Journal of
Cyber Security

Tech Science Press

https://doi.org/10.32604/jcs.2026.079097

ARTICLE

Domain Knowledge-Guided Training for NIDS: A Class-Agnostic Evaluation of Robustness on Imbalanced Datasets

Zakaria S. M. Abdelhalim*, Nahla Belal and Mohamed Seifeldin

College of Computing and Information Technology, Arab Academy for Science, Technology and Maritime Transport, Smart Village, Giza, Egypt

*Corresponding Author: Zakaria S. M. Abdelhalim. Email: z.abdelha53974@student.aast.edu

Received: 14 January 2026; Accepted: 10 March 2026; Published: 06 April 2026

ABSTRACT: The rapid expansion of IoT and cloud services has increased the scale and complexity of modern networks, making intrusion detection challenging. Although deep learning-based Network Intrusion Detection Systems (NIDS) often report high accuracy, such metrics can be misleading on highly imbalanced datasets, where performance is dominated by majority classes and rare attacks remain poorly detected. This issue stems from global optimization strategies that encourage models to rely on dominant feature patterns, limiting their ability to capture the class-specific features required to identify infrequent attack types. To address this limitation, this work proposes a domain knowledge-guided attentional training framework. In the first stage, SHAP is used to extract per-class feature importance vectors, identifying features that are relevant or negligible for each attack type. In the second stage, a knowledge-guided loss function introduces a weighted gradient regularization term that penalizes reliance on features deemed unimportant for the target class, encouraging the model to focus on class-specific features. The proposed approach is evaluated on the highly imbalanced CIC-IDS-2017 dataset. While a baseline CNN achieved an overall accuracy of 99.39%, it failed to detect rare attacks such as Heartbleed (F1 score = 0.0). In contrast, the knowledge-guided model demonstrated improved robustness across attack classes. Emphasizing Macro F1 score as a class-agnostic metric, the proposed framework improved the Macro F1 from 0.6180 to 0.7560, achieved an F1 score of 1.0 on Heartbleed, and maintained overall accuracy (99.56%). Overall, this work enables NIDS to move beyond static optimization by focusing on class-specific features, leading to improved generalization for rare attacks.

KEYWORDS: IDS; deep learning; imbalanced learning; SHAP; gradient regularization

1 Introduction

The rapid expansion of interconnected systems, from the Internet of Things to cloud computing, has led to a substantial increase in network scale and complexity. While this increased connectivity enables a wide range of services, it also expands the attack surface, making robust Network Intrusion Detection Systems (NIDS) a critical component of modern cybersecurity architectures [1,2].

In recent years, Machine Learning and Deep Learning have become the dominant approach for NIDS, with many publications reporting very high accuracy on benchmark datasets. For instance, studies on the CIC-IDS-2017 dataset often report accuracy exceeding 98% across a wide range of machine learning and deep learning approaches, including hybrid and ensemble models [1–5]. Other studies further improve performance through diverse strategies such as federated learning, optimization techniques, and enhanced data processing methods [6–10]. More recent works continue to explore advanced learning paradigms and

adaptive techniques for intrusion detection [11–14]. However, this paper argues that these impressive metrics are often an "illusion of high performance". Recent studies confirm that accuracy can be misleading in imbalanced datasets, as it is driven by the model's performance on the majority class while masking failures on critical minority attacks [15–17].

This illusion stems from the highly imbalanced nature of NIDS datasets [17–21]. As shown in Table 1, the widely used CIC-IDS-2017 dataset is composed of approximately 80% BENIGN traffic. In contrast, several attack classes are represented by extremely few samples—for instance, Heartbleed includes only 11 instances, and Infiltration includes just 36 instances. Such extreme class imbalance severely biases learning algorithms toward majority behavior, hindering their ability to generalize to rare but critical attack types. The imbalanced distribution is visualized in Fig. 1.

Table 1: Data distribution of the CIC-IDS-2017 dataset, highlighting extreme class imbalance.

Label	Count
BENIGN	**2,273,097**
DoS Hulk	231,073
PortScan	158,930
DDoS	128,027
DoS GoldenEye	10,293
FTP-Patator	7938
SSH-Patator	5897
DoS slowloris	5796
DoS Slowhttptest	5499
Bot	1966
Web Attack—Brute Force	1507
Web Attack—XSS	652
Infiltration	36
Web Attack—SQL Injection	21
Heartbleed	11

Note: The BENIGN class is highlighted in bold as it represents the dominant class in the dataset.

Standard deep learning models trained with conventional loss functions such as categorical cross-entropy (L_{CCE}) are mathematically incentivized to minimize overall error. In imbalanced datasets, this naturally leads the model to focus on the majority classes. This leads to high accuracy and F1 scores, as these metrics are dominated by the model's excellent performance on BENIGN traffic and high-volume attacks. This focus masks the model's total failure to learn the patterns of rare classes. This often results in poor generalization to new or dynamic attacks [1,4]. Critically, this failure occurs because different cyber-attacks are triggered by different sets of features [2,22,23]. Standard models, optimizing only for overall error reduction (L_{CCE}), learn patterns from globally dominant features. These features, while useful for common classes, often act as noise that obscures the subtle, distinct signatures of rare attacks. The model learns spurious correlations from these globally important but locally irrelevant features, effectively drowning out the weak signals of minority classes.

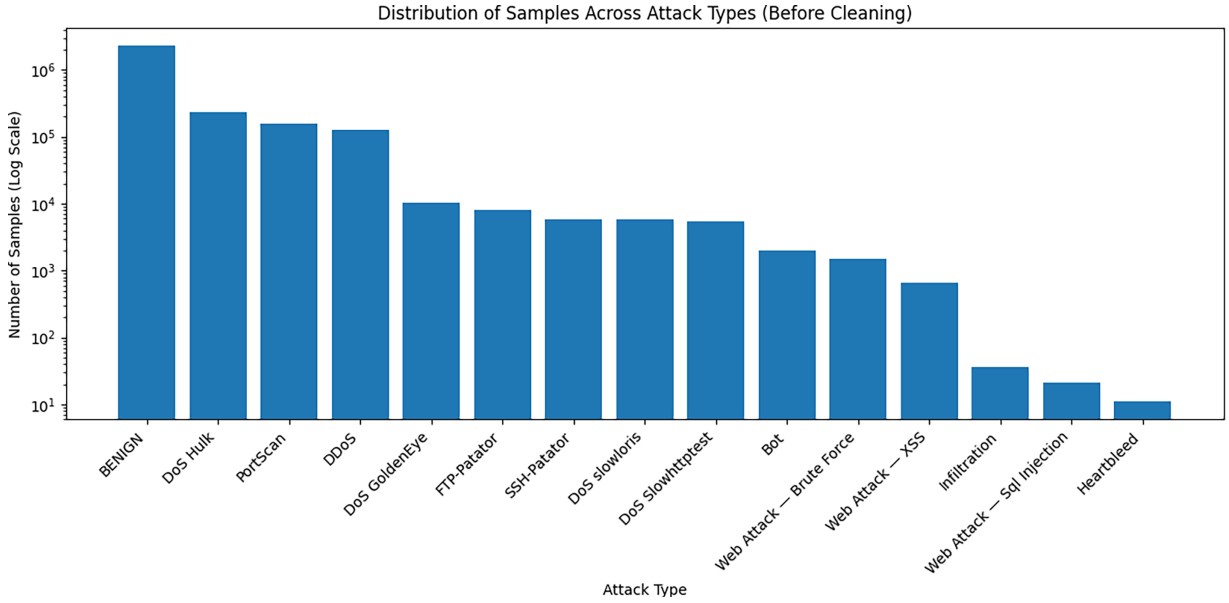

Figure 1: Data Distribution of the CIC-IDS-2017 Dataset.

The true measure of a model's ability to protect against all threats is the Macro F1 score, which averages the F1 score of each class equally, giving equal importance to both frequent and rare attacks. It is insensitive to class imbalance and thus better reflects a model's robustness across all attack types. On this metric, standard models perform poorly because they learn a single, static set of rules that is not robust or context-aware [2,22,23]. The features that identify a DDoS flood (e.g., high packet volume) are fundamentally different from those that identify a DoS Slowloris attack (e.g., long flow duration). Furthermore, the features identifying a rare attack like Heartbleed might be subtle variations within features that are otherwise unimportant for BENIGN or DDoS traffic.

To address this gap, this work introduces a Domain Knowledge-guided Attentional Training framework.

The main contributions of this work are as follows:

1. The primary contribution of this work lies in the proposed knowledge-guided loss function rather than in architectural modifications to the CNN, which is intentionally kept standard to isolate the effect of the proposed training objective.
2. This work introduces a new loss formulation, $L_{Knowledge}$, which integrates domain knowledge directly into the optimization objective. This loss employs a gradient regularization term that penalizes the model for relying on globally dominant but locally irrelevant features, thereby guiding it to dynamically focus on class-specific, informative features crucial for detecting rare attacks.
3. The knowledge term is derived using SHAP (Shapley Additive exPlanations) [23–26], which quantifies per-class feature importance from a baseline CNN. The resulting knowledge vectors act as an expert guide, enabling the proposed loss function to regularize training in a class-aware manner.
4. The proposed framework was evaluated on the CIC-IDS-2017 dataset, resulting in an improvement in the Macro F1 score (+13.8 percentage points) while maintaining competitive overall accuracy and F1 score, indicating that the model is better able to learn minority attack patterns without degrading performance on dominant classes.

The proposed framework demonstrates an improvement in detecting minority attack classes within the CIC-IDS-2017 dataset. While the baseline CNN achieved high overall performance but failed to detect

critical rare attacks such as Heartbleed, the proposed model achieved a Macro F1 score of 0.7560 compared to 0.6180 for the baseline and successfully enhanced the detection of minority threats, including achieving a perfect score for Heartbleed. These outcomes highlight the importance of class-aware, knowledge-guided optimization for addressing the extreme imbalance characteristics of modern intrusion detection datasets.

The remainder of this paper is structured as follows.

Section 2 presents an overview of existing approaches in network intrusion detection systems.

Section 3 explains the proposed methodology, including the construction of SHAP-derived knowledge vectors and the knowledge-guided loss function.

Section 4 details the experimental setup and reports the evaluation results.

Section 5 discusses the implications of the findings.

Section 6 concludes the paper and outlines potential directions for future work.

2 Related Work

Research on Network Intrusion Detection Systems (NIDS) has explored improvements from several directions, including data handling strategies, feature selection techniques, and model-level enhancements. These approaches aim to strengthen detection performance, address dataset complexity, and improve robustness across diverse attack types. Accordingly, efforts to improve NIDS performance on imbalanced datasets typically fall into three categories:

2.1 Data-Level Solutions

These methods focus on rebalancing the dataset, which are broadly categorized into **oversampling**, **undersampling**, and **generative** approaches. Common oversampling techniques include **Random Oversampling** [9,18,19] and **SMOTE** (Synthetic Minority Over-sampling Technique) [3,9,10,18,19], along with its variants like **Borderline SMOTE** [18] and **ADASYN** (Adaptive Synthetic Sampling) [18,19]. **Undersampling** methods, such as **Random Undersampling** [9,18,19], **NearMiss** or **Tomek Links** [19], are also used. While these techniques have shown effectiveness in improving baseline model performance, they cannot be used alone as a solution for class imbalance, as they can either create problems of overfitting or cause the model to learn from noise that does not truly represent real rare attacks [27]. Building upon these traditional data-level strategies, more advanced generative approaches, such as Generative Adversarial Networks (GANs), have been explored to produce higher-fidelity synthetic data [18,19,28]. Specific variants, including IGAN (Imbalanced GAN) [20] and CTGAN (Conditional Tabular GAN) [21], have been applied to NIDS datasets to better model complex minority distributions, though achieving semantic fidelity to real attacks remains an ongoing challenge.

2.2 Feature-Level Solutions

These solutions aim to find the most impactful subset of features to reduce complexity and noise. Methods range from dimensionality reduction techniques like Principal Component Analysis (PCA) [2,10,13] and feature embedding [13], to filter and wrapper methods like Recursive Feature Elimination (RFE) [24], XGBoost-based importance [1], and Fast Correlation-Based Filter (FCBF) [10]. Other approaches use optimization algorithms like the Honey Badger Optimization (HBO) [22] or XAI methods like SHAP for feature selection [23,24]. This approach has been validated in recent studies, which show that XAI-driven feature selection can significantly improve IDS efficiency [23,24]. While feature selection can maintain high accuracy, **for example, Ahmed et al. [4] achieved 98.79% accuracy with only 20 features, and Chen et al. [24] achieved approximately 98% accuracy with only 15 features**. The fundamental limitation is

selecting a single, static subset of features for all situations [1,2,4,10,22,23]. This is suboptimal, as the optimal feature set for a PortScan is different from that of a DoS Slowloris attack. Temporal-aware feature selection strategies are also being explored [14].

2.3 Model-Level Solutions

These approaches focus on developing more complex architectures or advanced models. Examples include complex hybrid deep learning models like CNN-LSTM [1], CapsNet-BiLSTM [11], or CNN-Decision Forest [2], as well as advanced ensembles [4,5] and general DL/ML hybrid approaches [7]. This category also includes high-performance machine learning models such as K-Nearest Neighbors (KNN) [3] or advanced Random Forest (RF) implementations [9,13]. Other solutions explore lightweight or efficient models, such as federated learning with pruning [6] or optimized Deep Neural Networks (DNNs) [8], and novel training paradigms like contrastive learning [12]. These often achieve high accuracy metrics, with ML models like KNN reaching 99.36% [3] and advanced RF models reaching 99.90% [9]. However, they are still trained using a standard loss function on the same imbalanced data. They are ultimately constrained by the same flawed optimization objective: to minimize error by focusing on the majority class, potentially neglecting rare attack detection.

Our work is distinct from all three. It does not alter the data (Data-Level) or rely on a single, globally static feature subset (Feature-Level) for all attacks. Instead, this work introduces a dynamic, class-aware feature weighting mechanism guided by SHAP-derived knowledge vectors. This approach allows the model to emphasize features that are important for each attack type during training including rare attacks, rather than selecting one fixed subset for all classes.

3 Proposed Methodology

To address the problem of static optimization that favors majority classes and overlooks minority ones, this paper uses a two-stage, model-agnostic framework. The framework is visualized in Fig. 2.

3.1 Preprocessing

The CIC-IDS-2017 dataset was formed by loading and merging the provided CSV files. Duplicate entries and all rows containing missing or infinite values were removed to ensure clean, reliable input. All numerical features were then scaled to the [0, 1] range using Min–Max normalization, and the class labels were converted to integer form using a LabelEncoder.

After preprocessing, the dataset was divided into training and testing subsets using an 80/20 stratified split. Stratification preserves the original class proportions, which is essential given the extreme imbalance of the dataset and ensures consistent representation of minority attacks in both subsets.

3.2 Stage 1: Offline Expert Knowledge Annotation

The first stage is a one-time offline pre-processing step designed to create a class-specific knowledge base. By quantifying feature relevance using SHAP before training begins, this offline annotation phase provides a static guide during the active training phase in Stage 2.

The first step involves training a baseline CNN model. Then, SHAP is used to determine the contribution of each feature to the detection of each attack class. SHAP values were calculated for samples belonging to each class, and the mean absolute SHAP value for each feature across the samples of a given class was used to construct the knowledge vector w for that class.

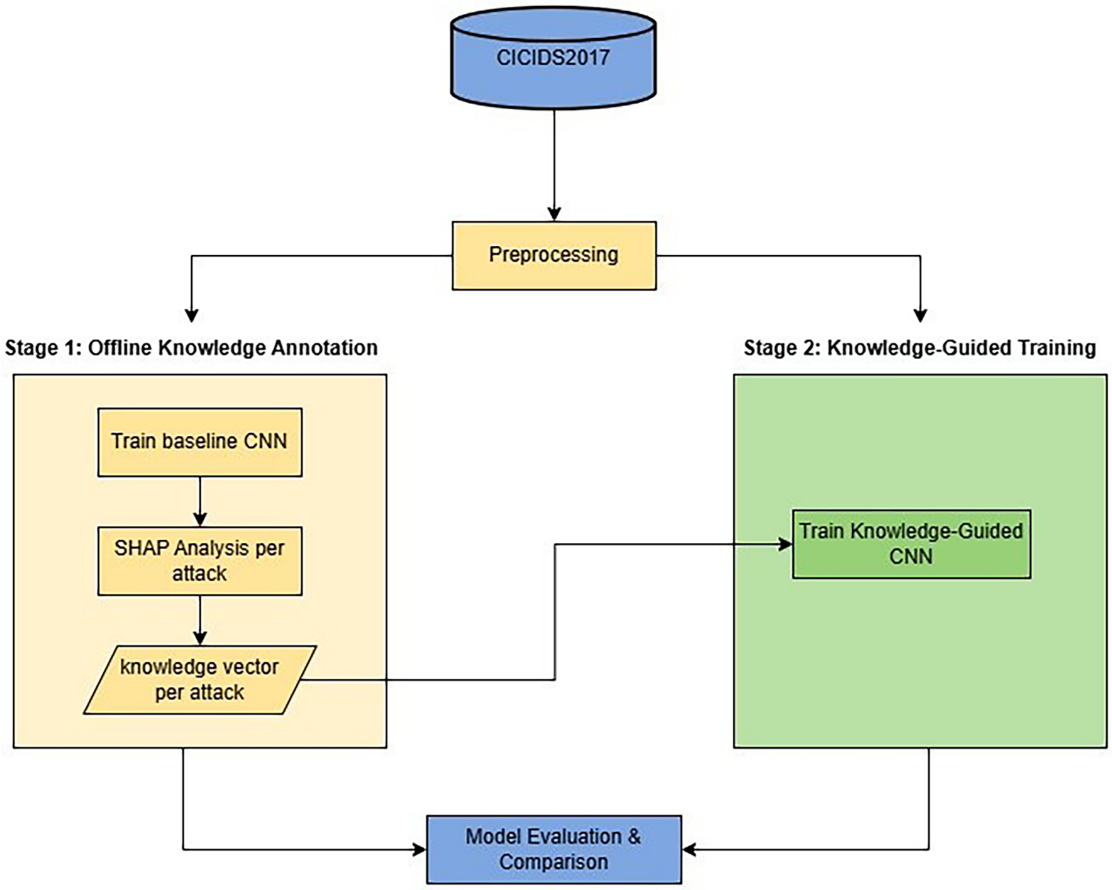

Figure 2: Overview of the proposed knowledge-guided framework.

To identify critical features for each attack class, SHAP values were aggregated separately per label by computing the mean absolute contribution of each feature over samples belonging to that class. This process produces class-specific feature importance rankings for all attack types. Fig. 3 visualizes the resulting feature–attack relationships using a SHAP-based heatmap, while Table 2 summarizes the top-ranked critical features per attack. These results demonstrate that feature relevance is strongly class-dependent, motivating the use of class-specific knowledge vectors in the proposed framework.

This work generates 14 knowledge vectors corresponding to the 14 attack classes. Each vector contains 78 elements, where each element indicates whether the associated feature contributes to the identification of that attack class. Features with zero SHAP contribution for a given class are assigned a value of 0 in the corresponding knowledge vector, marking them as irrelevant for that attack. This aggregation yields a robust estimate of feature importance per class.

- If w_j is low (equal 0), feature $\boldsymbol{j}$ is unimportant or misleading for the attack corresponding to the sample.
- If w_j is high (equal 1), feature $\boldsymbol{j}$ is critical for the attack corresponding to the sample. This process is performed only once to create a data-driven guide on a per-class basis.

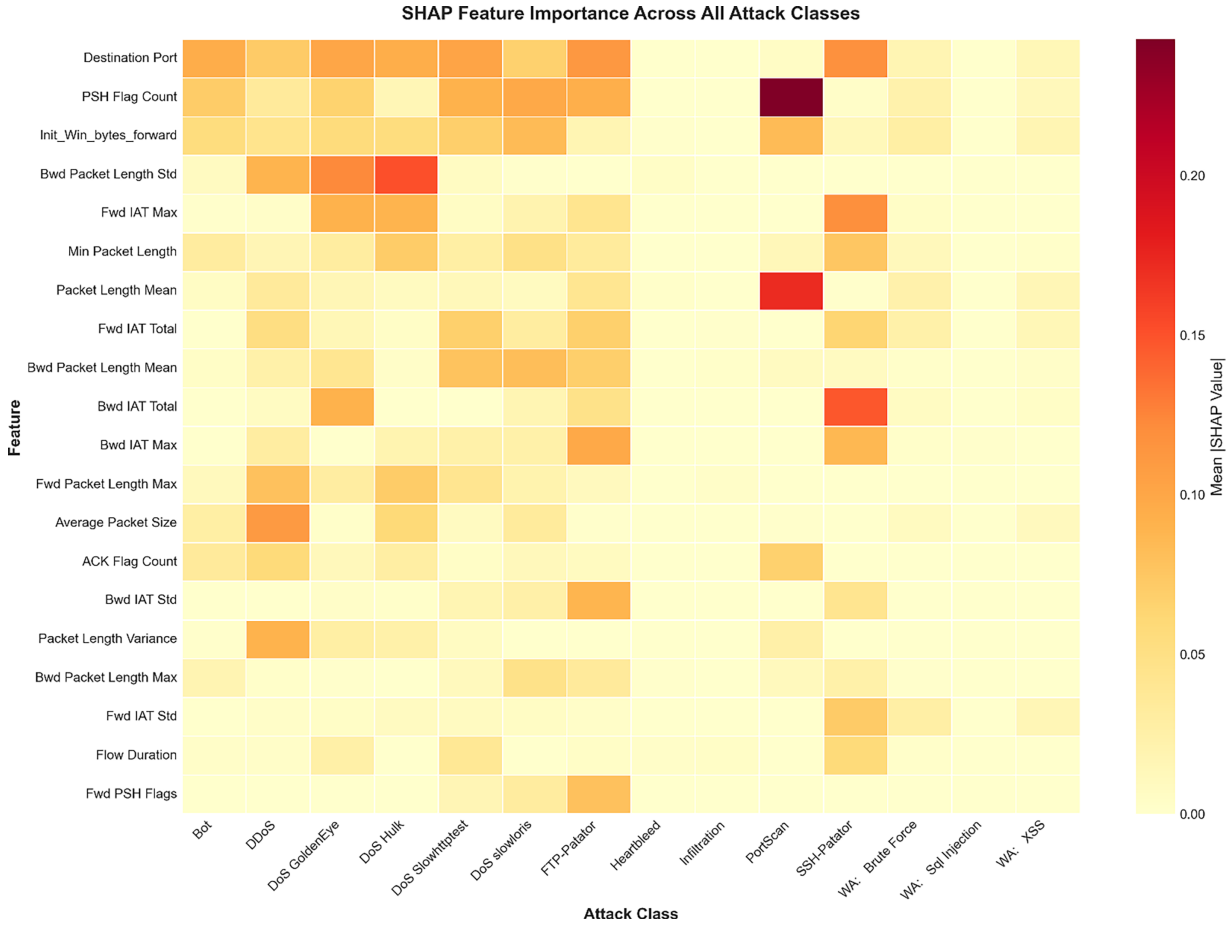

Figure 3: SHAP-based feature–attack importance heatmap illustrating class-specific feature relevance across all attack classes on CIC-IDS-2017 dataset.

Table 2: Top-ranked SHAP-identified critical features per attack class on CIC-IDS-2017 dataset.

Attack Class	Feature 1	Feature 2	Feature 3	Feature 4
Bot	Destination Port	Push (PSH) Flag Count	Init_Win_bytes _forward	Acknowledgment (ACK) Flag Count
DDoS	Average Packet Size	Packet Length Variance	Bwd Packet Length Std	Fwd Packet Length Max
DoS GoldenEye	Bwd Packet Length Std	Destination Port	Bwd IAT Total	Fwd IAT Max
DoS Hulk	Bwd Packet Length Std	Destination Port	Fwd IAT Max	Fwd Packet Length Max
DoS Slowhttptest	Destination Port	PSH Flag Count	Bwd Packet Length Mean	Idle Max

(Continued)

Table 2 (continued)

Attack Class	Feature 1	Feature 2	Feature 3	Feature 4
DoS slowloris	PSH Flag Count	Init_Win_bytes _forward	Bwd Packet Length Mean	Destination Port
FTP-Patator	Destination Port	Bwd Inter-Arrival Time (IAT) Max	PSH Flag Count	Bwd IAT Std
Heartbleed	Subflow Fwd Packets	Bwd Packet Length Std	Flow Duration	Packet Length Mean
Infiltration	Flow Duration	Subflow Fwd Packets	Fwd Packet Length Max	Total Length of Fwd Packets
PortScan	PSH Flag Count	Packet Length Mean	Bwd Packets/s	Init_Win_bytes _forward
SSH-Patator	Bwd IAT Total	Fwd IAT Max	Destination Port	Bwd IAT Max
Brute Force	Init_Win_bytes _backward	Init_Win_bytes _forward	Fwd IAT Std	Fwd IAT Total
SQL Injection	Bwd IAT Max	Min Packet Length	Flow IAT Max	Bwd Packet Length Std
XSS	Init_Win_bytes _backward	Init_Win_bytes _forward	Packet Length Mean	Fwd IAT Std

3.3 Stage 2: Knowledge-Guided Training

The core innovation of our framework is a novel loss function used to train a standard CNN. A typical model is trained using only the Categorical Cross-Entropy (L_{CCE}) loss, which measures predictive correctness, as defined in Eq. (1).

$$L_{CCE} = -\sum_{i=1}^{k} y_j \log(\hat{y}_j) \tag{1}$$

As discussed, L_{CCE} inherently favors majority classes in imbalanced datasets, as minimizing the overall loss is more easily achieved by correctly classifying the dominant BENIGN class. Consequently, gradients from minority (rare) attack classes contribute less to the total loss, leading the model to underfit or overlook them.

The total loss is defined as the sum of the conventional categorical cross-entropy and a knowledge-guided regularization term, $L_{Knowledge}$, which mitigates bias toward globally dominant features and enhances rare-attack detection by dynamically enforcing focus on SHAP-derived, class-specific critical features for each attack type. The regularization term $L_{Knowledge}$ depends on the knowledge vector corresponding to the attack class of the sample for which the loss is being calculated.

The total loss function and its knowledge-guided regularization component are defined in Eqs. (2) and (3), respectively.

$$L_{total} = L_{CCE} + \lambda \cdot L_{Knowledge} \tag{2}$$

$$L_{Knowledge} = \sum_{j=1}^{n}(1 - w_j)\left(\frac{\partial L_{CCE}}{\partial x_j}\right)^2 \quad (3)$$

where the terms are defined as:

- $\boldsymbol{w_j}$ is the SHAP-derived expert importance for feature *j* for the attack type belonging to the true sample
- $\frac{\partial L_{CCE}}{\partial x_j}$ is the input gradient, which measures the sensitivity of the model's error to a change in input feature *j*, which quantifies how much the model is relying on this feature for its prediction.

The gradient term is squared to ensure that both positive and negative sensitivities contribute equally to the penalty, emphasizing overall reliance magnitude rather than direction.

Mechanism of Action:

This loss function creates a dynamic and knowledge-based penalty:

- **If a feature *j* is unimportant** $(\boldsymbol{w_j = 0})$ **:** The $(\boldsymbol{1 - w_j})$ term becomes 1. The model receives the full penalty for relying on this irrelevant feature.
- **If a feature *j* is important** $(\boldsymbol{w_j = 1})$ **:** The $(\boldsymbol{1 - w_j})$ term becomes 0. The penalty is zeroed out. The model is allowed to rely on this feature.

This total loss function introduces a dual optimization objective: the model is designed to minimize classification error while placing greater emphasis on SHAP-identified, class-relevant features for each sample's true class. By penalizing reliance on globally dominant but class-irrelevant features, $\boldsymbol{L_{Knowledge}}$ reduces the influence of misleading global patterns, particularly when analyzing rare attacks that depend on subtle, locally important signals. As a result, this mechanism encourages the model to reduce noise and focus on class-specific feature relationships, leading to improved detection performance on minority attack classes.

The effectiveness of the proposed Knowledge-guided framework is evaluated in the following section.

4 Experimental Setup and Results

This section describes the computational environment, experimental configuration, model architecture, and evaluation procedure used to assess the proposed knowledge-guided framework.

4.1 Experimental Setup

Hardware:

All experiments were conducted on Google Colab using an NVIDIA T4 GPU with 16 GB of GPU memory.

Software:

The experiments were implemented in Python 3.10 using TensorFlow 2.17, Scikit-learn 1.3.2, Pandas 2.2.2, NumPy 1.26.4, and Matplotlib 3.8.4, with GPU acceleration enabled.

Model Architecture:

Both the baseline and Knowledge-guided models employed the same standard CNN architecture using TensorFlow Keras.

- Input Layer (shape = (78))
- Reshape Layer (target_shape = (78, 1))
- Conv1D (filters = 32, kernel_size = 3, activation = 'relu', padding = 'same')
- MaxPooling1D (pool_size = 2)
- Conv1D (filters = 64, kernel_size = 3, activation = 'relu', padding = 'same')

- MaxPooling1D (pool_size = 2)
- Flatten Layer
- Dense Layer (units = 128, activation = 'relu')
- Output Dense Layer (units = 15, activation = 'softmax')

It is important to note that the architectural components of the CNN remain unchanged; the proposed contribution is introduced exclusively at the loss-function level during training, where domain knowledge is integrated through gradient regularization.

Training Hyperparameters:

Baseline: Adam optimizer with learning rate 0.001, L_{CCE} loss (with label smoothing 0.05 added for evaluation comparability). Batch Size: 256. Trained for 10 epochs (implicit, weights loaded).

Knowledge-Guided:

- Phase 1 (2 epochs with $\lambda = 0$ from Eq. (2)) ensures the model first captures the *global feature structure* before the per-class SHAP regularization is applied.
- Phase 2 (8 epochs with $\lambda = 0.2$ from Eq. (2), selected based on sensitivity analysis; see Section 4.4) applies the per-class SHAP regularization to enhance the detection of minority attack patterns.

Dataset:

This work uses the CIC-IDS-2017 dataset as the primary dataset, which contains over 2.8 million network flows across 15 classes (1 benign and 14 attack types), each described by 78 features. As shown in Table 1, the dataset exhibits severe class imbalance.

In addition to CIC-IDS-2017, supplementary evaluations were conducted on the NSL-KDD and UNSW_NB15 benchmark datasets to assess the robustness of the proposed framework under different data distributions.

Unless otherwise stated, all detailed analyses, comparisons with state-of-the-art methods, and per-class evaluations are conducted on the CIC-IDS-2017 dataset, while NSL-KDD and UNSW_NB15 are used solely for supplementary sensitivity and robustness analysis.

Evaluation Metrics:

The performance of both the baseline and the proposed knowledge-guided model was assessed using accuracy, F1 score, and Macro F1 score. The equations for these metrics are as follows.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$F1\,Score = \frac{2 * (Precision * Recall)}{Precision + Recall}$$

$$MacroF1\,Score = \frac{1}{C}\sum_{i=0}^{c}$$

4.2 Baseline Model Performance: The Illusion of High Performance

An initial evaluation was performed on the standard CNN baseline. The results are shown in Table 3. The baseline model achieved 99.39% accuracy and a 0.9933 F1 score. These metrics, heavily weighted by the BENIGN class, are comparable to many SOTA results (See Table 4), suggesting a near-perfect model according to standard evaluation practices.

Table 3: Comparison of overall performance between the baseline CNN and the proposed Knowledge-Guided CNN.

Model	Accuracy	F1 Score	Macro F1 Score
Baseline CNN	99.39%	0.9933	0.6180
Proposed KG CNN	99.56%	0.9952	0.7560
Improvement	+0.170%	+0.0019	+0.138

Table 4: Comparison of accuracy and F1 score with State-of-the-Art methods.

Method	Accuracy	F1 Score
Bandarupalli [8] (DNN)	97.62%	0.6716
Ahmed et al. [4] (BMA Ensemble)	98.79%	0.9879
Ali et al. [3] (KNN)	99.88%	0.9746
Ali Afraji et al. [29] (CNN–LSTM–GRU)	99.49%	0.9949
Sharma [11] (CapsNet + BiLSTM)	99.00%	0.9800
Almuhanna and Dardouri [7] (DL/ML)	100%	1.0
Bouayad et al. [6] (Federated DL)	99.58%	0.9938
Talukder et al. [13] (RF)	99.95%	0.9995
Soumik [9] (Random Forest)	99.90%	0.9741
Our Baseline CNN	99.40%	0.9930
Our Proposed KG CNN	99.56%	0.9950

However, the Macro F1 score was only 0.6180. A detailed per-class analysis in Table 5 highlights a critical limitation masked by the high overall metrics: the baseline achieved an F1 score of 0.0 on Heartbleed and Infiltration. The model failed to detect these attack types, reinforcing the misleading nature of the high overall metrics.

Table 5: Per-class F1 score comparison (ordered by absolute improvement).

Class	Baseline F1 Score	Proposed KG F1 Score
Heartbleed	0	1
Infiltration	0	0.60
SSH-Patator	0.6823	0.9674
Web Attack—Brute Force	0.0935	0.2176
Web Attack—XSS	0.0000	0.0735
FTP-Patator	0.9906	0.9959
PortScan	0.9934	0.9959
DoS Hulk	0.9830	0.9865
DoS slowloris	0.9813	0.9875
DoS GoldenEye	0.9876	0.9878
BENIGN	0.9963	0.9973
DDoS	0.9984	0.9986
Bot	0.5802	0.5884
DoS Slowhttptest	0.9837	0.9436
Web Attack—SQL Injection	0.0000	0.0000

4.3 Knowledge-Guided Model Performance: Solving the Imbalance Problem

The proposed knowledge-guided framework, which uses the $L_{\text{Knowledge}}$ loss function, was then evaluated.

As shown in Table 3, the Knowledge-guided model achieved a Macro F1 score of 0.7560, representing a 13.8 percentage-point absolute gain over the baseline. Importantly, this substantial improvement in class-agnostic robustness was achieved while maintaining high overall accuracy (99.56%) and F1 score (0.9952) at levels comparable to the baseline and other state-of-the-art methods (see Table 4). These results indicate that the proposed approach improves minority class detection without sacrificing performance on standard evaluation metrics.

4.4 Sensitivity Analysis of the Knowledge-Guided Regularization Parameter λ

While $\lambda = 0.2$ was used in the main experiments, its selection was not arbitrary. To evaluate the robustness of the proposed framework and to analyze the impact of λ on model performance, a sensitivity analysis was conducted by training the model with multiple λ values: 0.01, 0.1, 0.2, and 0.3.

Table 6 reports the corresponding evaluation metrics. When λ is very small ($\lambda = 0.01$), the effect of the regularization term is weak, and the model's behavior remains close to that of the baseline CNN, resulting in limited improvement in Macro F1 score. Increasing λ to 0.1 yields a modest improvement, indicating that suppressing reliance on class-irrelevant features begins to positively affect minority-class detection. The best overall performance is achieved at $\lambda = 0.2$, which produces the highest accuracy, F1 score, and Macro F1 score. This value represents an effective balance between enforcing class-specific feature focus and preserving sufficient model flexibility. Further increasing λ to 0.3 leads to a decline in Macro F1 score, indicating the onset of over-regularization.

Table 6: Sensitivity analysis of the Knowledge-Guided regularization parameter λ on CIC-IDS-2017 dataset.

λ Value	Accuracy	F1 Score	Macro F1 Score
0.01	98.43	0.9839	0.7186
0.10	98.43	0.9841	0.7212
0.20	**99.56**	**0.9952**	**0.7560**
0.30	98.40	0.9836	0.7037

Note: Bold values indicate the best overall performance across the tested λ values on the CIC-IDS-2017 dataset.

The two-stage training schedule was designed to stabilize optimization when introducing the knowledge-guided regularization term. A short warm-up phase ($\lambda = 0$) allows the model to first learn global feature representations before applying class-specific constraints. Preliminary experiments with shorter warm-up durations (e.g., 1 epoch) led to unstable training, while longer warm-up phases reduced the impact of the knowledge-guided loss. Similarly, allocating fewer than 8 epochs to the knowledge-guided phase resulted in incomplete convergence, whereas extending this phase did not yield additional gains. Based on these observations, the 2 + 8 epoch schedule was selected as an effective and stable trade-off.

Table 4 compares the performance of the proposed and baseline models with state-of-the-art methods on the CIC-IDS-2017 dataset.

Table 5 presents the per-class F1 score comparison, illustrating the detailed contribution to the overall Macro F1 improvement. The knowledge-guided (KG) model, driven by the proposed loss function to learn class-specific patterns, achieved substantial gains on rare attack classes, **while maintaining consistent performance on majority classes**.

Table 6 presents the sensitivity analysis of the proposed knowledge-guided framework with respect to different values of the regularization parameter λ on CIC-IDS-2017 dataset.

Table 7 presents the sensitivity analysis of the proposed knowledge-guided framework with respect to different values of the regularization parameter λ on NSL-KDD Dataset.

Table 7: Sensitivity analysis of the Knowledge-Guided regularization parameter λ on NSL-KDD Dataset.

λ Value	Accuracy	F1 Score	Macro F1 Score
0 (Baseline)	75.84	0.5469	0.7296
0.01	75.95	0.5897	0.7362
0.10	75.92	0.5995	0.7351
0.20	76.45	0.5976	0.7400
0.30	**76.37**	**0.6159**	**0.7438**

Note: Bold values indicate the best overall performance across the tested λ values on the NSL-KDD dataset.

Table 8 presents the sensitivity analysis of the proposed Knowledge-guided framework with respect to different values of the regularization parameter λ on UNSW_NB15 Dataset.

Table 8: Sensitivity analysis of the Knowledge-Guided regularization parameter λ on UNSW_NB15.

λ Value	Accuracy	F1 Score	Macro F1 Score
0 (Baseline)	69.76	0.3883	0.6957
0.01	**72.90**	**0.4412**	**0.7129**
0.10	72.05	0.4368	0.7156
0.20	72.22	0.4364	0.7175
0.30	71.59	0.4305	0.7126

Note: Bold values indicate the best overall performance across the tested λ values on the UNSW_NB15 dataset.

Since imbalance-aware regularization controls the degree of feature concentration toward minority classes, its strength must be adapted to the dataset's imbalance structure; reusing the same λ across datasets can lead to excessive constraint in scenarios with different class distributions.

The following section discusses these results and provides further analysis.

5 Discussion

The results in Section 4 provide empirical evidence for the observed behavior of the baseline model. When trained solely with the conventional loss function (L_{CCE}), the model tends to converge toward a majority-driven local minimum. Because the dataset is dominated by BENIGN traffic (approximately 80%) and several high-frequency attack classes, the optimization process prioritizes these categories, resulting in high overall accuracy and F1 scores comparable to state-of-the-art reports. However, this behavior causes the model to rely heavily on globally dominant features that maximize aggregate performance but introduce noise when analyzing minority classes such as Heartbleed.

Focal loss and class-balanced loss are commonly used imbalance-aware objectives that address class imbalance by reweighting samples or classes during training. Focal loss emphasizes hard-to-classify samples, while class-balanced loss compensates for unequal class frequencies. However, these methods operate at the sample or class level and do not explicitly constrain which features the model relies on for each attack type.

The proposed gradient regularization term scales linearly with the number of input features, as it operates directly on input gradients. While this is computationally feasible for flow-based representations such as CIC-IDS-2017 with 78 features, higher-dimensional inputs or deep feature embeddings would increase computational cost. Future work will explore sparse regularization, feature grouping, or embedding-level constraints to extend the framework to higher-dimensional representations.

The knowledge-guided component of the loss function plays an important role during training by increasing the penalty whenever the model depends on features that SHAP analysis identifies as irrelevant for the corresponding class. This mechanism reduces the model's sensitivity to globally dominant but misleading features and directs learning toward patterns that are more strongly associated with minority attacks. The resulting improvement of 13.8 percentage points in Macro F1 score highlights the impact of this constraint on class-agnostic generalization. Achieving an F1 score of 1.0 for the Heartbleed class and a marked improvement for Infiltration further demonstrates the effectiveness of the proposed framework in extracting meaningful representations from highly underrepresented classes. Importantly, these gains are achieved without degrading performance on majority classes, as reflected by the accuracy and F1 values reported in Table 4.

Some limitations remain where the proposed knowledge-guided (KG) model does not show significant improvement. In particular, the model fails to detect Web Attack—SQL Injection and performs poorly on Web Attack—XSS and Web Attack—Brute Force, although the latter two classes show modest gains over the baseline F1 scores of 0.0, 0.0, and 0.0935, respectively (Table 5). This behavior is likely due to two compounding factors. First, Web Attack—SQL Injection and Web Attack—XSS are primarily characterized by malicious patterns embedded within application-layer payloads (e.g., SQL commands or injected JavaScript), which are not captured by the flow-based network features available in the CIC-IDS-2017 dataset. As a result, neither the baseline nor the KG model has access to the information required to reliably distinguish such payload-dependent attacks.

Second, the dataset contains extremely few samples for these classes (e.g., only 21 total samples for SQL Injection, 652 for XSS, and 1507 for Web Brute Force—**Table 1**). Our method relies on SHAP in Stage 1 to generate good knowledge vectors. If there are too few samples, SHAP analysis may struggle to identify statistically significant or stable feature importance patterns. Recent studies have confirmed that SHAP's stability and statistical significance can be compromised in scenarios with very small sample sizes or sparse data, as the explanations can fluctuate and may lack statistical power [26,27]. Consequently, the knowledge guide (w_j) for these classes might be weak or unreliable, limiting the effectiveness of the $L_{Knowledge}$ term during training.

Furthermore, it is important to note that the effectiveness of the proposed knowledge-guided framework is inherently dependent on the quality of the baseline model used to generate SHAP knowledge vectors. For the Bot class, the model saw no meaningful improvement (0.5802 to 0.5884 F1 score), even though this class (1966 samples) is not as resource-starved as the Web Attack classes. This highlights a key dependency of our method: the quality of the knowledge guide is contingent on the baseline model's performance from which SHAP values are derived. The baseline model itself performed poorly on the Bot class (0.58 F1 score), suggesting that it learned spurious or weak correlations. Consequently, the SHAP-derived knowledge generated in Stage 1 was likely suboptimal. Our $L_{Knowledge}$ term, working as intended, was then forced to learn from this ineffective guide, preventing any significant improvement over the already poor baseline.

This finding directly motivates our future work. The supplementary sensitivity analyses on NSL-KDD and UNSW_NB15 provide additional insight into the behavior of the proposed framework beyond a single dataset. On NSL-KDD, appropriate selection of the regularization strength leads to clear improvements in Macro F1 score relative to the baseline. On UNSW_NB15, the proposed method yields more moderate gains,

suggesting that the injected knowledge provides partial guidance in settings where baseline representations remain limited. Together, these results demonstrate that the proposed approach remains stable across datasets and that its effectiveness depends on the alignment between the regularization strength, the baseline model, and the underlying data distribution. It suggests a need for a more robust knowledge source, such as an LLM fine-tuned on cybersecurity data, which is not solely dependent on a potentially flawed baseline model's performance on certain classes. Such external knowledge could provide a stronger signal for the $L_{Knowledge}$ term, especially for classes where the baseline (and thus SHAP) struggles.

The next section summarizes the main findings of this study and outlines directions for future work.

6 Conclusion and Future Work

This paper addresses a key limitation in modern Network Intrusion Detection Systems (NIDS), the reliance on static optimization strategies that can lead to misleadingly high performance on imbalanced datasets. To mitigate this issue, we introduced a Domain knowledge-guided Attentional Training framework that integrates SHAP-derived knowledge with a custom gradient regularization loss function. This design reduces reliance on globally dominant but class-irrelevant features and encourages the model to focus on class-specific, informative patterns relevant to each attack type.

Experimental results demonstrate the effectiveness of the proposed framework, yielding a 13.8 percentage-point improvement in the Macro F1 score and achieving reliable detection of rare attacks such as Heartbleed, while maintaining competitive overall accuracy and F1 scores compared to state-of-the-art methods. These findings suggest that incorporating class-specific knowledge into the training process can improve robustness and generalization in highly imbalanced intrusion detection scenarios.

For future work, this framework could be extended to other security domains characterized by severe class imbalance, such as malware analysis. In addition, exploring alternative sources of domain knowledge represents a promising direction. In particular, Large Language Models (LLMs), potentially fine-tuned on specialized cybersecurity reports, could serve as a complementary, non-empirical source of expert knowledge to guide learning, helping to address the limitations of SHAP in extremely low-sample settings.

Acknowledgement: The authors would like to express their deepest gratitude to their academic supervisors, Prof. Dr. Nahla Belal and Dr. Mohamed Seifeldin, for their invaluable guidance, expertise, and unwavering support throughout this research.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: Zakaria S. M. Abdelhalim conceived and designed the study; developed the methodology; implemented the proposed models; performed data preprocessing, experiments, and analysis; and wrote the original draft of the manuscript. Nahla Belal provided academic supervision, methodological guidance, and critical review of the manuscript. Mohamed Seifeldin provided academic supervision, methodological guidance, and critical review of the manuscript. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The dataset used in this study is the publicly available **CIC-IDS-2017** dataset, which can be obtained from the Canadian Institute for Cybersecurity. No new data were generated during this study.

Ethics Approval: This study does not involve human participants, animals, or personal data, and therefore does not require ethical approval.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Sajid M, Malik KR, Almogren A, Malik TS, Khan AH, Tanveer J, et al. Enhancing intrusion detection: a hybrid machine and deep learning approach. J Cloud Comput. 2024;13(1):123. doi:10.1186/s13677-024-00685-x.
2. Bella K, Guezzaz A, Benkirane S, Azrour M, Fouad Y, Benyeogor MS, et al. An efficient intrusion detection system for IoT security using CNN decision forest. PeerJ Comput Sci. 2024;10:e2290. doi:10.7717/peerj-cs.2290.
3. Ali ML, Thakur K, Schmeelk S, Debello J, Dragos D. Deep learning vs. machine learning for intrusion detection in computer networks: a comparative study. Appl Sci. 2025;15(4):1903. doi:10.3390/app15041903.
4. Ahmed U, Zheng J, Almogren A, Khan S, Sadiq MT, Altameem A, et al. Explainable AI-based innovative hybrid ensemble model for intrusion detection. J Cloud Comp. 2024;13(1):150. doi:10.1186/s13677-024-00712-x.
5. Abdou Vadhil F, Lemine Salihi M, Farouk Nanne M. Machine learning-based intrusion detection system for detecting web attacks. IAES Int J Artif Intell. 2024;13(1):711. doi:10.11591/ijai.v13.i1.pp711-721.
6. Bouayad A, Alami H, Janati Idrissi M, Berrada I. Lightweight federated learning for efficient network intrusion detection. IEEE Access. 2024;12:172027–45. doi:10.1109/ACCESS.2024.3494057.
7. Almuhanna R, Dardouri S. A deep learning/machine learning approach for anomaly based network intrusion detection. Front Artif Intell. 2025;8:1625891. doi:10.3389/frai.2025.1625891.
8. Bandarupalli G. Efficient deep neural network for intrusion detection using CIC-IDS-2017 dataset. In: 2025 First International Conference on Advances in Computer Science, Electrical, Electronics, and Communication Technologies (CE2CT); 2025 Feb 21–22; Nainital, India. p. 476–80. doi:10.1109/CE2CT64011.2025.10940012.
9. Soumik MS. A comparative analysis of Network Intrusion Detection (NID) using Artificial Intelligence techniques for increase network security. Int J Sci Res Arch. 2024;13(2):4014–25. doi:10.30574/ijsra.2024.13.2.2664.
10. Saidane S, Telch F, Shahin K, Granelli F. Optimizing intrusion detection system performance through synergistic hyperparameter tuning and advanced data processing. In: Computer science, engineering and information technology. Chennai, India: Academy & Industry Research Collaboration Center; 2024. p. 141–60. doi:10.5121/csit.2024.141411.
11. Sharma V. Improving intrusion detection with hybrid deep learning models: a study on CIC-IDS2017,UNSW-NB15, and KDD CUP 99. J Inf Syst Eng Manag. 2025;10(11s):633–50. doi:10.52783/jisem.v10i11s.1665.
12. Li L, Lu Y, Yang G, Yan X. End-to-end network intrusion detection based on contrastive learning. Sensors. 2024;24(7):2122. doi:10.3390/s24072122.
13. Talukder MA, Islam MM, Uddin MA, Hasan KF, Sharmin S, Alyami SA, et al. Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction. J Big Data. 2024;11(1):33. doi:10.1186/s40537-024-00886-w.
14. Mia N, Haque Gazi MM, Ahmed Nabin J, Tamim FS, Mohammad S, Ul Islam MR. A novel temporal-aware adaptive feature selection strategy for network intrusion detection systems. In: 2025 Global Conference in Emerging Technology (GINOTECH); 2025 May 9–11; Pune, India. p. 1–6. doi:10.1109/GINOTECH63460.2025.11076711.
15. Ghanem M, Ghaith AK, El-Hajj VG, Bhandarkar A, de Giorgio A, Elmi-Terander A, et al. Limitations in evaluating machine learning models for imbalanced binary outcome classification in spine surgery: a systematic review. Brain Sci. 2023;13(12):1723. doi:10.3390/brainsci13121723.
16. Albattah W, Khan RU. Impact of imbalanced features on large datasets. Front Big Data. 2025;8:1455442. doi:10.3389/fdata.2025.1455442.
17. Altalhan M, Algarni A, Turki-Hadj Alouane M. Imbalanced data problem in machine learning: a review. IEEE Access. 2025;13(1):13686–99. doi:10.1109/ACCESS.2025.3531662.
18. Medvedieva K, Tosi T, Barbierato E, Gatti A. Balancing the scale: data augmentation techniques for improved supervised learning in cyberattack detection. Eng. 2024;5(3):2170–205. doi:10.3390/eng5030114.
19. Shanmugam V, Razavi-Far R, Hallaji E. Addressing class imbalance in intrusion detection: a comprehensive evaluation of machine learning approaches. Electronics. 2025;14(1):69. doi:10.3390/electronics14010069.
20. Rao YN, Suresh Babu K. An imbalanced generative adversarial network-based approach for network intrusion detection in an imbalanced dataset. Sensors. 2023;23(1):550. doi:10.3390/s23010550.

21. Allagi S, Pawan T, Leong WY. Enhanced intrusion detection using conditional-tabular-generative-adversarial-network-augmented data and a convolutional neural network: a robust approach to addressing imbalanced cybersecurity datasets. Mathematics. 2025;13(12):1923. doi:10.3390/math13121923.
22. Chinnasamy R, Subramanian M, Sengupta N. Empowering intrusion detection systems: a synergistic hybrid approach with optimization and deep learning techniques for network security. Int Arab J Inf Technol. 2025;22(1):60–76. doi:10.34028/iajit/22/1/6.
23. Arreche O, Guntur T, Abdallah M. XAI-IDS: toward proposing an explainable artificial intelligence framework for enhancing network intrusion detection systems. Appl Sci. 2024;14(10):4170. doi:10.3390/app14104170.
24. Chen X, Liu M, Wang Z, Wang Y. Explainable deep learning-based feature selection and intrusion detection method on the Internet of Things. Sensors. 2024;24(16):5223. doi:10.3390/s24165223.
25. Yang Y, Khorshidi HA, Aickelin U. A review on over-sampling techniques in classification of multi-class imbalanced datasets: insights for medical problems. Front Digit Health. 2024;6:1430245. doi:10.3389/fdgth.2024.1430245.
26. Lakshya V, Sai Sri H, Subham J, Aju D. Unravelling complexity: investigating the effectiveness of SHAP algorithm for improving explainability in network intrusion system across machine and deep learning models. Int J Perform Eng. 2024;20(7):421. doi:10.23940/ijpe.24.07.p2.421431.
27. Gaspar D, Silva P, Silva C. Explainable AI for intrusion detection systems: LIME and SHAP applicability on multi-layer perceptron. IEEE Access. 2024;12:30164–75. doi:10.1109/ACCESS.2024.3368377.
28. Alkhawaldeh IM, Albalkhi I, Naswhan AJ. Challenges and limitations of synthetic minority oversampling techniques in machine learning. World J Methodol. 2023;13(5):373–8. doi:10.5662/wjm.v13.i5.373.
29. Ali Afraji DMA, Lloret J, Peñalver L. An integrated hybrid deep learning framework for intrusion detection in IoT and IIoT networks using CNN-LSTM-GRU architecture. Computation. 2025;13(9):222. doi:10.3390/computation13090222.