Final Project: Design and Evaluation of Machine Learning Models for Flow-Based Intrusion Detection Systems
Course: Basics of Machine Learning and Artificial Intelligence

1. Project Overview

Students are required to design, implement, and evaluate machine learning algorithms for the detection and classification of multiple types of **intrusion activity and cyber attacks** using labeled network traffic datasets. The datasets contain both benign traffic and diverse categories of malicious activity, including brute-force attacks, denial-of-service and distributed denial-of-service attacks, web-based attacks, infiltration attempts, botnet traffic, and other common intrusion patterns. Students must justify their modeling choices and analyze generalization capability, scalability, and potential limitations within the context of network intrusion detection systems.

2. Learning Outcomes
Upon successful completion, students will be able to:

a. Select and justify machine learning algorithms based on data properties.
c. Analyze bias and variance tradeoffs and model capacity.
d. Compare different models.
e. Design and execute rigorous experimental evaluations.
f. Reason about computational complexity and scalability.

3. Modeling Requirements
Each project must include the following modeling components:

- At least one baseline model, selected from the following:

  - Logistic Regression with regularization
  - k-Nearest Neighbors
  - Decision Tree

- At least one advanced model, selected from the following:

  - Random Forest
  - Gradient Boosting
  - XGBoost
  - Neural Networks

All model selections must be **theoretically justified**, with an accompanying discussion of **computational complexity and scalability considerations**. Models may be implemented using standard machine learning libraries or developed from scratch when appropriate.

To ensure **reproducibility**, all training and evaluation scripts must use fixed random seeds.

It is desirable to apply an anomaly detection algorithm introduced during the course and compare its performance with the supervised algorithms listed above.

4. Evaluation Protocol - These are the evaluations that students must show in their final project deliverables. Projects must be evaluated using the following quantitative metrics:

- Precision, Recall, and F1-score.

- Confusion matrices.
- ROC-AUC curves, with appropriate representation and evaluation strategies for multi-class classification problems.

5. Project Deliverables

   The primary objective of the project is to demonstrate understanding, deliberate decision-making, and analytical reasoning. All design choices, including feature selection, model selection, and experimental setup, must be clearly justified.

   Students must submit the following:

   - Technical report - The technical report should consist of **several pages** and should **not be written in the form of a traditional seminar paper** that describes algorithms or models in a theoretical manner. Instead, it must focus exclusively on **the experimental workflow**, providing a step-by-step narrative of how the project was conducted. Each stage of the project, including data preparation, model selection, training, evaluation, and analysis, should be described as a coherent progression of decisions and observations, emphasizing practical reasoning and experimentation rather than theoretical explanations.
   - A **12-minute** presentation summarizing the results.
     Power point presentation must include:
     - Problem formulation and dataset description
     - Feature engineering and key features
     - Model selection, theoretical justification, and complexity discussion
     - Evaluation methodology and experimental results
     - Analysis of model limitations and generalization of capabilities

   - A GitHub repository with read-only access granted to the teaching assistant (acc. *danilo789*), containing all scripts.

Detailed Grading Rubric

1. Feature Engineering (10 percent)
   - Feature preprocessing and normalization performed appropriately
   - Students should design additional features rather than relying solely on existing ones. These engineered features may include statistical flow features, such as the mean and variance of packet sizes, flow duration, and byte ratios, as well as temporal features, including inter-arrival times, burst statistics, and autocorrelation metrics, where such features are not already present in the dataset.
   - PCA transformation, where applicable
     For each applied data transformation, students should evaluate its impact on the final results and justify whether its use is appropriate.
2. Algorithmic Design (40 percent)
   - Implementation of at least one baseline model
   - Implementation of at least one advanced models.
   - Justification of model selection based on theoretical considerations
   - Discussion of computational complexity and algorithmic efficiency
   - Correct application of model training and hyperparameter selection procedures

3. Experimental Rigor (20 percent)
    - Evaluation metrics computed correctly: precision, recall, F1-score, ROC-AUC, confusion matrices
    - Proper train/test splits and handling of class imbalance
    - Reproducibility of results using submitted code and data
4. Analysis and Insight (20 percent)
    - Interpretation of feature importance or model coefficients
    - Discussion of limitations, generalization and potential biases
    - Identification of errors or misclassifications and proposed explanations
    - Dataset size sensitivity (scalability) analysis**:** Train each model on progressively larger subsets of the dataset and analyze how performance and computational cost change as the dataset grows.

5. Technical report (10 percent)

Dataset for project:

UNSW-NB15 - N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," 2015 Military Communications and Information Systems Conference (MilCIS), Canberra, ACT, Australia, 2015, pp. 1-6, doi: 10.1109/MilCIS.2015.7348942.

Original dataset: https://research.unsw.edu.au/projects/unsw-nb15-dataset

Kaggle url: https://www.kaggle.com/datasets/dhoogla/unswnb15?select=UNSW_NB15_training-set.parquet

Github: https://github.com/iammyr/encrypted-network-datasets

NetFlow-Derived Version -Standardized NetFlow feature sets derived from the this dataset provide flow-centric representations suitable for ML. staff.itee.uq.edu.au

The provided URLs and reference materials contain all the necessary information about the dataset. The primary objective is the detection and classification of cyber attacks in network traffic. While there is a substantial amount of publicly available material related to these dataset, students are encouraged to consult the teaching assistant if they encounter any issues with the data or have questions during their work.