ÉCOLE SUPÉRIEURE EN INFORMATIQUE
SIDI BEL ABBÈS
ALGERIA

INTERNSHIP REPORT Machine

Learning-Based Intrusion Detection Using the NSL-KDD Dataset

Intern:

Mohammed Bekkouche
LabRI-SBA Laboratory
École Supérieure en Informatique, Sidi Bel Abbès, Algeria

Internship Period:	6 August – 16 August 2026
Location:	Tunis, Tunisia
Funding Institution:	École Supérieure en Informatique, Sidi Bel Abbès, Algeria
Research Topic:	Machine Learning-Based Intrusion Detection

2026

Executive Summary

This internship, funded by the **École Supérieure en Informatique (ESI-SBA), Sidi Bel Abbès**, is being carried out in Tunis, Tunisia, from 6 August to 16 August 2026. The main objective is to investigate machine learning approaches for intrusion detection using the NSL-KDD dataset.

The work focuses on establishing a systematic and reproducible experimental framework for multiclass intrusion detection. Several dimensions are considered, including data preprocessing, feature selection, class-imbalance handling, machine learning models, evaluation metrics, and model ranking.

Several feature-selection methods and machine learning algorithms have already been integrated into the experimental framework. Random Over-Sampling and SMOTE have also been incorporated to study the influence of class imbalance. A multi-metric evaluation strategy has been implemented using accuracy, balanced accuracy, precision, recall, F1-score, specificity, kappa, ROC-AUC, PR-AUC, Log Loss, and Brier Score, when applicable.

A normalized weighted ranking method has also been developed to compare different experimental configurations. This method accounts for the fact that some evaluation metrics may not be available for every machine learning model.

Preliminary experiments show significant differences between machine learning models and feature-selection techniques. The results obtained so far confirm the importance of evaluating intrusion-detection systems using several complementary metrics rather than relying exclusively on accuracy.

Contents

Introduction

Intrusion Detection Systems (IDS) are an important component of cybersecurity infrastructure. Their purpose is to identify malicious or abnormal activities in computer networks and systems.

The increasing diversity and complexity of cyberattacks make the development of reliable intrusion-detection mechanisms a challenging research problem. Machine learning techniques provide promising approaches because they can learn patterns from network traffic and automatically classify network connections according to their characteristics.

This internship focuses on the application of machine learning to intrusion detection using the NSL-KDD dataset. The objective is not only to compare different classification algorithms, but also to study the influence of feature-selection techniques and class-balancing strategies.

The internship therefore adopts a systematic experimental methodology in which several components of an intrusion-detection pipeline are evaluated under a common framework.

Chapter 1

Internship Context and Motivation

1.1 General Information

The internship is funded by the **École Supérieure en Informatique (ESI-SBA), Sidi Bel Abbès** and is being carried out in **Tunis, Tunisia**, from **6 August to 16 August 2026**.

The scientific work is focused on machine learning-based intrusion detection and the experimental evaluation of different approaches using the NSL-KDD benchmark dataset.

1.2 Motivation

1.2.1 Scientific Motivation

The main scientific motivation of this internship is to strengthen research activities in the field of artificial intelligence and cybersecurity.

In particular, the internship aims to investigate how different machine learning algorithms, feature-selection methods, and class-balancing techniques influence the performance of multiclass intrusion-detection systems.

Another important motivation is to develop a reproducible experimental framework that can be used to conduct systematic comparisons rather than evaluating machine learning algorithms independently.

1.2.2 Motivation for Conducting the Internship in Tunisia

An important personal and cultural motivation for conducting this internship in Tunisia is the opportunity to carry out scientific research in a **Muslim country**. As a Muslim researcher, working in a Muslim cultural environment provides an opportunity to pursue scientific activities in a context that is compatible with my cultural and personal values.

1.3 Main Objectives

The main objectives of the internship are:

1. Develop a reproducible machine-learning framework for intrusion detection.
2. Investigate different feature-selection methods.
3. Compare several machine learning classification algorithms.
4. Study the effect of class-imbalance handling techniques.
5. Evaluate machine learning models using multiple complementary metrics.
6. Develop a principled method for ranking different experimental configurations.
7. Identify promising combinations of machine learning model, feature-selection method, and balancing strategy.
8. Generate experimental results suitable for subsequent scientific publication.

Chapter 2

Dataset and Experimental Methodology

2.1 NSL-KDD Dataset

The experiments are conducted using the NSL-KDD dataset, which is a widely used benchmark for intrusion-detection research.

The current study considers the **multiclass classification problem**, where network connections are classified according to their corresponding attack category.

The experimental pipeline consists of the following main stages:

1. Dataset loading and preparation;
2. Data preprocessing;
3. Feature scaling;
4. Feature selection;
5. Class balancing;
6. Machine learning model training;
7. Model evaluation;
8. Multi-metric ranking;
9. Result visualization and analysis.

2.2 Feature Selection

Several feature-selection approaches have been incorporated into the experimental framework to investigate the effect of feature reduction on multiclass intrusion-detection performance. The methods used in the experiments are divided into two main categories: **filter methods** and **embedded methods**.

2.2.1 Filter Methods

Filter methods select features according to statistical properties or their relationship with the target variable, independently of a specific machine learning model. The following filter methods are considered:

- **Variance**: selects features according to their variability and removes features with low variance.
- **Chi-square (Chi2)**: evaluates the statistical association between feature values and the target classes.
- **ANOVA**: evaluates whether the distributions of feature values differ significantly across the target classes.
- **Mutual Information**: measures the statistical dependency between each feature and the target variable.

2.2.2 Embedded Methods

Embedded methods perform feature selection during the model-training process. The following embedded methods are used:

- **Random Forest**: uses feature importance derived from an ensemble of decision trees to identify informative features.
- **Extra Trees**: uses feature importance obtained from an Extremely Randomized Trees ensemble to identify relevant features.
- **Logistic L1**: uses L1 regularization to reduce the coefficients of less informative features toward zero, thereby selecting a subset of relevant features.

Overall, the experimental framework evaluates **seven feature-selection methods**: four filter methods and three embedded methods. This combination allows the study to compare statistical feature-selection approaches with model-based feature-selection approaches.

For each feature-selection experiment, the selected features and the resulting number of features are recorded. This makes it possible to analyze both the predictive performance of the resulting intrusion-detection model and the dimensionality reduction achieved by each feature-selection method.

2.3 Machine Learning Models

Several machine learning algorithms have been integrated into the framework, including:

- Logistic Regression;
- Random Forest;
- Extra Trees;
- Gradient Boosting;
- Decision Tree;
- K-Nearest Neighbors;
- Naive Bayes;
- AdaBoost;
- SGD-based classification.

The objective is to compare different machine learning approaches under a common experimental protocol.

2.4 Class-Balancing Strategies

Class imbalance is explicitly considered in the experimental study because the distribution of samples among the different intrusion classes can significantly affect the performance of machine learning classifiers. To investigate the effect of different imbalance-handling strategies, the experimental framework incorporates several approaches organized into four main categories: **oversampling**, **undersampling**, **hybrid methods**, and **cost-sensitive learning**.

2.4.1 Oversampling Methods

Oversampling methods increase the representation of minority classes by generating or replicating additional samples. The following methods are considered:

- **Random Over-Sampling**: randomly duplicates samples from minority classes.
- **SMOTE**: generates synthetic samples for minority classes based on neighboring samples.
- **Borderline-SMOTE**: focuses synthetic sample generation on minority samples located near the class decision boundary.
- **SVM-SMOTE**: uses support vector machine-based information to generate synthetic minority samples.
- **ADASYN**: adaptively generates synthetic samples, giving greater importance to minority samples that are more difficult to learn.

2.4.2 Undersampling Methods

Undersampling methods reduce the number of samples belonging to majority classes. The following approaches are used:

- **Random Under-Sampling**: randomly removes samples from majority classes.
- **NearMiss**: selects majority-class samples according to their distances from minority-class samples.
- **Tomek Links**: removes samples forming Tomek links to reduce overlapping between classes.
- **Edited Nearest Neighbours (ENN)**: removes samples that are inconsistent with their neighboring samples.

2.4.3 Hybrid Methods

Hybrid methods combine oversampling and undersampling techniques to improve the representation of minority classes while reducing potentially ambiguous majority-class samples. The following methods are considered:

- **SMOTE-Tomek**: combines SMOTE oversampling with Tomek Links to generate minority samples and reduce class overlap.
- **SMOTE-ENN**: combines SMOTE with Edited Nearest Neighbours to generate minority samples and remove potentially noisy or ambiguous samples.

2.4.4 Cost-Sensitive Learning

Cost-sensitive learning addresses class imbalance by modifying the learning process rather than explicitly changing the class distribution. The following approach is considered:

- **Class Weight**: assigns different weights to classes during model training, giving greater importance to underrepresented classes.

Overall, the experimental framework includes **twelve class-imbalance handling strategies**: five oversampling methods, four undersampling methods, two hybrid methods, and one cost-sensitive method.

The effect of these strategies is evaluated in combination with different feature-selection methods and machine learning models. This allows the study to determine whether and to what extent class-imbalance handling improves multiclass intrusion-detection performance on the NSL-KDD dataset.

Consequently, each experiment can be described by the combination:

$$\text{Model} + \text{Feature Selection} + \text{Balancing Method}.$$

This provides a systematic way to investigate the interaction between the three experimental dimensions.

Chapter 3

Evaluation and Model Ranking

3.1 Evaluation Metrics

The following metrics have been considered:

- Accuracy;
- Balanced Accuracy;
- Precision;
- Recall;
- F1-score;
- Specificity;
- Kappa;
- ROC-AUC;
- PR-AUC;
- Log Loss;
- Brier Score.

The use of multiple metrics is important because accuracy alone may not provide a complete assessment of an intrusion-detection system, particularly when the classes are imbalanced.

Some metrics, such as ROC-AUC and PR-AUC, may not be computable for all machine learning models or under all prediction configurations. The ranking methodology therefore explicitly handles unavailable metrics.

3.2 Normalized Ranking Method

A normalized weighted ranking method has been developed to compare experimental configurations.

For metrics for which higher values indicate better performance, the normalization is defined as:

$$N(x) = \frac{x - x_{\min}}{x_{\max} - x_{\min}}.$$

For metrics for which lower values indicate better performance, such as Log Loss and Brier Score, the normalization is:

$$N(x) = \frac{x_{\max} - x}{x_{\max} - x_{\min}}.$$

A metric is considered available for the global ranking when it is computed by at least one evaluated model.

For an individual model, unavailable metrics are excluded from its score. The weights of the metrics that are actually available for that model are automatically renormalized.

This approach allows models with different sets of available evaluation metrics to be compared without assigning an artificial penalty for unavailable metrics.

Chapter 4

Results Achieved So Far

4.1 Machine Learning Model Comparison

Preliminary experiments have been conducted using 20 selected features. A case for the obtained results is presented in Table 4.1.

Table 4.1: Preliminary machine learning results

Model	Accuracy	Bal. Acc.	F1	ROC-AUC	PR-AUC
Logistic Regression	0.807	0.704	0.805	0.940	0.885
Gradient Boosting	0.809	0.642	0.801	0.948	0.894
Random Forest	0.786	0.575	0.761	0.941	0.918
Extra Trees	0.781	0.567	0.756	0.941	0.912
SGD	0.757	0.675	0.757	–	–
KNN	0.783	0.620	0.766	0.785	0.693
Decision Tree	0.778	0.595	0.753	0.763	0.658
Naive Bayes	0.669	0.639	0.714	0.865	0.753
AdaBoost	0.663	0.625	0.649	0.903	0.816

These preliminary results show that model performance depends strongly on the evaluation criterion.

For instance, Gradient Boosting obtains the highest accuracy in the presented experiment, whereas Logistic Regression obtains a higher balanced accuracy and F1-score. This demonstrates the importance of using several complementary metrics.

4.2 Feature-Selection Results

A preliminary comparison of feature-selection methods is presented in Table 4.2.

Table 4.2: Preliminary feature-selection results

Method	Accuracy	Bal. Acc.	F1	ROC-AUC	PR-AUC
ANOVA	0.786	0.575	0.761	0.941	0.918
Logistic L1	0.783	0.608	0.759	0.934	0.897
Extra Trees	0.783	0.575	0.760	0.938	0.919
Mutual Information	0.776	0.582	0.734	0.954	0.923
Variance	0.761	0.601	0.721	0.957	0.925
Random Forest	0.765	0.572	0.720	0.952	0.921
Chi2	0.731	0.486	0.685	0.829	0.792

The preliminary results demonstrate that feature selection has a considerable influence on classification performance.

ANOVA, Logistic L1, Extra Trees, and Mutual Information produced competitive results in the experiment presented above. However, the final conclusions require the completion of the remaining experiments, particularly the comparison of balancing strategies.

4.3 Experimental Framework

A modular experimental framework has been developed to automate the execution of experiments.

The framework allows the following components to be varied independently:

- machine learning model;
- feature-selection method;
- balancing method;
- evaluation metrics.

The experimental results are automatically stored in CSV files, while the ranking results and visualizations can be generated automatically.

Several visualization approaches have also been implemented, including:

- feature-selection heatmaps;
- experimental configuration ranking plots;
- feature-selection comparison plots.

Chapter 5

Main Achievements

The main achievements obtained during the internship so far are:

1. Development of a modular NSL-KDD intrusion-detection experimental framework.
2. Integration of several machine learning classification algorithms.
3. Implementation and comparison of multiple feature-selection techniques.
4. Integration of Random Over-Sampling and SMOTE.
5. Implementation of a comprehensive multi-metric evaluation procedure.
6. Development of a normalized weighted ranking mechanism.
7. Implementation of automatic handling of unavailable evaluation metrics.
8. Automatic storage of experimental results in CSV format.
9. Development of visualization tools for comparative analysis.
10. Obtaining preliminary experimental results showing differences between models and feature-selection methods.

Chapter 6

Remaining Work

The internship is still in progress. The following activities remain to be completed:

1. Complete the experiments for all selected machine learning models.
2. Complete the comparison between Random Over-Sampling and SMOTE.
3. Perform a systematic comparison of all feature-selection methods.
4. Investigate the influence of the number of selected features.
5. Verify the stability of the obtained results.
6. Refine the multi-metric ranking methodology.
7. Investigate anomalous ranking results and verify their interpretation.
8. Generate the final comparative tables and figures.
9. Identify the most robust experimental configuration.
10. Prepare the results for scientific dissemination and publication.

Conclusion

This internship, funded by the **École Supérieure en Informatique (ESI-SBA), Sidi Bel Abbès**, and carried out in Tunis, Tunisia, from 6 August to 16 August 2026, has enabled significant progress in the study of machine-learning-based intrusion detection using the NSL-KDD dataset.

The work completed so far has resulted in a systematic experimental framework covering preprocessing, feature selection, class balancing, machine learning, multi-metric evaluation, ranking, and visualization.

The preliminary results demonstrate that intrusion-detection performance varies according to the machine learning algorithm, feature-selection technique, and evaluation metric. These observations justify the use of a multi-dimensional experimental methodology rather than relying on a single model or performance indicator.

The internship also has an important cultural dimension. The choice of Tunisia was motivated not just by scientific considerations but also by the desire to conduct research in a **Muslim country**.

The remaining work will focus on completing the experimental campaign, validating the preliminary findings, refining the ranking methodology, and identifying the most robust configuration for multiclass intrusion detection on the NSL-KDD dataset.