Towards Robust Intrusion Detection: Exploring Feature Selection, Balancing Strategies, and Deep Learning for Minority Class Optimization

Khalid LABHALLA, Amal BATTOU
Faculty of Science-IRF-SIC Laboratory, Ibn Zohr University, Agadir, Morocco

Abstract—**The increasing connectivity of systems and the rapid growth of the Internet have intensified cybersecurity threats. It has been demonstrated that conventional signature-based intrusion detection methods are deficient, especially against Zero-Day attacks. An alternative approach involves the deployment of Intrusion Detection Systems (IDS) that are based on deep learning algorithms. However, these systems face a significant challenge in detecting minority classes of attacks, such as Remote-to-Local (R2L) and User-to-Root (U2R) attacks, which, although rare, are of critical importance. Misclassifying these attacks is costly. Therefore, the reduction of false negatives is achieved by coupling feature selection techniques (Chi square, correlation, information Gain, Extreme Gradient Boosting (XGBoost), Autoencoder), oversampling methods (Synthetic Minority Oversampling Technique (SMOTE), Adaptive Synthetic Sampling (ADASYN)) and deep learning models (Deep Neural Network (DNN), Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM) and hybrid model CNN LSTM). The present study uses the NSL-KDD dataset, with a particular focus on the minority classes R2L, which represents 2.61% of the dataset, and U2R, representing 0.08% of the dataset. The findings indicate that data balancing is paramount. ADASYN facilitates 100% U2R detection, while SMOTE enhances R2L accuracy to above 95%. The application of correlation and autoencoder feature selection techniques proved to be the most effective. The effectiveness of CNN models in addressing U2R classification tasks has been extensively demonstrated, while the use of DNN or CNN-LSTM models has been shown to yield optimal results for R2L tasks. DNN remains the most stable model overall. For the two minority classes, the most effective pipelines are Correlation + SMOTE + DNN, achieving 93.84 % recall for U2R and 99.88 % for R2L, and Autoencoder + SMOTE + CNN-LSTM, achieving 89.66 % recall for R2L and 99.68 % for U2R.**

Keywords—Network intrusion detection system; imbalanced data; minority class detection; deep learning; feature selection; balancing techniques

I. Introduction

The Internet connects most of our computer systems, and our social life today is deeply linked to the Internet, which increases security threats in various ways. Targets range from financial platforms and e-commerce or governmental institutions to major corporations, attacked for economic gain or ideological motives. Cisco projects that the number of Distributed Denial of Service (DoS) [1] incidents will reach 15.4 million in 2023 [2]. Companies are expected to devote nearly 6.69 billion USD to cloud security in 2023, an increase of roughly 27 % over the previous year. According to Gartner's reports, published at the end of 2024, global spending on information security is expected to reach 183 billion USD, and then to grow steadily at a compound annual growth rate (CAGR) of 11.7% between 2023 and 2028 [3]. The costs of damage caused by cybercrime are expected to rise from 3 trillion USD in 2015 to 10.5 trillion USD in 2025 [4]. In this context, and to improve intrusion detection systems (IDS), several approaches have been developed, notably machine learning-based approaches, in order to address the shortcomings of signature-based IDS, which are very limited against zero-day attacks [5].

Among data-driven approaches, deep learning has emerged as a powerful tool for IDS due to its capacity to model complex, nonlinear relationships in high-dimensional data. Various deep learning architectures, including Deep Neural Network (DNN), Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and a hybrid CNN-LSTM model, have shown promising results in identifying sophisticated attack vectors. However, one of the most pressing and unresolved challenges in deep learning-based IDS is the class imbalance problem, the training data for the majority class significantly outnumbers that of the minority class [6]. Well-known IDS datasets such as NSL-KDD inherit this imbalance issue from their predecessors. Specifically, the Remote-to-Local (R2L) and User-to-Root (U2R) attack categories constitute a very small fraction of the total samples, accounting for only 2.61% and 0.08% of the dataset, respectively. In contrast, classes such as normal connections and DoS attacks dominate the dataset. This severe imbalance leads to biased learning where the models tend to favor majority classes, resulting in high false negative rates for minority classes. These false negatives are particularly concerning because they correspond to stealthy and potentially severe intrusions that evade detection. Several techniques have been proposed to address class imbalance in IDS, including oversampling or undersampling methods, cost-sensitive learning, deep learning models, or hybrid methods. Among resampling techniques, Synthetic Minority Oversampling Technique (SMOTE) and Adaptive Synthetic Sampling (ADASYN) are two of the most commonly used methods for generating synthetic examples of minority class instances. These methods aim to rebalance the dataset distribution and enhance the learning capability of classifiers for rare attack types.

Another key component influencing detection performance is feature selection. Effective feature selection helps reduce the dimensionality of input data, eliminates irrelevant or redundant features, and highlights the most informative attributes. This is especially crucial in IDS, where the presence of noisy or nondiscriminative features can degrade the model's ability to detect subtle attacks. This work employs a dual strategy that incorporates both statistical and machine learning-based feature selection techniques. The statistical methods include the Chi-square test, Pearson correlation, and Information Gain, all of which evaluate feature relevance based on individual relationships with the target class. On the other hand, Extreme Gradient Boosting (XGBoost) is used to rank features based on their predictive contribution within tree-based models, while an Autoencoder, a neural network designed for unsupervised representation learning, is used to extract low-dimensional embeddings that preserve the most salient data characteristics. In this study, a comprehensive framework is presented to explore how different combinations of feature selection techniques, data balancing methods, and deep learning architectures impact the detection performance of minority classes in IDS.

Our models will be evaluated using the NSL-KDD dataset, and recall will be chosen as the primary evaluation metric, as the main objective is to improve intrusion detection performance by reducing false negatives for minority classes while maintaining a high detection rate for majority classes. In addition, other evaluation metrics such as F1 score and precision will also be measured to provide a more comprehensive assessment of model performance. The proposed models were evaluated and compared with approaches from recent studies. In order to determine the most effective combination of key components in the design of an intrusion detection system (IDS), namely the feature selection method, class balancing technique, and learning algorithm, a total of 72 different models were evaluated. Our contributions can be summarized as follows:

- Systematic evaluation of feature selection techniques: Three statistical methods are compared (Chi-square, Pearson correlation, and Information Gain) with two machine learning-based approaches (XGBoost and Autoencoder) to analyze their influence on minority class detection in NSL-KDD.

- Investigation of data balancing methods: Both SMOTE and ADASYN are applied to address the class imbalance and evaluate their impact across different DL models.

- Comparative analysis of deep learning models: The effectiveness of four deep architectures: DNN, CNN, LSTM, and CNN-LSTM is addressed under different feature and data configurations.

- Optimization of detection pipelines for minority classes: Optimal combinations are identified for each minority class. For U2R, several combinations achieved perfect detection performance (100% recall). For R2L, the combination Correlation + SMOTE + DNN yields over 96.58% accuracy with high stability.

Despite significant advances in deep learning-based intrusion detection systems, most existing work focuses on performance improvements achieved through combinations of feature selection techniques, class rebalancing, and deep learning models. However, these approaches suffer from poor generalizability of results, which are often evaluated on a limited number of configurations or metrics. In particular, few studies offer an in-depth and reproducible analysis to identify robust and consistent pipelines for the reliable detection of critical minority classes, such as R2L and U2R, in our case study.

The experiments were conducted on the NSL-KDD dataset, widely used in IDS literature due to its balanced structure for comparative evaluation and the presence of highly minority attack classes such as R2L and U2R. This dataset is a relevant test bed for analyzing the effectiveness of rebalancing techniques and deep learning models in the face of severe imbalances. Its use also allows for direct comparison with numerous previous studies. In this study, the macro-average metric was adopted as the primary measure of overall performance. This choice was motivated by the highly imbalanced nature of the NSL-KDD dataset, where majority classes such as Normal and DoS dominate, while minority classes such as R2L and U2R represent a very small portion of the data.

The rest of this study is organized as follows: Section II reviews related work on intrusion detection systems and concludes with a comparative analysis of the proposed model and recent studies. Section III provides a detailed scenario of our approach. Section IV draws result and discussion. Finally, Section V presents a conclusion.

II. Related Works

The concept of intrusion detection systems has been extensively explored in the literature that reflects both the complexity of the subject and the diversity of methodological approaches explored. Research efforts have primarily focused on two critical dimensions: enhancing detection accuracy by reducing false positives, and enabling real-time intrusion detection, particularly within large-scale data environments [7]. In this context, the following section provides an overview of existing work that leverages machine learning techniques to support real-time processing capabilities.

The authors in [8] proposed the CWFLAM-VAE architecture by integrating Class-Wise Focal Loss, Extreme Gradient Boosting, and a Variational Autoencoder. The framework synthesizes rare-class attack samples while faithfully maintaining the original feature distributions. It was tested on NSL-KDD and CSE-CIC-IDS2018 datasets, and it reached F-scores of 97.6% and 98.1%, while [9] proposes a three-layer approach to improve the detection of minority attacks in intrusion detection systems. The first layer uses a weighted deep neural network (WDNN) to detect suspicious traffic. The second layer employs a CNN and an LSTM to classify attacks as majority or minority. Finally, the third layer applies XGBoost to refine the classification of minority attacks. Undersampling (unilateral selection) and oversampling (ADASYN) optimize class balance. The system achieves an accuracy of over 97.9% on the NSL-KDD dataset. Undersampling (unilateral selection) and oversampling (ADASYN) optimize class balance. The system achieves an accuracy higher than 97.9% on the NSL-KDD dataset. The work in [10] overcame the limitations of traditional methods in detecting minority class attacks by a multimodal approach based on deep learning and GANs to generate high-quality attack samples. A specialized model then learns their features, and an integrated classifier performs multi-class classification. Tested on CICIDS2017 and NSL-KDD, this method achieves up to 99.96% accuracy with a low false positive rate (3.4%) on the NSL-KDD dataset. Another method proposed in the article [11] proposed a feature selection method (feature ranking) based on the correlation and entropy (information gain) of each feature. After ranking, the features are partitioned into three subsets for each method. Information-Gain attributes form IG-1, IG-2, and IG-3, while correlation-based attributes form CR-1, CR-2, and CR-3. IG-1 and CR-1 contain the ten highest-ranked features (positions 1–10); IG-2 and CR-2 contain those ranked 11–30; and IG-3 and CR-3 contain the remainder. A new feature set is created by combining IG-1 and CR-1, and by intersecting IG-2 and CR-2, while discarding all features in IG-3 and CR-3. The resulting selection is evaluated on five datasets, with both per-dataset and average results reported. The proposed model improves recall, increasing it from 81% without feature selection to 86% with the proposed method for the minority class U2R. A network intrusion detection system based on deep learning and using a chaotic optimization strategy proposed by the authors in [12], after preprocessing and balancing the dataset using the Extended Synthetic Sampling method, features were extracted from the dataset using kernel-assisted principal component analysis. The Chaotic Honey Badger Optimizer first identified the most informative feature. These selected features were then fed into the gated-attention dual-LSTM (Dugat-LSTM) model to classify the attacks, achieving a recall of 98.76% on the NSL-KDD dataset.

The work in [13] tried to improve the IDS by proposing a system based on two phases of classification. The first phase consists of using three variants of the Naive Bayes classification method (categorical, Bernoulli, and Gaussian) for each type of data (nominal data, binary data, and real or integer data), respectively. The second phase of classification consists of using unsupervised elliptic envelope classification to predict whether the behavior is normal or anomalous. The elliptic envelope is a machine learning method for anomaly detection. It is based on modeling the data distribution using the theory of the elliptical envelope. The goal is to identify data points that are distant from the main distribution, assuming that the latter follows an elliptical distribution. This method is particularly useful for detecting anomalies in multivariate data, i.e., data with multiple features. After the second phase, they obtained an accuracy of 97% on the NSL-KDD dataset, 86.9% on the UNSW_NB15 dataset, and 98.59% on the CICIDS2017 dataset. However, accuracy for the minority classes remains modest; for example, on the NSL-KDD dataset, the U2R and R2L classes achieved precisions of 40.29% and 58.4%, respectively. The comparison of authors in [14] showed that synthetic oversampling SMOTE with the Random Forests model enhances R2L and U2R detection on NSL-KDD.

Another way to improve the detection of minority classes is proposed in [15] by applying two oversampling and two undersampling techniques to balance the dataset. Five machine learning models, including XGBoost and CatBoost, are evaluated using grid search and 10-fold cross-validation. The results confirm that resampling improves classification performance across models. Among them, XGBoost with SMOTE achieves the best results, with an accuracy of 75% and a weighted F1-score of 78%.

In [16], the authors combine ADASYN oversampling to address class imbalance with the LightGBM model. Experiments on NSL-KDD, UNSW-NB15, and CICIDS2017 demonstrate improved detection rates for minority attacks. The method achieves high accuracy 92.57%, 89.56%, and 99.91% in the three test sets. Improving the minority class in poorly balanced datasets was also a challenge for the authors in [17], who implemented a deep neural network trained and tested on the CICIDS-2017 and CICIDS-2018 datasets. Chi-square and correlation are two proposed statistical methods for feature selection methods statistical proposed. The findings indicate that certain coarse-grained features are highly discriminative, enabling the complete and accurate detection of attacks represented by as few as three instances. In reference [18] proposed to improve the detection for minority classes by a hybrid approach. This approach is a combination of SMOTE) technique and Tomek's links for reducing noise. Furthermore, to strengthen the performance of the intrusion detection system, the study harnesses two deep learning architectures, LSTM networks and CNNs. The approach was evaluated on the NSL-KDD and CICIDS-2017 benchmarks. For NSL-KDD multiclass classification, the LSTM configuration achieved 99.57 % accuracy and a 98.98 % F-score, while the CNN reached 99.70 % accuracy and 99.27 % F-score. On CICIDS-2017, the LSTM attained 99.82 % accuracy with a 98.65 % F-score, and the CNN obtained 99.85 % accuracy and a 98.98 % F-score. Authors in [19] focused on improving IDS detection rates by combining advanced t-SNE for feature extraction with intelligent classification methods. A hybrid model is proposed, integrating Genetic Fuzzy Systems (GFS) with Generative Adversarial Networks (GANs) in a paired learning framework. The proposed model was evaluated on TII-SSRC-23 and NSL-KDD datasets, attaining detection accuracy of 99.23% and 99.13%, respectively, to identify the most informative attributes.

Table I shows that our methodology, which systematically searches for the optimal combination of leading feature selection techniques, data-balancing strategies, and deep-learning algorithms, outperforms previous work by a considerable margin. The results further reveal that correlation analysis is the most stable and universally suitable statistical method, delivering strong performance irrespective of the balancing technique or model used. The DNN architecture appears in several of the best-performing combinations, highlighting its importance for this kind of problem. Finally, the autoencoder remains a particularly robust option, especially when paired with an LSTM model or the hybrid CNN-LSTM architecture.

TABLE I. COMPARES RESULTS FROM THE NSL KDD DATASETS TO OTHER ADVANCED TECHNIQUES

Work	R2L Recall (Value in %)	U2R Recall (Value in %)
Akashdeep et al., 2017 [11]	91.9	86.6
Bedi et al., 2021 [20]	32	50
Gupta et al., 2022 [21]	55	54
Ahmad et al., 2022 [22]	84	52
Vishwakarma and Kesswani, 2023 [13]	58	40
Harini et al., 2023 [9]	88	65
Xu et al., 2024 [23]	100	99
Long et al., 2025 [24]	99.6	100
Abdulganiyu et al., 2025 [8]	93.61	92.47
Xue et al., 2025 [25]	89	55
Best pipelines in proposed work		
Correlation + SMOTE + DNN pipeline	93.84	99.88
Autoencoder + SMOTE + CNN-LSTM pipeline	89.66	99.68
Correlation + ADASYN + DNN pipeline	86.64	100
AutoEncoder + ADASYN + DNN pipeline	85.88	98.49

III. PROPOSED METHODOLOGY

The aim of this study is to identify the best possible combination of feature selection algorithms, class balancing techniques, and deep learning models to improve the performance of intrusion detection systems (IDS). The main objective is to reduce the false negative rate, particularly for minority classes such as R2L and U2R, which are often poorly detected in conventional approaches. My approach is based on a systematic analysis of several pipelines combining different selection methods (such as Chi2, Correlation, Information Gain, or Autoencoder), several balancing strategies (such as SMOTE and ADASYN), and various deep learning models (DNN, CNN, LSTM, or CNN-LSTM). By evaluating these combinations according to precise metrics such as recall, which is the most relevant indicator for assessing the reduction of false negatives, the study aims to determine the most effective synergies for strengthening the detection of rare attacks and minimizing critical errors that compromise the reliability of modern IDSs.

To improve the detection of minority classes and reduce false negatives in an intrusion detection system, the study proposes a methodology based on five main phases:

- Data Pre-processing
- Feature Selection
- Class Rebalancing
- Applied Deep Learning Models
- Performance Evaluation

As shown in Fig. 1, first, the dataset is downloaded and prepared by removing duplicates and outliers and handling missing values. The data is then converted to hot encoding and normalized. This pre-processing step ensures a clean, consistent, and usable basis for downstream methods. The second phase focuses on feature selection, in which statistical approaches are applied. such as chi-2, correlation, and information gain, and approaches based on supervised learning, such as XGBoost, and unsupervised learning, such as Autoencoder for non-linear reduction with a dual objective: to reduce the dimension to limit overfitting and improve model efficiency, while retaining the relevant signal for rare classes. This phase produces five variants derived from the dataset, each corresponding to a different selection technique.

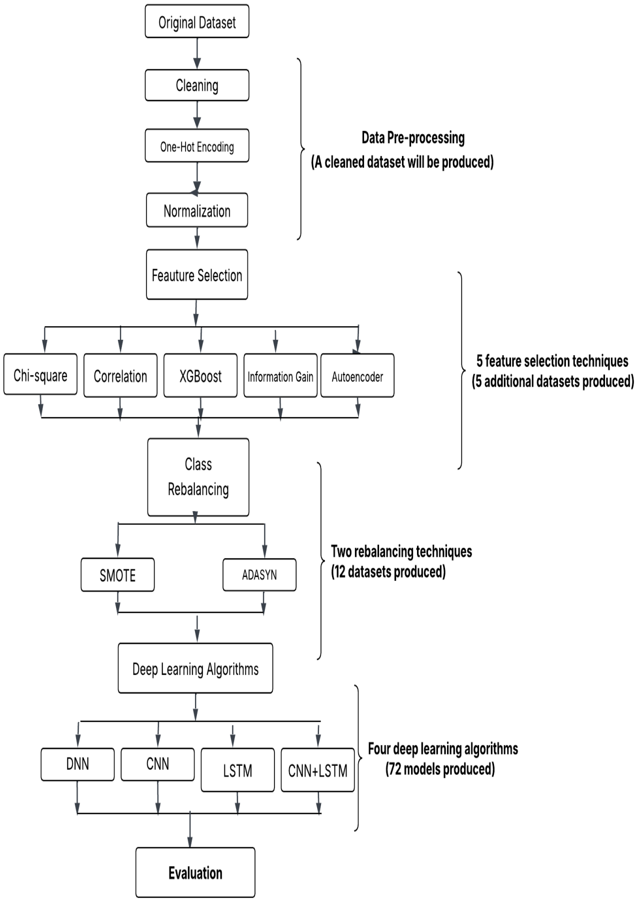

Fig. 1. Proposed methodology.

Then, faced with class imbalance, particularly U2R, which accounts for only 0.08% of the dataset, and R2L, which accounts for only 2.61%, resampling techniques such as SMOTE and ADASYN are applied in conjunction with each feature selection technique in order to identify the optimal balancing strategy and the feature selection methods that optimize it. At the end of this stage, twelve additional dataset variants are generated, each resulting from the combination of two rebalancing strategies with the five feature selection methods.

Finally, deep learning models (DNN, CNN, LSTM, hybrid CNN-LSTM architectures) are trained on the eighteen pre-processed and balanced versions from the previous phase in order to build 72 well-trained models with hyperparameter search and threshold calibration to prioritize the detection of minority classes. The evaluation phase of the 72 models is based on recall metrics, which provide a good indicator of false negative reduction.

The objective of this research is to propose a rigorous ranking of the most effective combinations and pipelines between feature selection, balancing strategy, and learning model. This ranking serves as a practical guide, indicating for each imbalance scenario which feature selection + balancing pairs and which deep learning algorithms offer the best performance in order to provide operational and sustainable recommendations for improving intrusion detection, particularly for minority classes, and to ensure their transferability to other corpora and network environments.

To this end, 72 different models resulting from the combination of five feature selection methods, two data balancing techniques, and four deep learning algorithms are tested. All experiments were conducted on the NSL-KDD benchmark dataset, with recall chosen as the main evaluation metric, since the primary objective is to improve the detection of minority attack classes while maintaining strong performance on majority classes.

A. Dataset Description

The first step in data preparation consists of downloading and loading the dataset. The dataset chosen to evaluate the effectiveness of the proposed models is the NSL-KDD dataset, which is a benchmark dataset commonly used for evaluating intrusion detection systems (IDS). The NSL-KDD dataset is composed of 41 features that represent a network connection. The features of the NSL-KDD dataset are categorized into four groups [26]:

- Basic feature group (No. 1 to 9).
- Content feature group (No. 10 to 22).
- Time-based traffic feature group (No. 23 to 31).
- Host-based traffic feature group (No. 32 to 41).

The loaded NSL-KDD dataset contains 148,517 records. There are four main categories of attacks represented in the dataset, as detailed in Table II, namely: DoS, R2L, U2R, and Probe.

The NSL-KDD dataset suffers from a significant class imbalance, which skews learning algorithms in favor of majority cases and limits their ability to model rare cases. Fig. 2 displays the percentage distribution of the NSL-KDD dataset by class.

TABLE II. MAIN ATTACKS CATEGORY PRESENTED IN NSL-KDD

Attack Category	Attack Type
DoS (Denial of Service)	Neptune, land, pod, smurf, teardrop, back, worm, udpstorm, processtable, apache2
Probe	ipsweep, satan, nmap, portsweep, mscan, saint
R2L	ftp_write, guess_password, imap, multihop, phf, spy, warezclient, warexmaster, snmpguess, named, xlock, xsnoop, snmpgetattack, httptunnel, sendmail
U2R	buffer_overflow, loadmodule, perl, rootkit, ps, xterm, sqlattack

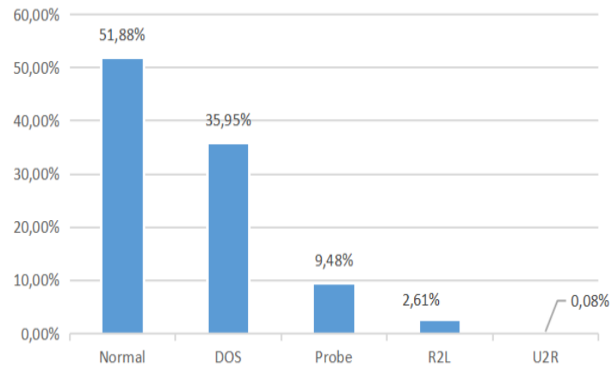

Fig. 2. NSL-KDD dataset distribution by class.

B. Data Pre-Processing

Data pre-processing is a crucial step, especially in intrusion detection systems. This phase aims to improve the quality of the dataset by cleaning, normalizing, and hot-encoding features. The cleaning process consists of removing duplicates and handling missing values, while hot-encoding converts each categorical feature value into a binary vector in which only one element is set to 1, and all other elements are 0. The element with a value of 1 indicates the presence of a specific category corresponding to the categorical feature. For example, the label (target) attribute in the NSL-KDD dataset has five distinct values: normal, DoS, Probe, R2L, and U2R. Using one-hot encoding, 'normal' can be represented with (1, 0, 0, 0, 0).

The values in the dataset consist of numerical features with completely different scales, which can vary significantly. Therefore, it is necessary to reduce these scale differences; this is where normalization comes into play. The normalization method adopted in this work is the min-max scaling, which transforms the dataset values into the range [0, 1]. This process helps improve the performance of the learning model.

In this experiment, numerical features in the NSL-KDD dataset are normalized using Min–Max normalization, as defined in Eq. (1), which is one of the most commonly used normalization techniques.

$$\tilde{\mathbf{x}} = \frac{\mathbf{x} - \mathbf{Min(x)}}{\mathbf{Max(x)} - \mathbf{Min(x)}} \qquad (1)$$

where, $\tilde{x}$ is the normalized value of x.

C. Feature Selection

Feature selection is an important step in the machine learning process after data cleaning and normalization, as it aims to identify the most relevant or informative features. It can help eliminate noise and reduce dimensionality in order to improve model performance. This process reduces the dimensionality of the data, enhances model accuracy, and decreases computational time. There are several techniques for feature selection. This study considers five feature selection techniques based on two main approaches: statistical approaches like chi-square test, correlation, and information gain, and learning-based approaches like XGBoost and AutoEncoder.

- Chi-square [22,27]: By applying the Chi-square statistical test to our dataset, the method selected 20 out of 41 features, resulting in a 51.22% reduction. The number of top features to select, k, was treated as a hyperparameter and tuned over the candidate set {10, 13, 15, 18, 20, 23, 25, 28, 30, 33, 35, 38}. Based on cross-validation, k=20 yielded the highest accuracy and was therefore adopted for the final model.
- Correlation [28]: The calculated correlation values were sorted in descending order, and all features with a correlation close to 1 are selected (above the threshold of 0.9). This threshold was manually tuned to achieve the best performance. This method selected 35 out of 41 features, yielding approximately a 14.62% reduction.
- Information Gain [29]: For information gain, the features whose values were greater than 0.17 are selected, resulting in 21 retained features, resulting in a 48.78% reduction of the dataset.
- XGBoost [30,31]: In XGBoost, the features with importance values between 1224 and 100 are selected, which led to the selection of 22 features, a 46% reduction.
- Autoencoder [32]: An optimal size for the low-dimensional representation is determined while maintaining a reasonably low reconstruction error. The encoding_dims hyperparameter was tuned over the range (15, 42), and the lowest reconstruction error Mean Squared Error (MSE) was achieved with 35 latent dimensions, yielding 35 retained features, achieving a 31% reduction.

These methods allowed us to generate five different variants of the NSL-KDD dataset, and the result is detailed in Table III.

TABLE III. NUMBER OF FEATURES RETAINED FOR EACH FEATURE SELECTION METHOD

Datasets	Number of features
Original	41
Chi 2	20
Correlation	35
Information Gain	21
XGBoost	22
AutoEncoder	35

D. Class Rebalancing

The dataset is well-cleaned, meaning it contains no missing or redundant information. However, it suffers from class imbalance. As a result, the learning algorithms gravitate toward the majority class, limiting their ability to recognize infrequent, yet highly damaging intrusions such as U2R and R2L attacks. To address the class imbalance problem, this work applies approaches based on two techniques:

1) SMOTE (Synthetic Minority Oversampling Technique) [33]: SMOTE enables the creation of new synthetic instances by linearly interpolating the features of neighboring minority instances. The SMOTE balancing technique selects a minority sample, then randomly chooses one or more of its nearest neighbors to generate synthetic instances. The SMOTE hyperparameter k_neighbors = 3 was chosen manually from the candidate values {2, 3, 4}.

2) ADASYN (Adaptive Synthetic) [34]: ADASYN also generates synthetic data for minority instances but places more emphasis on the samples that are harder to learn, based on their relative density. Minority instances that are farther from the decision boundary are prioritized for synthetic data. For ADASYN, the sampling strategy is set to target the minority class, meaning that only the minority class is oversampled.

E. Deep Learning Models Applied

Following data cleaning, feature selection, and dataset balancing, the resulting datasets are used to train four different neural network models:

The first model, a DNN, is a type of standard artificial neural network composed of multiple layers of neurons. Each layer is connected to both the preceding and the following layer, forming a layered architecture [35]. This study proposes a DNN model composed of an input layer followed by two hidden layers: the first with 32 neurons and the second with 16, both using the ReLU activation function. The final output layer contains 5 neurons, one per class, with a Softmax activation function to support multi-class classification.

The second model is the CNN, which is capable of handling complex data, something that is often not feasible with traditional DNNs [31]. The primary goal of a CNN in this context is to extract important features from raw network data. During this process, the earliest layers serve as convolutional feature extractors, convolving the input with learned filters. Our CNN model consists of an input layer, a one-dimensional convolutional layer with 32 filters of size 3 and ReLU activation, followed by a MaxPooling layer (pool size = 2), a flattening layer, and a fully connected layer of 16 units employing the ReLU activation function. It ends with an output layer of 5 neurons using the Softmax activation function.

The third model LSTM, as described in [36], is an improved type RNN engineered to alleviate both vanishing and exploding gradient problem. These issues arise during back propagation when gradients either shrink too much or grow too rapidly, making it difficult to update weights properly and often leading to the failure of the learning process. LSTM composed of three main components called gates: the forget gate, input gate, and output gate [37].

The LSTM model used in this study consists of two LSTM layers: the first with 16 units and L2 regularization to reduce overfitting, followed by a second LSTM layer with 8 units. The output is then flattened and passed to a Softmax-activated layer with 5 neurons.

Finally, a hybrid CNN-LSTM model is developed, combining convolutional and recurrent components. It starts with an input layer followed by a 1D convolutional layer (32 filters, ReLU), a MaxPooling layer (size 2), then a second convolutional layer (16 filters, ReLU) and another MaxPooling layer. The extracted features are passed through two stacked LSTM layers (16 and 8 units, respectively), with the second one configured to return only the final output. The network concludes with a flatten layer and a Softmax output layer with 5 neurons. Each architecture was specifically designed to explore how spatial, temporal, and hybrid feature learning impacts the detection of various types of intrusions.

For all four deep learning models, a consistent set of training parameters was applied. Specifically, the loss function used is 'categorical_crossentropy', while the optimizer selected is 'adam'; the model performance was monitored using 'accuracy' as the evaluation metric during both training and validation phases. Each model was trained over 100 epochs with a batch size of 32 to ensure stable convergence.

IV. Results and Discussion

This section analyzes and compares the performance of the different models by conducting our experiments on the benchmark NSL-KDD dataset. The NSL-KDD dataset consists of 41 features that describe a network connection and contains a total of 148 517 records. However, to address the issue of class imbalance, only 58 397 records are selected for the experiments. These records are distributed and detailed in Table IV.

TABLE IV. Distribution of the Number of Records Per Class

Attacks type	Number of Records in the Dataset	Number of Records Selected for the Experiments
No attacks (normal)	77054	20034
Dos attacks	53387	20287
Probe attacks	14077	14077
R2L attacks	3880	3880
U2R attacks	119	119

TABLE V. Number of Records Per Class and Per Balancing Technique

	SMOTE	ADASYN
Normal	20287	20287
DoS	20287	20034
Probe	20287	14077
R2L	20287	3880
U2R	20287	20304

For both balancing techniques, SMOTE and ADASYN, were applied to each of the six dataset variants generated during the feature selection phase, resulting in 12 additional datasets: 101 435 records for SMOTE and 78 550 records for ADASYN, distributed as in Table V.

The training of the four neural networks was performed on the 18 datasets produced during the feature selection and sampling phases, resulting in a total of 72 models and the main goal of IDS is to minimize false negatives, which represent undetected attacks and pose a significant security risk. Therefore, to evaluate the performance of the 72 models, recall is used, defined by Eq. (2), as the key metric, as it is crucial in IDS.

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

Other evaluation metrics that were used, other than recall, precision, defined by Eq. (3), and F1-score defined by Eq. (4):

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

$$F1 - score = \frac{2\ .Precision\ .Recall}{Precision+\ Recall} \quad (4)$$

where,

- TP (True Positives): correctly detected intrusions.
- FN (False Negative): intrusions that were not detected.
- FP (False Positive): A benign event incorrectly flagged as intrusion (false alarm).

As previously mentioned, the NSL-KDD dataset often suffers from class imbalance, where certain types of attacks, particularly R2L and U2R, are significantly underrepresented compared to majority classes such as normal connections or DoS attacks. Our objective is to improve IDS performance by reducing false negatives in these minority classes, which are the most harmful, by finding the optimal combination of feature selection techniques (Chi2, Correlation, Information Gain (IG), XGBoost, and Autoencoder), two dataset balancing techniques (SMOTE and ADASYN), and deep learning neural network models (DNN, CNN, LSTM, and CNN-LSTM).

Therefore, the evaluation results will focus on the recall metric for the two minority classes: R2L, which accounts for 2.61% of the dataset, and U2R, which accounts for 0.08%. Table VI shows the results obtained on the NSL-KDD dataset variants for the R2L class using only feature selection techniques without any data balancing.

TABLE VI. Recall Scores by Different Models on the R2L Class With and Without Feature Selection

	DNN	CNN	LSTM	CNN-LSTM
Chi2	0.82	0.70	0.59	0.81
Correlation	0.91	0.84	0.63	0.87
Information Gain	0.62	0.46	0.47	0.73
XGBoost	0.89	0.91	0.80	0.82
Autoencoder	0.92	0.90	0.89	0.85
Origine	0.90	0.90	0.88	0.94

The results in Table VI show that without any feature selection or data balancing techniques, the CNN-LSTM hybrid model is the best choice, achieving a recall of 94% for the R2L class. Comparing these results with those obtained using feature selection techniques, it is observed that the autoencoder combined with the DNN model produces the best results among all feature selection methods, achieving 92%. The correlation method is also achieving a recall of 91 % for the R2L class with the DNN model. It is also worth noting that the Chi-square method, despite selecting only 20 features, provides good results, especially with the DNN and CNN-LSTM models with the recall reaches 82% and 81%, respectively. In contrast, the Information Gain method yields the worst performance across all models, indicating that it likely discards critical features necessary for detecting R2L attacks. Therefore, this technique is not recommended for this class.

Table VII presents the recall model results for the U2R class, the most underrepresented in the dataset, accounting for only 0.08% of the data, with and without feature selection. The results show that the U2R class is extremely sensitive to feature selection methods. Without any selection, all models yield almost the same result, with a slight advantage for the hybrid CNN-LSTM model, with 45%. It is also observed that the Chi-square selection method with just 20 features was able to reach 42% with the LSTM model, like the correlation method with the DNN model. On the other hand, the Information Gain and XGBoost methods should be avoided, as they result in a significant loss of critical information for this class.

TABLE VII. RECALL SCORES USING DIFFERENT MODELS ON THE U2R CLASS WITH AND WITHOUT FEATURE SELECTION

	DNN	CNN	LSTM	CNN-LSTM
Chi2	0.32	0.29	0.42	0.39
Correlation	0.42	0.39	0.39	0.32
Information Gain	0.00	0.00	0.06	0.00
XGBoost	0.10	0.10	0.06	0.00
Autoencoder	0.23	0.29	0.32	0.00
Origine	0.42	0.39	0.42	0.45

Table VIII, Table IX and Table X show the results for DNN with autoencoder feature selection technique, the hybrid model CNN-LSTM without selection feature method, LSTM with chi-square selection feature method, and DNN with correlation feature selection method, respectively. The results reveal that, when the dataset is left unbalanced, the U2R minority class is poorly detected. Nevertheless, the overall performance remains strong when correlation-based feature selection is combined with a DNN model, indicating that this pairing effectively mitigates part of the imbalance problem and yields solid results across the remaining classes. The low results obtained for the minority classes have influenced the macro-avg F1-score, which reached around 80%, leading to a relatively modest overall value. However, the majority of classes achieved excellent detection rates, exceeding 90%.

The results show that, without applying any balancing technique, the correlation-based feature selection method provided the best performance, achieving over 98% F1-score for the majority classes and a macro-avg F1-score of approximately 87%. This demonstrates that even without resampling, the correlation method effectively captures the most discriminative features, ensuring strong detection performance across most classes.

TABLE VIII. PERFORMANCE OF THE DNN-BASED MULTICLASS CLASSIFICATION ON THE NSL-KDD DATASET USING AUTOENCODER WITHOUT BALANCING TECHNIQUES

Class	Precision	Recall	F1-mesure
Normal	0.99	0.98	0.99
DoS	0.97	0.94	0.95
Probe	0.97	0.99	0.98
R2L	0.78	0.92	0.85
U2R	0.64	0.23	0.33
Macro avg	0.87	0.81	0.82

TABLE IX. PERFORMANCE OF THE CNN-BASED MULTICLASS CLASSIFICATION ON THE NSL-KDD DATASET WITHOUT ANY FEATURE SELECTION METHOD

Class	Precision	Recall	F1-mesure
Normal	0.99	0.99	0.99
DoS	0.98	0.95	0.97
Probe	0.98	0.98	0.98
R2L	0.81	0.94	0.87
U2R	0.54	0.45	0.49
Macro avg	0.86	0.86	0.86

TABLE X. PERFORMANCE OF THE DNN-BASED MULTICLASS CLASSIFICATION ON THE NSL-KDD DATASET WITH CORRELATION METHOD

Class	Precision	Recall	F1-mesure
Normal	1.00	1.00	1.00
DoS	0.97	0.96	0.97
Probe	0.98	0.99	0.98
R2L	0.88	0.91	0.89
U2R	0.65	0.42	0.51
Macro avg	0.90	0.85	0.87

Table XI presents the recall obtained for the R2L class after applying the SMOTE balancing technique to the six NSL-KDD dataset variants produced following the feature selection phase. The results in show that after applying the SMOTE balancing technique, almost all recall values improved for the R2L class, except for the combination of Chi2 and CNN-LSTM, which initially recorded a recall of 81% a better result than after applying SMOTE, which dropped to 73.15%. The experiments and the recall results presented indicate that the most effective pipeline for detecting the R2L class combines correlation based feature selection, a CNN model, and SMOTE, achieving a recall of 96.58 %. Several other configurations also performed very well, reaching 95.86 % recall specifically, the pipeline pairing autoencoder-based selection with a CNN-LSTM architecture and SMOTE, and the one combining XGBoost-based selection, a DNN model, and SMOTE.

TABLE XI. RECALL SCORES BY MODELS ON THE R2L CLASS WITH FEATURE SELECTION AND THE SMOTE METHOD

	DNN	CNN	LSTM	CNN-LSTM
Chi2	0.8947	0.8857	0.7584	0.7315
Correlation	0.9348	0.9658	0.9278	0.9512
Information Gain	0.9234	0.9376	0.7421	0.9511
XGBoost	0.9586	0.8378	0.9105	0.9496
Autoencoder	0.9405	0.9465	0.8966	0.9586
Origine	0.9488	0.9312	0.8409	0.9563

Table XII presents the results of recall for the U2R class after applying the SMOTE balancing technique on the six variants of the NSL-KDD dataset generated during the feature selection phase. For the most underrepresented class, U2R, the results show a significant performance improvement compared to the results without balancing. For example, the combination of DNN with information gain improved from 0% to 98.57% recall after applying the SMOTE balancing technique. The same dramatic improvement is observed with the CNN-LSTM model combined with an autoencoder. SMOTE proves to be highly effective in boosting performance on the minority class U2R, especially when used with the correlation feature selection method and DNN model 99.88% or with the autoencoder feature selection method and CNN-LSTM hybrid model, which was able to reach 99.68%. The results also show that the autoencoder appears to be the most stable method, as it yielded very good results across the majority of models.

TABLE XII. RECALL SCORES BY MODELS ON THE U2R CLASS WITH FEATURE SELECTION AND THE SMOTE METHOD

	DNN	CNN	LSTM	CNN-LSTM
Chi2	0.9631	0.8960	0.9457	0.9422
Correlation	0.9988	0.8906	0.9576	0.9767
Information Gain	0.9857	0.9410	0.8987	0.9454
XGBoost	0.9196	0.9757	0.9366	0.9680
Autoencoder	0.9778	0.9623	0.9899	0.9968
Origine	0.9576	0.9851	0.9380	0.9572

Table XIII, Table XIV and Table XV show the results for the XGBoost + SMOTE + DNN pipeline, the Autoencoder + SMOTE + CNN-LSTM pipeline, and the Correlation + SMOTE + DNN pipeline, respectively. The confusion matrices demonstrate that balancing the dataset dramatically improves the detection of minority classes. Among all evaluated setups, the Correlation + SMOTE + DNN pipeline and the Autoencoder + SMOTE + CNN-LSTM combination deliver the best overall performance, most notably achieving the highest recall across all classes, including the minority ones.

Alongside the significant improvement observed in the detection of minority classes after applying the balancing techniques, the majority classes maintained consistently high detection rates. This indicates that both the balancing methods and the feature selection strategies did not negatively impact the performance of the majority classes, which achieved F1-scores exceeding 95%. Notably, the combinations Correlation + SMOTE + DNN and Autoencoder + SMOTE + CNN-LSTM produced the best results, where the macro-average F1-score reached values above 97%, demonstrating the overall efficiency and stability of the proposed pipelines across all attacks.

Table XVI presents the results of recall for the R2L class using the ADASYN balancing technique across the six NSL-KDD dataset variants generated after the feature selection phase. While the SMOTE balancing method remains stable for the R2L class and yields very good results for most combinations, the ADASYN method appears less stable for this minority class. The combinations that achieved the highest recall for the R2L class were Correlation + ADASYN + DNN and Autoencoder + ADASYN + DNN, which recorded recall scores of 86.64 % and 85.88 %, respectively, on the other hand, it gives modest results in combinations like LSTM with information gain (32.98% recall) or with Chi2 (49.65% recall).

For this class, the DNN model demonstrates greater stability and performance than the LSTM model, which continues to struggle, likely due to its sensitivity to imbalanced or synthetically generated data.

TABLE XIII. PERFORMANCE OF THE DNN-BASED MULTICLASS CLASSIFICATION ON THE NSL-KDD DATASET USING XGBOOST ET SMOTE PIPELINE

Class	Precision	Recall	F1-mesure
Normal	0.9920	0.9843	0.9881
DoS	0.9636	0.9272	0.9454
Probe	0.9763	0.9899	0.9830
R2L	0.8846	0.9584	0.9202
U2R	0.9724	0.9196	0.9453
Macro avg	0.9578	0.9560	0.9564

TABLE XIV. PERFORMANCE OF THE CNN-LSTM-BASED MULTICLASS CLASSIFICATION ON THE NSL-KDD DATASET WITH AUTOENCODER ET SMOTE PIPELINE

Class	Precision	Recall	F1-mesure
Normal	0.9860	0.9929	0.9894
DoS	0.9843	0.9236	0.9530
Probe	0.9844	0.9898	0.9871
R2L	0.9524	0.9586	0.9555
U2R	0.9548	0.9968	0.9753
Macro avg	0.9724	0.9723	0.9721

TABLE XV. PERFORMANCE OF THE DNN-BASED MULTICLASS CLASSIFICATION ON THE NSL-KDD DATASET WITH CORRELATION ET SMOTE PIPELINE

Class	Precision	Recall	F1-mesure
Normal	0.9959	0.9929	0.9944
DoS	0.9586	0.9512	0.9549
Probe	0.9828	0.9911	0.9869
R2L	0.9677	0.9348	0.9510
U2R	0.9623	0.9988	0.9802
Macro avg	0.9735	0.9738	0.9735

TABLE XVI. RECALL SCORES BY DIFFERENT MODELS ON THE R2L CLASS WITH FEATURE SELECTION AND THE ADASYN METHOD

	DNN	CNN	LSTM	CNN-LSTM
Chi2	0.7812	0.6480	0.4965	0.6585
Correlation	0.8664	0.6803	0.5113	0.8134
Information Gain	0.6132	0.7265	0.3298	0.6434
XGBoost	0.6132	0.7667	0.5823	0.8502
Autoencoder	0.8588	0.7691	0.7567	0.7025
Origine	0.8468	0.7602	0.4234	0.7610

Table XVII presents the results for the U2R class after applying the ADASYN balancing technique on the six NSL-KDD dataset variants generated during the feature selection phase. The results from Table X for the minority class, U2R, show exceptionally high recall scores, almost always between 98% and 100%, regardless of the method or model used. Like SMOTE, ADASYN has a very positive effect on the detection of this ultra-minority class. Unlike SMOTE, ADASYN focuses more on difficult-to-learn examples, which helps the models better capture edge cases.

TABLE XVII. RECALL SCORES BY DIFFERENT MODELS ON THE U2R CLASS WITH FEATURE SELECTION AND THE ADASYN METHOD

	DNN	CNN	LSTM	CNN-LSTM
Chi2	0.9766	1.00	0.9567	0.9983
Correlation	1.00	0.9974	0.9812	0.9822
Information Gain	0.9817	0.9789	0.9965	0.9843
XGBoost	1.00	0.9979	0.9331	0.9949
Autoencoder	0.9849	1.00	0.9937	1.00
Origine	0.9867	0.9992	0.9779	0.9890

The CNN model appears to be very stable and well-suited to most feature selection methods, likely because CNN does not require a strict temporal structure like LSTM, making it more robust to synthetic data generated by SMOTE or ADASYN.

Table XVIII, Table XIX, and Table XX show the results for the Autoencoder + ADASYN + CNN pipeline, the Autoencoder + ADASYN + DNN pipeline, and the Correlation + ADASYN + DNN pipeline, respectively. Our experiments reveal several clear trends. First, ADASYN oversampling causes a considerable increase in recall for the U2R class, the smallest minority class, demonstrating its effectiveness in contexts of extreme imbalance. Second, ADASYN provides excellent performance for most feature selection/model combinations. Among the selection methods, autoencoder and CNN stand out for their stability and overall superiority. When coupled with ADASYN, the gain is even more pronounced. The best pipelines consistently include ADASYN, often combined with autoencoder selection and a deep model capable of exploiting the new densified minority regions. The most effective combinations are: Autoencoder + ADASYN + CNN, Autoencoder + ADASYN + LSTM, and ADASYN + CNN, Autoencoder + ADASYN + DNN, Correlation + ADASYN + DNN.

These configurations consistently achieve the best recall rates, particularly for U2R and R2L. According to the results, the ADASYN technique appears to be the most suitable balancing method, as it significantly improves the detection of minority classes. Moreover, the majority classes also benefited from this approach, showing enhanced performance particularly with the combination Correlation + ADASYN + DNN, which achieved an F1-score exceeding 96%. In addition, the macro-average F1-score reached around 92% for most combinations, confirming the overall effectiveness of ADASYN in improving detection across all classes.

TABLE XVIII. PERFORMANCE OF THE CNN-BASED MULTICLASS CLASSIFICATION ON THE NSL-KDD DATASET WITH AUTOENCODER ET ADASYN PIPELINE

Class	Precision	Recall	F1-mesure
Normal	09802	09885	09843
DoS	0.9727	0.9275	0.9496
Probe	0.9686	0.9875	0.9779
R2L	0.8482	0.7691	0.8067
U2R	0.9601	1.00	0.9800
Macro avg	0.9461	0.9345	0.9397

TABLE XIX. PERFORMANCE OF THE DNN-BASED MULTICLASS CLASSIFICATION ON THE NSL-KDD DATASET WITH AUTOENCODER ET ADASYN PIPELINE

Class	Precision	Recall	F1-mesure
Normal	0.9842	0.9855	0.9848
DoS	0.9712	0.9305	0.9504
Probe	0.9741	0.9870	0.8805
R2L	0.7894	0.8588	0.8226
U2R	0.9710	0.9849	0.9779
Macro avg	0.9380	0.9493	0.9433

TABLE XX. PERFORMANCE OF THE DNN-BASED MULTICLASS CLASSIFICATION ON THE NSL-KDD DATASET WITH CORRELATION ET ADASYN PIPELINE

Class	Precision	Recall	F1-mesure
Normal	0.9969	0.9918	0.9943
DoS	0.9766	0.9618	0.9691
Probe	0.9805	0.9887	0.9846
R2L	0.8953	0.8664	0.8802
U2R	0.9802	1.00	0.9900
Macro avg	0.9659	0.9617	0.9637

By comparing all the results obtained (without balancing, with SMOTE, and with ADASYN) and focusing on the R2L and U2R classes, the results obtained without class balancing indicate that minority classes, particularly U2R, are difficult to detect reliably, often resulting in low or even zero performance for certain combinations (e.g., Information Gain with DNN or CNN-LSTM). Adding feature selection techniques like Chi2, correlation, or autoencoder slightly improves results for R2L but remains unstable for U2R. However, the CNN-LSTM and DNN models show strong potential when combined with methods capable of preserving complex relationships, such as autoencoders or even using no selection.

With the introduction of class balancing, performance on U2R improves significantly, often exceeding 95% with SMOTE and even reaching 100% with ADASYN for most models. This highlights that class imbalance was the main barrier to effective detection. ADASYN proves to be slightly more effective than SMOTE, as it targets the more challenging examples and provides more consistent results. The CNN model stands out due to its strong stability on U2R, likely because of its ability to capture local patterns that are resilient to synthetic noise.

For R2L, the results are more variable: the DNN architecture is the most stable, while LSTM remains sensitive to feature selection and artificial data. Overall, the combinations ADASYN + DNN or CNN-LSTM + Autoencoder represent a good compromise between performance and stability for effectively detecting R2L and U2R attacks.

V. Conclusion

This study systematically explores the combined impact of feature selection techniques, including Chi-square, correlation, information gain, XGBoost, and autoencoders, class balancing methods (SMOTE and ADASYN), and various deep learning models (DNN, CNN, LSTM, and CNN-LSTM) on the detection of R2L and U2R attacks two minority classes that are notoriously difficult to detect in intrusion detection systems (IDS). Our results show that recall, a key metric for reducing false negatives, can be significantly improved through appropriate methodological choices. More specifically, class balancing using SMOTE and especially ADASYN proved essential for restoring effective detection capabilities for the U2R class, which is often overlooked by conventional models. At the same time, advanced feature selection methods such as autoencoder, based on unsupervised learning, with LSTM or the model hybrid CNN-LSTM or correlation, static approach, with almost all models demonstrated strong potential in generating optimal input representations for deep architectures. This was particularly evident with the CNN-LSTM hybrid, which is well-suited for capturing complex inter-attribute relationships. The combination of ADASYN, autoencoder, and CNN-LSTM emerged as the most robust and high-performing configuration for maximizing recall on both minority classes.

The study concludes that no model is perfect for all attack classes: some configurations achieve high recall on R2L but perform less well on U2R, while others excel on U2R but not on R2L. As future research directions, will evaluate the proposed methodological framework on the newer and larger CIC-IDS2017 dataset, considering both minority and majority classes, and develop a voting-based approach that combines the most effective models and only issues an alert when the majority converges, which enhances overall robustness and balances the detection of all classes. Beyond the performance achieved, this work highlights the importance of methodical design of IDS pipelines for detecting minority classes. The results provide concrete guidance for practitioners to choose robust combinations of feature selection techniques, rebalancing, and deep learning models. These conclusions can guide the deployment of more reliable IDS in real-world environments, where stability and reduction of false negatives are critical. Future prospects include evaluating the generalizability of the identified pipelines on more recent datasets and in real-time contexts.

References

[1] Nazario J. DDoS attack evolution. Network Security 2008;2008:7–10. https://doi.org/10.1016/S1353-4858(08)70086-2.

[2] Cisco. Cisco Annual Internet Report (2018–2023) 2023.

[3] Gartner's. Gartner 2024. https://www.gartner.com/en/articles/information-security (accessed July 28, 2025).

[4] Ventures C. 2022 0fficial Cybercrime Report 2023.

[5] Guo Y. A review of Machine Learning-based zero-day attack detection: Challenges and future directions. Computer Communications 2023;198:175–85. https://doi.org/10.1016/j.comcom.2022.11.001.

[6] Rekha G, Tyagi AK. Necessary information to know to solve class imbalance problem: From a user's perspective. Lecture Notes in Electrical Engineering 2020;597:645–58. https://doi.org/10.1007/978-3-030-29407-6_46.

[7] Quincozes SE, Albuquerque C, Passos D, Mossé D. A survey on intrusion detection and prevention systems in digital substations. Computer Networks 2021;184:107679. https://doi.org/10.1016/j.comnet.2020.107679.

[8] Abdulganiyu OH, Tchakoucht TA, Alaoui AEH, Saheed YK. Attention-driven multi-model architecture for unbalanced network traffic intrusion detection via extreme gradient boosting. Intelligent Systems with Applications 2025;26:200519. https://doi.org/10.1016/j.iswa.2025.200519.

[9] Harini R, Maheswari N, Ganapathy S, Sivagami M. An effective technique for detecting minority attacks in NIDS using deep learning and sampling approach. Alexandria Engineering Journal 2023;78:469–82. https://doi.org/10.1016/j.aej.2023.07.063.

[10] Yu L, Xu L, Jiang X. A high-performance multimodal deep learning model for detecting minority class sample attacks. Symmetry 2023;16:42.

[11] Akashdeep, Manzoor I, Kumar N. A feature reduced intrusion detection system using ANN classifier. Expert Systems with Applications 2017;88:249–57. https://doi.org/10.1016/j.eswa.2017.07.005.

[12] Devendiran R, Turukmane AV. Dugat-LSTM: Deep learning based network intrusion detection system using chaotic optimization strategy. Expert Systems with Applications 2024;245:123027. https://doi.org/10.1016/j.eswa.2023.123027.

[13] Vishwakarma M, Kesswani N. A new two-phase intrusion detection system with Naïve Bayes machine learning for data classification and elliptic envelop method for anomaly detection. Decision Analytics Journal 2023;7:100233. https://doi.org/10.1016/j.dajour.2023.100233.

[14] Alshamy R, Ghurab M, Othman S, Alshami F. Intrusion Detection Model for Imbalanced Dataset Using SMOTE and Random Forest Algorithm. In: Abdullah N, Manickam S, Anbar M, editors. Advances in Cyber Security, Singapore: Springer; 2021, p. 361–78. https://doi.org/10.1007/978-981-16-8059-5_22.

[15] Fan Z, Sohail S, Sabrina F, Gu X. Sampling-Based Machine Learning Models for Intrusion Detection in Imbalanced Dataset. Electronics 2024;13:1878. https://doi.org/10.3390/electronics13101878.

[16] Liu J, Gao Y, Hu F. A fast network intrusion detection system using adaptive synthetic oversampling and LightGBM. Computers & Security 2021;106:102289. https://doi.org/10.1016/j.cose.2021.102289.

[17] Milosevic MS, Ciric VM. Extreme minority class detection in imbalanced data for network intrusion. Computers & Security 2022;123:102940. https://doi.org/10.1016/j.cose.2022.102940.

[18] Mbow M, Koide H, Sakurai K. An Intrusion Detection System for Imbalanced Dataset Based on Deep Learning. 2021 Ninth International Symposium on Computing and Networking (CANDAR), 2021, p. 38–47. https://doi.org/10.1109/CANDAR53791.2021.00013.

[19] Rani R HJ, Barve A, Malviya A, Ranjan V, Jeet R, Bhosle N. Enhancing detection rates in intrusion detection systems using fuzzy integration and computational intelligence. Computers & Security 2025;157:104577. https://doi.org/10.1016/j.cose.2025.104577.

[20] Bedi P, Gupta N, Jindal V. I-SiamIDS: an improved Siam-IDS for handling class imbalance in network-based intrusion detection systems. Appl Intell 2021;51:1133–51. https://doi.org/10.1007/s10489-020-01886-y.

[21] Gupta N, Jindal V, Bedi P. CSE-IDS: Using cost-sensitive deep learning and ensemble algorithms to handle class imbalance in network-based intrusion detection systems. Computers and Security 2022;112. https://doi.org/10.1016/j.cose.2021.102499.

[22] Ahmad R, Alsmadi I, Alhamdani W, Tawalbeh L. A comprehensive deep learning benchmark for IoT IDS. Computers & Security 2022;114:102588. https://doi.org/10.1016/j.cose.2021.102588.

[23] Xu B, Sun L, Mao X, Liu C, Ding Z. Strengthening Network Security: Deep Learning Models for Intrusion Detection with Optimized Feature Subset and Effective Imbalance Handling. Computers, Materials & Continua 2024;78.

[24] Long H, Li H, Tang Z, Zhu M, Yan H, Luo L, et al. BOA-ACRF: An intrusion detection method for data imbalance problems. Computers and Electrical Engineering 2025;124:110320. https://doi.org/10.1016/j.compeleceng.2025.110320.

[25] Xue Y, Kang C, Yu H. HAE-HRL: A network intrusion detection system utilizing a novel autoencoder and a hybrid enhanced LSTM-CNN-based residual network. Computers & Security 2025;151:104328. https://doi.org/10.1016/j.cose.2025.104328.

[26] Choudhary S, Kesswani N. Analysis of KDD-Cup'99, NSL-KDD and UNSW-NB15 Datasets using Deep Learning in IoT. Procedia Computer Science 2020;167:1561–73. https://doi.org/10.1016/j.procs.2020.03.367.

[27] Devi AG, Borra SPR, Haritha T, Mandava VSR, Balaji T, Sagar KV, et al. An Improved CHI2 Feature Selection Based a Two-Stage Prediction of Comorbid Cancer Patient Survivability. RIA 2023;37:83–92. https://doi.org/10.18280/ria.370111.

[28] Yu L, Liu H. Efficient Feature Selection via Analysis of Relevance and Redundancy. J Mach Learn Res 2004;5:1205–24.

[29] Mohammadi S, Desai V, Karimipour H. Multivariate Mutual Information-based Feature Selection for Cyber Intrusion Detection. 2018 IEEE Electrical Power and Energy Conference (EPEC) 2018:1–6. https://doi.org/10.1109/EPEC.2018.8598326.

[30] Alsahaf A, Petkov N, Shenoy V, Azzopardi G. A framework for feature selection through boosting. Expert Systems with Applications 2022;187:115895. https://doi.org/10.1016/j.eswa.2021.115895.

[31] Albawi S, Mohammed TA, Al-Zawi S. Understanding of a convolutional neural network. 2017 international conference on engineering and technology (ICET), Ieee; 2017, p. 1–6.

[32] Alaghbari KA, Lim H-S, Saad MHM, Yong YS. Deep Autoencoder-Based Integrated Model for Anomaly Detection and Efficient Feature Extraction in IoT Networks. IoT 2023;4:345–65. https://doi.org/10.3390/iot4030016.

[33] Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research 2002;16:321–57. https://doi.org/10.1613/jair.953.

[34] He H, Bai Y, Garcia EA, Li S. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), 2008, p. 1322–8. https://doi.org/10.1109/IJCNN.2008.4633969.

[35] Tang C, Luktarhan N, Zhao Y. SAAE-DNN: Deep Learning Method on Intrusion Detection. Symmetry 2020;12.

[36] Sherstinsky A. Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. Physica D: Nonlinear Phenomena 2020;404:132306. https://doi.org/10.1016/j.physd.2019.132306.

[37] Yu Y, Si X, Hu C, Zhang J. A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. Neural Comput 2019;31:1235–70. https://doi.org/10.1162/neco_a_01199.