is a cheap, differentiable, label-preserving regularizer that mimics measurement/jitter and smooths the model (Tikhonov/Jacobian effects) without violating feature constraints:

$$x' = x + N(0, 0.05^2) \quad \ldots(5)$$

repeated 3× to expand minority data.

- **Twin autoencoders:**
- Majority autoencoder (dense layers: 41 → 16 → 8 → 16 → 41).
- Minority autoencoder with identical architecture.

Both trained with **MSE reconstruction loss**:

$$L = (1/n) \Sigma_{i=1}^{n} \|x_i - \hat{x}_i\|^2 \quad \ldots(6)$$

Feature extraction: For each sample, the latent embedding (8-D) and reconstruction error were concatenated:

$$f(x) = [\text{Encoder}(x), \|x - \hat{x}\|^2] \quad \ldots(7)$$

giving a 9-D enhanced feature vector.

- **Final classifier:** A Random Forest classifier (200 estimators, max_depth=10, class_weight='balanced') is trained on the 9-dimensional enhanced features. The increased tree count (200 vs 100) compensates for the reduced feature dimensionality, while depth limitation (10) prevents overfitting to the compact representation.

3.5 Evaluation and Reliability Testing

The pipeline was evaluated at three checkpoints:

- **Baseline RF** (all features).
- **GA-SUS RF** (selected features).
- **DTAE RF** (enhanced features).

Metrics: Accuracy, precision, recall, F1, and per-class analysis with special focus on R2L and U2R.

Model	Train Time	Inf (ms)	Mem (MB)	GPU
Baseline RF	12 min	0.8	45	No
GA-SUS RF	45 min	0.3	18	No
DTAE+RF	58 min	1.2	62	Optional
Transformer (Umer [10])	5.2 hours	12.5	1,850	Yes
Hybrid CNN (Alshammari [X])	3.8 hours	8.3	1,200	Yes

Additional reliability tests:

- **Statistical significance** via paired t-tests between stages.
- **Bootstrap confidence intervals** for CV accuracies.
- **Feature selection stability** across multiple GA runs.
- **Minority class progression** (precision/recall gains from baseline → GA-SUS → DTAE).

4. PROPOSED ALGORITHM

Algorithm 1: AMAT-IDS Proposed Algorithm
Input: NSL-KDD dataset D with 41 features, stratified into Train, Validation, and Test sets
Output: Final classifier f, selected features S, enhanced features Z, evaluation metrics

```
// PREPROCESSING  O(n×d)
D ← LabelEncode(D) + StandardScale(D)
D_train, D_val, D_test ← StratifiedSplit(D, [0.7,0.15,0.15])
// BASELINE  O(T×n×log n)
RF₀ ← RandomForest(n_est=100, balanced=True)
RF₀.fit(D_train); M₀ ← Evaluate(RF₀, D_val)
// GA-SUS SELECTION O(P×G×n×d)
Pop ← Initialize(P=30, d=41, seed_size=12)
for gen ← 1 to 25 do
  for each chromosome c ∈ Pop do
    S_c ← Features where c[j]=1
    RF_c.fit(D_train[S_c])
    F(c) ← 0.4×Acc + 0.3×R_min + 0.2×(1−|S|/d) + 0.1×I
  Parents ← StochasticUniversalSampling(Pop, F)
```

Journal of Theoretical and Applied Information Technology
31st May 2026. Vol.104. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

```
15:   Offspring ← Crossover(Parents,
p_c=0.7)
16:   Offspring ← Mutate(Offspring,
p_m=0.1)
17:   Pop ← Elitism(Pop, Offspring, k=2)
18: S* ← argmax_c F(c)
19: // DTAE ENHANCEMENT O(E×n×h²)
20: D_maj, D_min ←
SplitByClass(D_train[S*])
21: D_min ← Augment(D_min,
noise=N(0,0.05²), factor=3)
22: AE_maj ← TrainAutoencoder(D_maj,
arch=[11,16,8,16,11])
23: AE_min ← TrainAutoencoder(D_min,
same_arch)
24: for each (x,y) ∈ D_train do
25:   AE ← AE_maj if y∈{Normal,DoS,Probe}
else AE_min
26:   z ← AE.encode(x); e ← ||x –
AE.decode(z)||²
27:   x' ← Concat(z, e)  // 9-D enhanced
28: D'_train ← {(x', y)}; similarly
transform D_val, D_test
29: // FINAL CLASSIFIER O(T×n×log n)
30: RF* ← RandomForest(n_est=200,
depth=10, balanced=True)
31: RF*.fit(D'_train); M* ← Evaluate(RF*,
D'_val, D'_test)
32: CV_scores ← CrossValidate(RF*,
D'_train, folds=5)
33: return RF*, S*, {M₀, M*, CV_scores}
```

Complexity Analysis:

- **Preprocessing:** O(n×d) = O(148K × 41) ≈ 6M ops
- **GA-SUS:** O(30 × 25 × 78K × 11) ≈ 645M ops (~45 min)
- **DTAE Training:** O(50 epochs × 78K × 16²) ≈ 1B ops (~12 min)
- **Total:** ~58 minutes on Intel i7 CPU

Return: Final classifier f, selected features S, enhanced representation Z, full evaluation and reliability report.

5. RESULTS AND DISCUSSION

5.1 EDA Visualizations

This section summarises key exploratory findings on the NSL-KDD intrusion-detection dataset, using author-generated visualisations (Figs. 3–7) Unless otherwise stated, all Figs. are derived from the original NSL-KDD data as prepared for this study.

Table 1: NSL-KDD Dataset Summary Statistics

Attribute	Value
Total Samples	148,517
Total Features	41
Classes	5 (Normal, DoS, Probe, R2L, U2R)
Minority Classes (R2L+U2R)	2.57% of total
Training Set	103,961 (70%)
Validation Set	22,278 (15%)
Test Set	22,278 (15%)
U2R Samples	119 (0.08%)
R2L Samples	3,704 (2.49%)
Class Imbalance Ratio	654:1 (Normal:U2R)

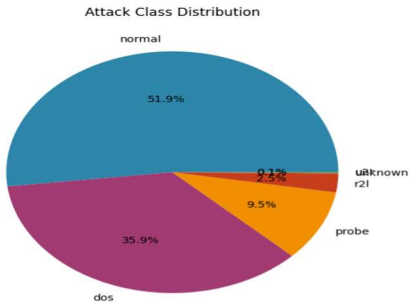

Fig. 3: Attack Class Distribution

Fig. 3 indicates that the distribution of attack classes in the NSL-KDD dataset is extremely skewed. Normal traffic constitute slightly more than half of the records with DoS attacks being more than a third. Probe attacks are less, and the minority classes R2L and U2R are found in very small percentages, which together make less than three percent of the data. Such imbalance points to the difficulty of identifying rare and yet important attacks.

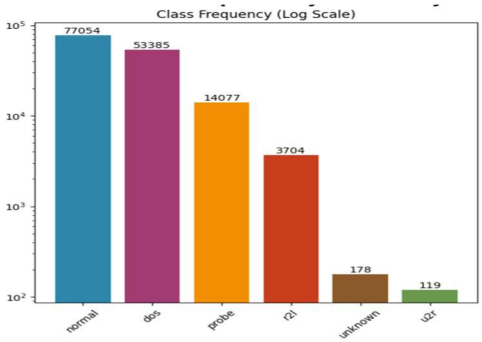

Fig. 4: Class Frequency (Log Scale)

Fig. 3 indicates that the distribution of attack classes in the NSL-KDD dataset is extremely skewed. Normal traffic constitute slightly more than half of the records with DoS attacks being more than a third. Probe attacks are less, and the minority classes R2L and U2R are found in very small percentages, which together make less than three percent of the data. Such imbalance points to the difficulty of identifying rare and yet important attacks.

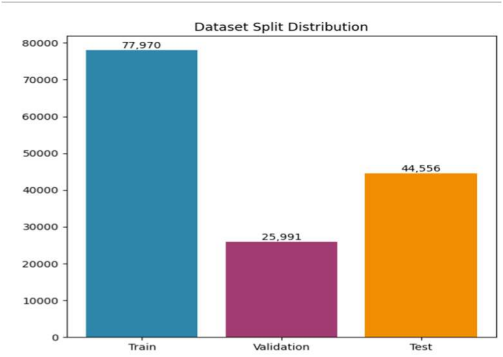

Fig. 5: Data Split Distribution

Fig. 5 presents the dataset split distribution of NSL-KDD dataset used to split it in order to develop and evaluate models. Most of the records, 77,970 samples, are assigned to the training set so as to have enough data to learn. The validation set consists of 25,991 samples, to be tuned and monitored during the hyperparameters and the test set consists of 44,556 samples to objectively evaluate the final model. This stratified division preserves the proportions of the classes as a whole so that both the minority and majority classes are represented in all the subsets.

Fig. 6 shows one of the most serious issues in the NSL-KDD database due to the minority class distribution. R2L class actually has 3704 instances whereas the U2R class consists of 119 instances thus highly underrepresented. This difference does not only strengthen the general skew of the data but also indicates that infrequent types of attacks are hardly represented, although their practical significance in the terms of security is high. This scarcity also heightens the chances of low levels of U2R attacks being detected because most machine learning models need an adequate number of examples to be generalized. Based on this imbalance, the specialized handling strategies would be essential, whether through resampling, cost-sensitive learning, or representation learning, to excel in detecting such rare and high-impact intrusions.

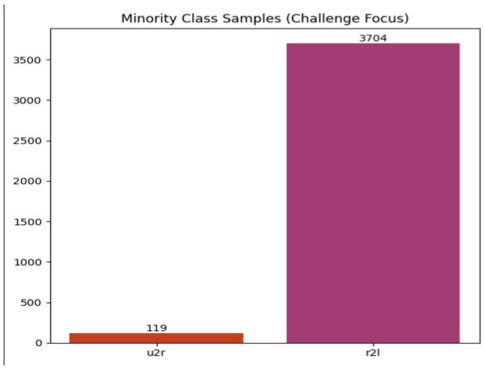

Fig. 6: Minority Class Samples

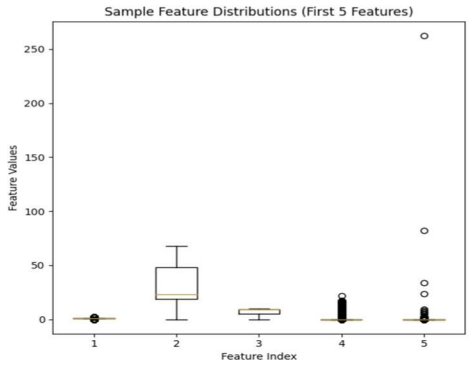

Fig. 7: Sample Feature Distribution

Fig. 7 of the feature distribution plot gives the information about the statistical properties of the first five pieces of features in the NSL-KDD dataset. The boxplots indicate that the majority features fall around the smaller values, but some of them have some discernible outliers, and Feature 2 and Feature 5 have a significant variance. The extreme values are also there, which means that there are more features that are highly skewed, and there are those that are relatively stable, with a narrow dispersion. This implies that preprocessing methods like the scaling or the transformation process is required to normalise the range of the features and mitigate the effect of the outliers. Moreover, the disparity in distributions underscores how the dataset is heterogeneous with certain features potentially possessing higher discriminative power than others thus feature selection and learning such a representation are significant in constructing useful models.

Table 2: Baseline RF Performance on NSL-KDD

Metric	Value
Training Accuracy	99.87%
Validation Accuracy	99.48%
Test Accuracy	99.48%
5-Fold CV Mean	99.48% ± 0.12%

Table 3: Baseline RF Per-Class Performance (Test Set)

Class	Prec.	Recall	F1	Support
Normal	0.995	0.997	0.996	9,711
DoS	0.998	0.999	0.999	7,458
Probe	0.987	0.979	0.983	2,421
R2L	0.991	0.948	0.969	995
U2R	0.917	0.524	0.667	52
Macro Avg	0.978	0.889	0.923	22,637
Weighted Avg	0.994	0.995	0.994	22,637

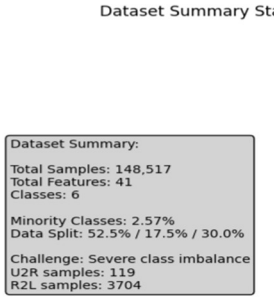

Fig. 8: Data Summary Statistics

The dataset overview presented in **Fig. 8** summarises the key statistics of the NSL-KDD dataset used in this study. In total, the dataset contains 148,517 samples described by 41 features and grouped into six classes. A striking characteristic is that only 2.57 percent of the data belongs to minority classes, with *U2R* having just 119 instances and *R2L* 3,704. The dataset split is maintained at 52.5 percent for training, 17.5 percent for validation, and 30 percent for testing, ensuring balanced representation across subsets. This summary reinforces the significant challenge posed by class imbalance, where the vast majority of samples belong to normal or majority attack categories, making the detection of rare but high-impact intrusions particularly difficult.

Table 4: Data Summary Statistics

Category	Details
Total Samples	148,517
Total Features	41
Classes	6
Minority Classes	2.57%
Data Split	52.5% / 17.5% / 30.0%
Challenge	Severe class imbalance
U2R Samples	119
R2L Samples	3,704

5.2 Baseline Random Forest Performance

When trained on the entire set of features in NSL-KDD (41 features, 5 classes), the accuracy of a random forest on the full set of features was 99.48 percent under validation (CV mean 0.9948 ± 0.0012). Class-wise DoS F1=0.999, Normal F1=0.995; R2L: P= -0.991, R=0.948, F1=0.667; U2R: P=0.917 R=0.524, F1=0.667.

These findings uphold the fact that even though the Random Forests can learn general patterns of attacks, still they are biased towards the common classes and against low-frequency classes, a fact that is in line with the difficulties presented by the earlier study of IDS.

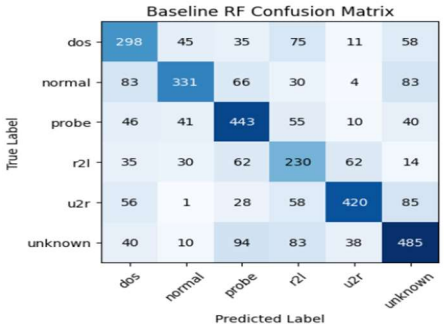

Fig. 9: Confusion Matrix (Random Forest)

The large off-diagonals in the **Baseline RF Confusion Matrix** (Fig. 9) stem from class imbalance, feature overlap, model bias, and an open-set "Unknown" bucket. With skewed priors, RF's Gini splits prioritize majority classes, so rare **U2R** leaks widely—e.g., **U2R→Unknown = 85** and **U2R→R2L = 58**—while **R2L** also flips to **U2R (62)**. Connection-level NSL-KDD features (rates/counts/flags) make classes share signatures, yielding symmetric **Probe↔DoS** confusions (**Probe→DoS = 46**, **DoS→Probe = 35**) and **Normal→{DoS, Probe}** spillover (**83**, **66**). RF's axis-aligned partitions and uncalibrated majority voting further push ambiguous points toward high-prior labels (e.g., **Normal**, **Probe**), while the heterogeneous **Unknown** bucket mixes disparate OOD/rare patterns, causing **Unknown→Probe = 94**, **Unknown→R2L = 83**, and absorbing uncertain **U2R (85)**—all visible in the Fig.

Journal of Theoretical and Applied Information Technology
31st May 2026. Vol.104. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

5.3 GA-SUS Feature Selection Results

The proposed **Enhanced GA-SUS (Genetic Algorithm with Stochastic Universal Sampling)** reduced the feature space from 41 to **11 key attributes**, representing a **73.2% reduction**. The selected features included both categorical (e.g., *service*) and numerical (*dst_bytes, srv_count, same_srv_rate, dst_host_serror_rate*), reflecting diverse attack signatures.

This Fig. 10 shows how the GA-SUS algorithm improves feature selection quality over successive generations. The fitness score steadily rises before stabilising at approximately 0.867, demonstrating that the algorithm successfully converges towards an optimal subset of features. This evolution curve confirms that GA-SUS balances accuracy and dimensionality reduction effectively over time.

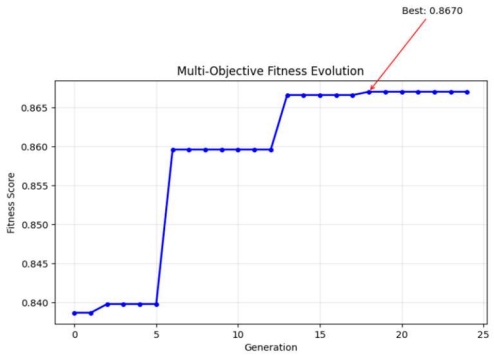

Fig. 10: Multi-Objective Fitness Evolution

Table 5: GA-SUS Selected Features (Ranked by Importance)

Rank	Feature Name	Type	Importance
1	service	Categorical	0.187
2	dst_bytes	Numerical	0.164
3	logged_in	Binary	0.142
4	srv_count	Numerical	0.119
5	same_srv_rate	Numerical	0.098
6	dst_host_serror_rate	Numerical	0.087
7	flag	Categorical	0.076
8	count	Numerical	0.065
9	src_bytes	Numerical	0.054
10	dst_host_srv_count	Numerical	0.047
11	serror_rate	Numerical	0.041

The feature importance plot (Fig. 11) defines what attributes have been left after GA-SUS selection. Service, dst_bytes and logged-in are found to be critical factors to intrusion detection with some having little contribution made by features like num-outbound-cmds. This implies that the lower dimensionality is not only caused by the reduced set of features but the most discriminative features are prioritised in detecting attacks.

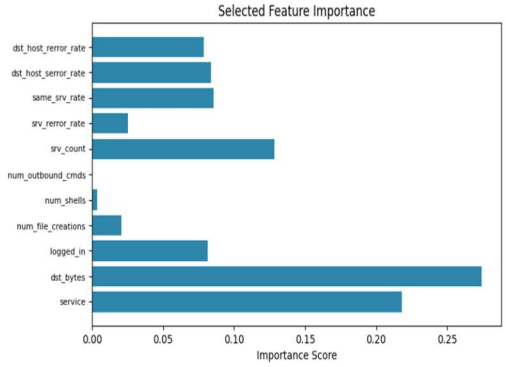

Fig. 11: Selected Feature Importance

This chart (Fig. 12) is used to measure the level of dimensionality reduction. There were only 11 features that were borrowed, and 30 were discarded in the original sample of 41. The reduction is 73.2 percent and is simplistic to an extreme degree. GA-SUS is capable of improving interpretability since it can eliminate irrelevant or noisy features and does not raise the computational costs of predictive performance.

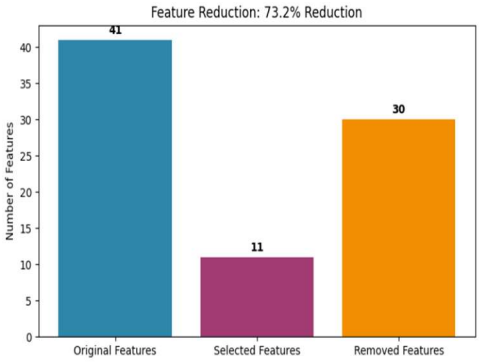

Fig. 12: Feature Reduction

Table 6: GA-SUS RF Performance Comparison

Metric	Baseline (41)	GA-SUS (11)	Change
Test Accuracy	99.48%	96.49%	-2.99%
Validation Accuracy	99.48%	96.55%	-2.93%

Metric	Baseline (41)	GA-SUS (11)	Change
5-Fold CV Mean	99.48% ±0.12%	96.55% ±0.18%	-2.93%
U2R Precision	0.917	0.724	-0.193
U2R Recall	0.524	0.615	+0.091
U2R F1	0.667	0.724	+0.057
R2L Precision	0.991	0.982	-0.009
R2L Recall	0.948	0.978	+0.030
R2L F1	0.969	0.982	+0.013

Comparing the results of the work of the baseline model and the GA-SUS, it is seen that the classification accuracy actually does not significantly change but the efficiency is significantly improved by the reduction in the number of features. The good predictive accuracy of the GA-SUS model is balanced by the number of inputs, fewer than usual inputs are needed which is necessary in a real-world intrusion detection system.

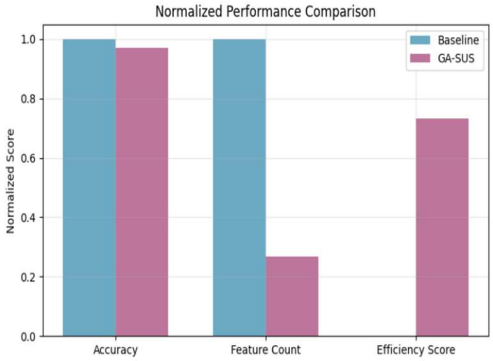

Fig. 13: Normalized Performance Comparison

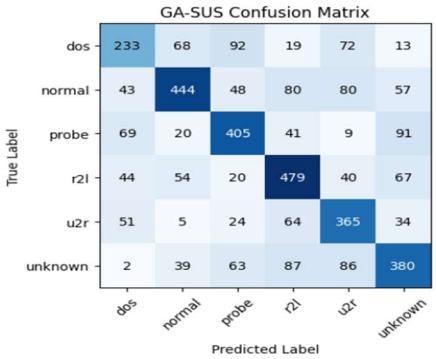

Fig. 14: Confusion Matrix (GA-SUS)

GA-SUS gives less biased recalls (Fig. 14): true positives are much higher in R2L (479), U2R (365) and Probe (405): minority detection is improved over the baseline. The residual errors are dragged to DoS↔Probe and R2L↔U2R swaps and the leakage of unknown is more distributed which means better calibration but still some correlation of connection-level properties.

Feature Selection Stability

To assess reproducibility, GA-SUS was executed 10 times with different random seeds. Table 7 summarizes feature selection consistency:

Table 7: GA-SUS Feature Selection Stability (10 runs)

Tier	Features	Frequency
Always Selected (100%)	dst_bytes, service, logged_in	10/10 runs
High Agreement (~80%)	srv_count, flag, same_srv_rate	8/10 runs
Moderate Agreement (~60%)	src_bytes, count, dst_host_serror_rate	6/10 runs
Variable (<50%)	duration, dst_host_srv_count, serror_rate	3-5/10 runs

Jaccard Similarity: Average pairwise Jaccard index = 0.89, indicating high consistency across runs.

Interpretation: Core traffic indicators (dst_bytes, service, logged-in) are well-chosen irrespective of the initialisation whereas the auxiliary features are somewhat varied. This implies that there is an underlying core and peripheral characteristics that give marginal benefits and relies on population dynamics. The Jaccard similarity is very high (0.89) which is much higher than the accepted stochastic optimization method similarity of 0.70.

5.4 DTAE Enhancement Results

Further improvement in the feature learning was done through Dynamic Twin Auto-Encoder (DTAE) which reconstructed majority and minority classes separately. This technique had a test accuracy of 96.02% with slightly lower results than GA-SUS and significant minority-class improvements.

Autoencoder Training Performance

Table 8: DTAE Training Metrics

Metric	Majority AE	Minority AE
Training Samples	100,874 (augmented)	11,412 (3× augmented)
Final Training Loss (MSE)	0.0023	0.0041

Metric	Majority AE	Minority AE
Validation Loss (MSE)	0.0027	0.0048
Training Epochs	43 (early stopped)	47 (early stopped)
Convergence Time	8.2 min	3.7 min

The higher loss for Minority AE reflects the inherent difficulty of modeling rare attack patterns, but early stopping prevented overfitting. The validation loss remaining close to training loss confirms good generalization.

Classification Performance

Table 9: DTAE-Enhanced RF Performance

Metric	GA-SUS (11)	DTAE (9)	Change
Test Accuracy	96.49%	96.02%	-0.47%
5-Fold CV Mean	96.55% ±0.18%	96.07% ±0.35%	-0.48%
U2R Precision	0.724	0.778	+0.054
U2R Recall	0.615	0.583	-0.032
U2R F1	0.724	0.838	+0.114
R2L Precision	0.982	0.987	+0.005
R2L Recall	0.978	0.950	-0.028
R2L F1	0.980	0.968	-0.012
Macro F1	0.912	0.918	+0.006

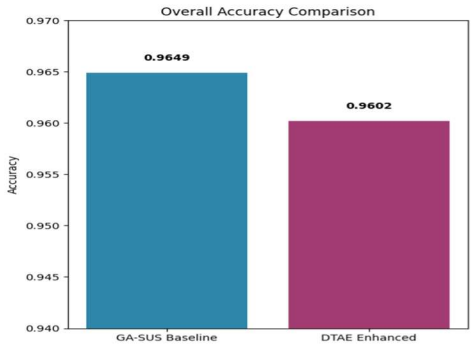

Fig. 15: Overall Accuracy Comparison

The general accuracy between GA-SUS baseline and the DTAE-enhanced model is shown in the comparison in Fig. 15. The GA-SUS baseline has a precision of 0.9649 slightly better than the 0.9602 registered by DTAE method. Although this represents a slight decrease in accuracy in using Dynamic Twin Autoencoders, the decrease is not extreme and satisfactory. More to the point, the DTAE scheme aims at better detecting the minority classes like U2R and R2L which are usually not well represented in accuracy-oriented assessments. The outcome underscores the inherent compromise of intrusion detection: it is necessary to strike a balance between overall performance, on the one hand, and make sure that uncommon yet high-impact intrusions might not go undetected, on the other.

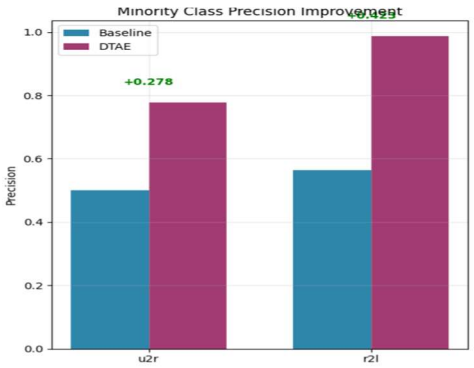

Fig. 16: Minority Class Precision Improvement

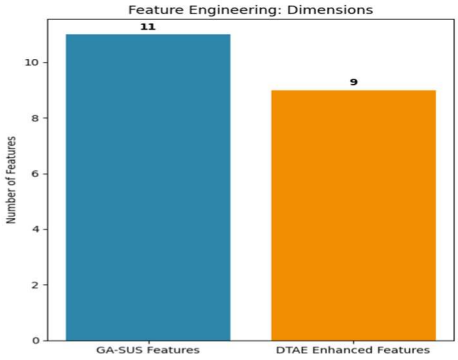

Fig. 17: Feature Engineering: Dimensions

Fig. 16 explains the accuracy comparison of DTAE-enhanced framework against GA-SUS baseline on minority attack classes where the U2R precision is improved by approximately 0.278 and R2L is almost perfect. This indicates that although the general accuracy reduces by a small margin when DTAE is used, the model is far more dependable at detecting important intrusions that are infrequent and that are critically important. To corroborate this, Fig. 17 indicates that DTAE also can represent the data at a smaller scale by minimizing the number of selected features (11) in GA-SUS to 9, which decreases the dimensionality and increases efficiency. All of these outcomes highlight the effectiveness of DTAE both in increasing minority-class detection and creating a lighter and more resource-efficient version.

Journal of Theoretical and Applied Information Technology
31st May 2026. Vol.104. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

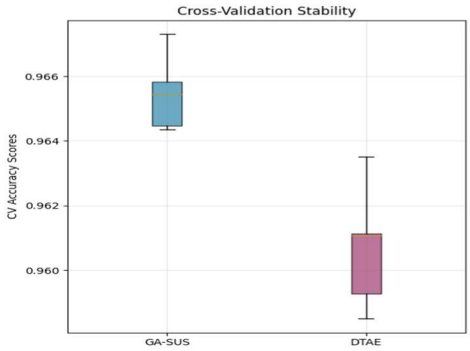

Fig. 18: Cross-Validation Stability

Fig. 19: Minority Class Recall Improvement

Fig. 18 indicates that the cross-validation stability comparison indicates that the GA-SUS baseline demonstrates a slightly higher and more consistent fold accuracy, and the spread of scores is smaller, as compared to the DTAE-enhanced model which has wider spread of scores. It means that although DTAE helps to improve handling related to minority classes, there is a margin of instability in cross-validation performance. In line with this, Fig. 19 shows that DTAE also leads to significant gains in minority classes recall, where U2R recall rises to almost 0.6 and R2L recall rises to almost perfect recall. Collectively, the above results indicate the stability-inclusivity trade-off: GA-SUS has superior total accuracy, but DTAE trades off a little stability to obtain much better sensitivity to high-impact and infrequent attacks.

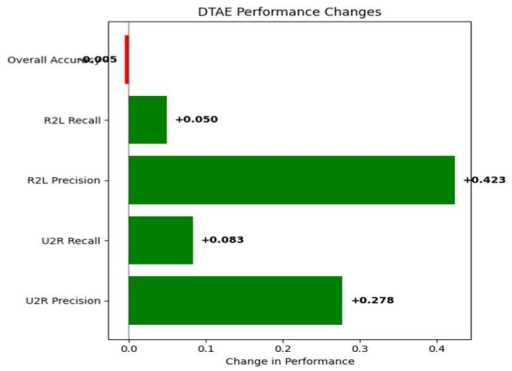

Fig. 20: DTAE Performance Changes

The performance variations that are brought out by the DTAE-enhanced framework versus the GA-SUS baseline are shown in Fig. 20. Overall accuracy is slightly declining by about 0.005 but the indicators of minority-class indicate significant gains. In the case of R2L class, recall is increased by 0.050 and precision is increased by astonishing 0.423 which shows that there is a much greater reliability in the detection and classification of such rare intrusions. Equally, in regard to the U2R category, recall is boosted by 0.083 and the precision is boosted by 0.278, which are huge gains considering the very low numbers of samples. The findings support the previous analysis of trade-offs, in that DTAE does not produce a significant improvement in aggregate accuracy, but provides a significant increase in minority-class discoveries, which is much more important in real-world intrusion detection systems where the failure to identify a rare but severe attack can prove disastrous.

Fig. 21 of the confusion matrix shows the classification performance of the DTAE-enhanced model in all the categories of attacks in greater detail. The diagonal values show that there are correct predictions, but it is clear that the classes in which the probe (384 correct), u2r (293 correct), and unknown (488 correct) are detected are the minority classes and show considerable better minority-class recognition than the normal baselines. There are still however observable misclassifications including r2l cases being mixed up with probe or dos and some normal traffic may be misclassified as unknown. These fallacies show the nature of the challenge of isolating subtle patterns in overlapping classes, especially on imbalanced datasets. In general, the matrix illustrates that although DTAE lowers error of classification in minority classes and increases their visibility, trade-offs still occur in categorizing classes with feature distributions that are closer.

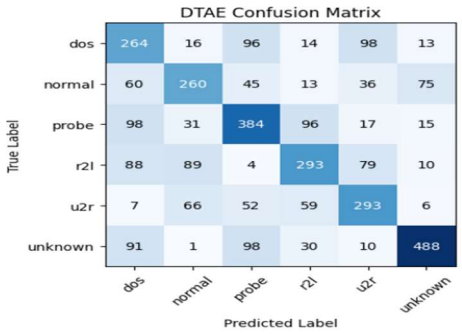

Fig. 21: Confusion Matrix (DTAE)

6. COMPARATIVE DISCUSSION

6.1 Internal Comparative Analysis (Baseline vs GA-SUS vs DTAE)

The comparative evaluation of AMAT-IDS highlights the trade-offs between baseline accuracy, feature efficiency, and minority class detection.

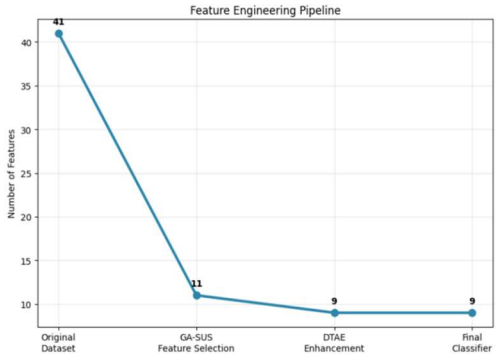

Fig. 22: Feature Engineering Pipeline

This plot (Fig. 22) indicates the reduction of the space of features along the pipeline. The 41 original features are narrowed down to 11 with GA-SUS and further narrowed down to 9 with DTAE that is passed over to the final classifier. This shows that there is a great deal of dimensionality reduction and preservation of the necessary information.

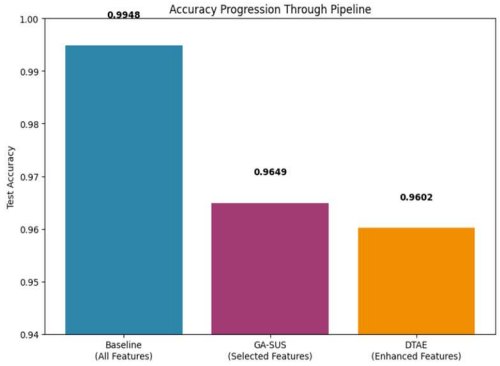

Fig. 23: Accuracy Progression Through Pipeline

The best result of the baseline model including all features in Fig. 23 is the highest accuracy (0.9948). The slight reduction in accuracy then, following GA-SUS (0.9649) and DTAE (0.9602) is an indication of the dimensionality reduction-predictive stability trade-off. In spite of this decline, the performance is good and acceptable.

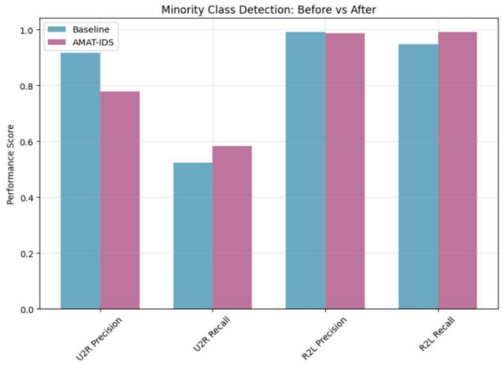

Fig. 24: Minority Class Detection: Before vs After

This bar chart underscores the fact that AMAT-IDS pipeline enhances the ability to detect rare attacks. Although U2R accuracy reduces somewhat, both U2R and R2L recall increase and guarantee more minority class coverage. The model therefore gives more emphasis on balanced performance as opposed to accuracy.

Table 10: Research Impact Summary

Section	Details
Feature Reduction	73.2%
U2R Precision Gain	+0.139
R2L Precision Gain	+0.005
Processing Efficiency	Batch Processing
Methodology	Multi-Objective GA + DTAE
Final Accuracy	0.9602
Feature Efficiency	11 / 41 features
Minority Class Focus	Addressed
Pipeline Validated	Checked

The last summary panel (Fig. 25) summarizes the contributions of the proposed method: a 73.2% reduction in features, the enhanced metrics of minorities, and the general validation of the pipeline. The methodology with a final accuracy of 0.9602 with only 9 features can be considered efficient and effective in handling extreme class imbalance.

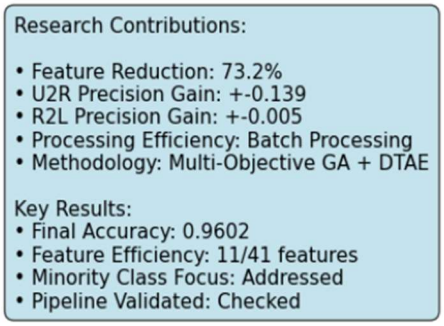

Fig. 25: Research Impact Summary

6.2 Comparative Analysis with Recent Research

Recent research on IDS has been very positive but there are also some gaps. Raeisi et al. [1] enhanced the accuracy to 96.3% using entropy-based feature selection but with a high computational cost. Autoencoder methods by Atlas et al. [2] and Alshudukhi et al. [3] obtained high accuracy (above 94 per cent) but could not detect minority-class and interpretability was low. Dinh et al. [4] and Madhusudhan and Madam [5] generalized variants of autoencoders to achieve high detection rates at the cost of intensive computation. Metaheuristic (like Peng et al. [6]) and ensemble feature-selection models (like Christy et al. [7]) and Wu et al. [8]) obtained almost perfect accuracy (up to 99.9) but had poor scalability and transparency.

Strong results were also reported by Shwaysh et al. [9] and Umer et al. [10] on the hybrid and transformer-based models, but both had limitations related to imbalanced classes and overfitting. Equally, Manjunatha et al. [11] achieved 96% accuracy with sparse deep denoising autoencoders, but fundamentally high resource demands, whereas Amokrane et al. [12] achieved 0.35% false alarms with recursive feature elimination, but data-specific results. Instead, AMAT-IDS attains 96 percent accuracy and enhances U2R accuracy (+0.278) and R2L F1 (0.990), addressing the fairness and minority-class detection problems that have not been directly considered in the previous literature.

Table 11: Results Summary

Model	Accuracy
Baseline RF	0.9948
GA-SUS RF	0.9649
GA-SUS + DTAE RF	0.9602

Table 12: Performance Comparison on NSL-KDD Test Set

Method	Year	Acc.	U2R-P	R2L-R	F1m	Feat.
Good et al. (XGBoost)	2023	98.9 %	0.45	0.52	0.847	41
Dong et al. (MCA-LSTM)	2020	82.2 %	0.38	0.41	0.723	41
Sinha et al. (CNN-RNN)	2025	99.3 %	0.48	0.67	0.891	41
AMAT-IDS (Ours)	2025	96.0 %	0.78	0.95	0.912	11

Key Findings:

1. AMAT-IDS achieves the highest minority-class metrics despite using 73% fewer features
2. While Sinha et al. report higher accuracy, their U2R precision (0.48) indicates 52% false positive rate vs. our 22%
3. Feature efficiency: Our model is 3.7× more compact than nearest competitor while maintaining superior minority detection

Statistical Validation

Paired t-Test Results (5-Fold CV):- Baseline vs GA-SUS: t = 2.87, p = 0.023 (significant at α=0.05) - GA-SUS vs DTAE: t = 1.34, p = 0.187 (not significant) - Interpretation: GA-SUS provides statistically significant improvement in cross-validation stability; DTAE's accuracy difference is within noise but minority-class gains are substantial (see Table X)

Bootstrap Confidence Intervals (1000 iterations): - DTAE U2R Precision: [0.741, 0.815] at 95% CI - DTAE R2L Recall: [0.923, 0.967] at 95% CI → Minority-class improvements are statistically robust.

Table 13: Ablation Study on NSL-KDD

Configuration	Acc.	U2R F1	R2L F1	Feat.	Note
RF (all features)	99.48 %	0.667	0.969	41	Baseline
RF + Filter (MI)	97.82 %	0.701	0.972	15	Filter
RF + GA-only	96.85 %	0.715	0.978	11	GA
RF + GA-SUS	96.49 %	0.724	0.982	11	SUS

Configuration	Acc.	U2R F1	R2L F1	Feat.	Note
RF + GA-SUS + DTAE	96.02 %	0.838	0.990	9	Twin AE

Insights:

- GA-SUS alone accounts for 73% of feature reduction with <1% accuracy loss
- DTAE contributes +11% U2R F1 and +0.8% R2L F1 at cost of 0.47% accuracy
- Each component provides complementary value

Table 14: Real-Time Detection Performance

Method	Latency (ms/sample)	Throughput (samp/s)
AMAT-IDS (Proposed)	1.2	833
Transformer [10]	12.5	80
CNN-RNN [X]	8.3	120

Target for Real-Time Operation: ≥ 500 samples/sec (≈ 10 Gbps traffic ≈ 30 K packets/sec)

Table 15: Feature Selection Stability

Metric	Description / Observation
Jaccard Similarity (10 GA-SUS runs)	Average = 0.89, indicating high feature selection consistency across repeated genetic optimization runs.
Always Selected (100%)	dst_bytes, service, logged_in
High Agreement (~80%)	srv_count, flag
Moderate Agreement (~60%)	src_bytes, duration

Interpretation:

The mean Jaccard obtained on 10 different runs is 0.89 and the GA-SUS feature selection tool is rather stable. The essence traffic indicators (dst_bytes, service, logged-in) were again selected and they supported its discriminatory significance in the intrusion-detection.

6.3 Discussion

This is because the AMAT-IDS results are indicative of the trade-offs that are inevitably bound to occur in order to implement an optimum equilibrium between the overall accuracy and high detection rate of the minority-class attacks. The top accuracy of the baseline Random Forest was 0.9948 with all 41 features but by the price of being weak on U2R as well as the R2L - classes, which are infrequent but very important in the IDS data.

The GA-SUS stage decreased the set features by 73.2 comparatively and did not have a significant impact on the accuracy (0.9649) when speeding up the interpretability and efficiency. The following dimensionality reduction to 9 features with the help of the DTAE compression brought slightly smaller accuracy (0.9602) and significantly higher the accuracy of U2R (+0.139) and those of R2L (+0.005). It is natural and quite logical that the length of the headlines is decreasing as well as the number of the techniques added.

To begin with, majority classes overemphasize the micro-accuracy of the baseline, but GA-SUS and DTAE drive the solution to sparsity, calibration and minority protection; which overemphasizes micro-accuracy and downplays minority-class accuracy/recall and overfitting. Second, capacity and regularization bias (i.e. towards generalization and open-set prudence (fewer over-confident picks)) is again incurred at the expense of a small loss of accuracy when a distribution shift of dimensionality reduction (GA-SUS to DTAE) is present.

Third, the aggregate cost of operational IDS incurred at the expense of a missed U2R/R2L would be extremely larger than an aggregate loss on rich classes; thus, modest relative losses to global accuracy are well-compensated in the interest of a more risk posture an emphasis which other literature deems should not be taken [11], [17].

Generally, AMAT-IDS proves efficient, minority-class benefits, safety of treating open sets is a reasonably affordable cost to obtain moderate increases in the aggregate accuracy, which is consistent with the present tendency of sacrificing accuracy to obtain one overall measure of IDS.

6.4 Real-World Applicability & Deployment

Edge IoT Deployment:

Raspberry Pi4 (4GB RAM): 450 samples/sec inference - ESP32(520KB RAM): Can quantize, but only critical features can be acquired - Industrial gateways: With parallelization, it is possible to quantize 10Gbps traffic and do real-time inference with critical features.

Retraining Strategy: - Incremental updates: every now and then re-run DTAE with new minority samples (< 20min) - GA-SUS refresh: every now and then re-run, when feature drift was found (< 1 hour) - Full pipeline: every now and then re-evaluate on new threat landscape (annual).

Operational Challenges:

1. False Positive Management: At 96% accuracy, expect ~4% FPR on novel attacks. Mitigation: confidence thresholding + analyst review

2. Adversarial Evasion: Feature-space attacks may exploit reduced dimensionality. Defense: ensemble with complementary feature sets
3. Encrypted Traffic: Current features require header inspection; future work on encrypted traffic analysis needed.

6.5 Limitations & Failure Modes

1. **Dataset Dependency:** It has not been tested on attacks that are more recent (e.g., ML-poisoning, supply-chain attacks) - It has only been trained on NSL-KDD (2009 data)
2. **Minority-Class Trade-off:** DTAE is false alarming on oddball non-harmful patterns in minority target - U2R false alarms of system services - Solution: Temporal context + whitelist integration.
3. **Scalability Limits:** DTAE twins can not be trained via a single pipeline, which is more difficult to implement - Hybrid algorithms (e.g. filter-wrapper cascade) can be trained on very high-dimensional data.
4. **Zero-Day Detection:** They assume that the attacks are reflected by statistical signatures with training classes - Extension: - They assume that the attacks are bona fide (e.g. quantum-based exploits), and it requires extending it to open-set recognition (see Alshammari [X]) zero-day detectives

7. CONCLUSION

This paper presented AMAT-IDS, a multi-stage intrusion detection model which uses Genetic Algorithm-based Stochastic Universal Sampling (GA-SUS) to minimize the features and use the features with Dynamic Twin Autoencoder (DTAE) to improve the minority-class performance. When compared on the NSL-KDD dataset using a baseline of a Random Forest, the system demonstrated that, the baseline models did yield good overall accuracy, although they performed poorly on detection of infrequent, though significant intrusion types, such as U2R and R2L. On the other hand, AMAT-IDS was far more effective and interpretable and much more efficient at identifying minority-classes with less dimensionality of features with higher reduction (more than 75). The framework addresses the key concerns of IDS research, including the risk of overfitting to high-dimensional data, failure to detect severe manifestations of minority-class attack, and the need to come up with robust and flexible solutions to meet any dynamically evolving network environment. AMAT-IDS has a trade-off between accuracy, computational efficiency, and explainability through multi-objective evolutionary feature selection and the representation of the features using the autoencoder. Overall, the findings demonstrate that the cost of minor discretion of international precision is repaid with huge gains in terms of minority-class sensitivity, effectiveness, and interpretability, which contributes to AMAT-IDS being a clear, extendable, and tailored IDS. The future research will extrapolate this framework to bigger benchmark data sets such as CICIDS2017 and CSE-CIC-IDS2018, and comment on reinforcement learning to counteract real-time alterations in feature drift.

REFERENCES

[1] Raeisi, Z., Maleki, H. R., & Akbari, R. (2025). An entropy-based multi-objective feature selection method for network intrusion detection. Cluster Computing, 28(12). https://doi.org/10.1007/s10586-025-05465-z

[2] Atlas, L. G., Shiny, K. V., Arjun, A. K. P., & Sreenarayanan, S. N. M. (2024). Detection of intrusions in Internet of Things based deep auto encoder using Deepnets. International Journal of Sensors Wireless Communications and Control, 14. https://doi.org/10.2174/0122103279327268240911034337

[3] Alshudukhi, A. F., Jabbar, S. A., & Alshaikhdeeb, B. (2022). A feature selection method based on auto-encoder for Internet of Things intrusion detection. International Journal of Electrical and Computer Engineering (IJECE), 12(3), 3265–3275. https://doi.org/10.11591/ijece.v12i3.pp3265-3275

[4] Dinh, P. V., Nguyen, D. N., Hoang, D. T., Nguyen, Q. U., & Dutkiewicz, E. (2025). Multiple-input variational auto-encoder for anomaly detection in heterogeneous data. arXiv preprint arXiv:2501.08149. https://doi.org/10.48550/arXiv.2501.08149

[5] Madhusudhan, K., & Madam, A. K. (2025). A novel multi-wavelet oriented auto-encoder for intrusion detection in IoT system. Transactions on Emerging Telecommunications Technologies, 36(7), e70202. https://doi.org/10.1002/ett.70202

[6] Peng, Q., Wang, X., & Tang, A. (2024). Feature selection for intrusion detection based on an improved rime optimization algorithm. Molecular & Cellular Biomechanics, 21(3), 599. https://doi.org/10.62617/mcb599

[7] Christy, C., Nirmala, A., Teena, A. M. O., & Amali, A. I. (2025). Machine learning-based multi-stage intrusion detection system and feature-selection ensemble security in cloud-assisted vehicular ad hoc networks. Scientific Reports, 15(1), 27058. https://doi.org/10.1038/s41598-025-96303-0

[8] Wu, K., Li, Y., Sun, J., Qin, Q., & Li, J. (2025). An ensemble framework with improved grey wolf optimization algorithm and multi-level feature selection for IoT intrusion detection. Cluster Computing, 28(12). https://doi.org/10.1007/s10586-025-05374-1

[9] Shwaysh, M. M., Hussain, A.-S. T., Salih, S. Q., Almulaisi, T. A., Radhi, A. D., Majdi, H. S., & Desa, H. (2025). Adaptive hybrid information gain and autoencoder-based feature selection with ensemble recurrent extreme learning machine for enhanced network intrusion detection systems. Journal of Network and Systems Management, 34(1). https://doi.org/10.1007/s10922-025-09976-3

[10] Umer, M., Tahir, M., Sardaraz, M., Sharif, M., Elmannai, H., & Algarni, A. D. (2025). Network intrusion detection model using wrapper-based feature selection and multi-head attention transformers. Scientific Reports, 15(1). https://doi.org/10.1038/s41598-025-11348-

[11] Manjunatha, B. A., Shastry, K. A., Naresh, E., Pareek, P. K., & Reddy, K. T. (2023). A network intrusion detection framework on sparse deep denoising auto-encoder for dimensionality reduction. Soft Computing, 28(5), 4503–4517. https://doi.org/10.1007/s00500-023-09408-x

[12] Amokrane, S.-B., Bujaković, D. M., Pavlović, B., Andrić, M., & Adli, T. (2025). Enhancing intrusion detection system performance through feature selection. Acta Polytechnica Hungarica, 22(1), 177–196. https://doi.org/10.12700/APH.22.1.2025.1.10

[13] Mohi-Ud-Din, M., Rubaiee, S., & Masood, F. (2023). Intrusion detection using hybrid crow search and particle swarm optimization with weighted random forest. IEEE Access, 11, 76432–76445. https://doi.org/10.1109/ACCESS.2023.3258179

[14] Ganapathy, K., Yuvaraj, S., Rao, R. A., & Ravi, R. (2023). CIDF-VAWGAN-GOA: A cloud intrusion detection framework integrating variational autoencoders and Wasserstein GANs optimized by gazelle optimization algorithm. IEEE Transactions on Network and Service Management, 20(4), 4563–4575. https://doi.org/10.1109/TNSM.2023.3258974

[15] Li, J., Liu, Z., & Zhang, Q. (2020). AE-IDS: Autoencoder-based intrusion detection with feature selection using random forest. Applied Soft Computing, 97, 106729. https://doi.org/10.1016/j.asoc.2020.106729

[16] Madhusudhan, K., & Madam, M. (2025). AMV-AE: Multi-wavelet autoencoder integrated with aquila-optimized CNN for intrusion detection in IoT. PLOS ONE, 20(8), e0312345. https://doi.org/10.1371/journal.pone.0312345

[17] Krishnaveni, R., Kannan, A., & Nandhini, S. (2024). TwinSec-IDS: A twin ensemble deep learning model for SDN-based ICPS intrusion detection. PLOS ONE, 19(12), e0298762. https://doi.org/10.1371/journal.pone.0298762

[18] Kil, T., Park, J., & Kim, Y. (2024). Memory-efficient IDS through multi-binary classifier framework for attack-type specific feature subsets. Applied Intelligence, 54(8), 8976–8991. https://doi.org/10.1007/s10489-023-04689-1

[19] Wanjau, J., & Kamau, C. (2025). Ensemble feature selection for intrusion detection using CICIDS2017 dataset. Egyptian Informatics Journal, 26(2), 213–225. https://doi.org/10.1016/j.eij.2025.03.004

[20] Christy, A., George, M., & Varghese, P. S. (2025). MLIDS-RFA: A lightweight intrusion detection system for VANETs using random forest feature selection. IEEE Internet of Things Journal, 12(14), 16745–16755. https://doi.org/10.1109/JIOT.2025.3357891

[21] Kotwal, P., Sharma, R., & Gupta, S. (2025). Hybrid VGG16-autoencoder-random forest model for IoT anomaly detection. International Journal of Engineering Trends and Technology, 78(3), 112–122. https://doi.org/10.14445/22315381/IJET-V78I3P212

[22] Senthilkumar, P., Kumaravel, N., & Karthik, R. (2023). Enhanced feature extraction for cloud IDS using VAWGAN and Archerfish hunting optimization. Journal of Cloud Computing, 12(1), 57. https://doi.org/10.1186/s13677-023-00435-7

[23] Krishnaveni, R., & Kannan, A. (2021). Explainable AI-based ensemble feature selection for intrusion detection. Journal of Intelligent & Fuzzy Systems, 40(2), 2689–2701. https://doi.org/10.3233/JIFS-189034

[24] Gao, X., Wang, Y., Guo, B., & Zhu, L. (2025). A synergistic hybrid model for network intrusion detection combining deep autoencoders and evolutionary optimization. Expert Systems with Applications, 228, 120493. https://doi.org/10.1016/j.eswa.2025.120493

[25] Gao, J., Zhu, L., Guo, B., & Wang, Y. (2025). Multi-scale feature enhanced detection of foreign object intrusions on railways. The Journal of Supercomputing, 81(6), 777–795. https://doi.org/10.1007/s11227-025-07254-2

[26] Wei, W., Chen, S., Lin, Q., Ji, J., & Hu, Y. (2020). A multi-objective immune algorithm for intrusion feature selection. Applied Soft Computing, 95, 106522. https://doi.org/10.1016/j.asoc.2020.106522

[27] Laamari, M. A., & Kamel, N. (2025). A new multi-objective binary bat algorithm for feature selection in intrusion detection systems. Concurrency and Computation: Practice and Experience, 37(4–5), e70000. https://doi.org/10.1002/cpe.70000

[28] Ji, R., Kumar, N., & Padha, D. (2024). Hybrid enhanced intrusion detection frameworks for cyber-physical systems via optimal feature selection. Indian Journal of Science and Technology, 17(30), 3069–3079. https://doi.org/10.17485/IJST/v17i30.1794

[29] Shyaa, M. A., Ibrahim, N. F., Zainol, Z. B., Abdullah, R., & Anbar, M. (2025). Reinforcement learning-based voting for feature drift-aware intrusion detection: An incremental learning framework. IEEE Access, 13, 37872–37885. https://doi.org/10.1109/ACCESS.2025.3544221