Reproducibility of “Predictive Model” - Malware Dataset Generation and Evaluation (Assignment 1 - MDLE)

Miguel Miragaia,
Mining Large Scale Datasets
Student 108317, DETI
Universidade de Aveiro
Aveiro, Portugal
miguelmiragaia@ua.pt

***Abstract*—Reproducibility is essential in machine learning to validate research findings. This report attempts to replicate the methodology and results of the paper ”Malware Dataset Generation and Evaluation” using the TUANDROMD dataset. Five classifiers—Random Forest, Extra Trees, AdaBoost, XGBoost, and Gradient Boosting—were implemented for malware classification.**

Preprocessing steps included handling missing values, encoding labels, and normalizing features. A 10-fold stratified cross-validation was used, and performance was evaluated with accuracy, F1-score, precision, and recall. Visualizations such as confusion matrices and ROC curves were generated.

Results show that the reproduced models achieve comparable accuracy to the original study, though discrepancies in dataset preprocessing and class imbalance posed challenges. This highlights the need for greater transparency to improve reproducibility in machine learning research.

***Index Terms*—Reproducibility, Malware Detection, Machine Learning, Classification, TUANDROMD Dataset, Data Preprocessing, Model Evaluation, Cross-Validation.**

I. Introduction

Reproducibility is a fundamental aspect of scientific research, ensuring that experimental results can be validated and extended. In machine learning, reproducing studies helps assess the reliability of methodologies and findings [1].

The objective of this work was to replicate the methodology and application of a Data Mining process presented in the paper “Malware Dataset Generation and Evaluation” [2]. The paper introduces two datasets, **TUMALWD** and **TUANDROMD** and evaluates their effectiveness using machine learning classifiers: Random Forest, Extra Trees, AdaBoost, XGBoost, and Gradient Boosting.

With access only to the **TUANDROMD** dataset, the classifiers mentioned in the paper were implemented, and their accuracy results were compared with those reported in the study. The dataset was preprocessed by handling missing values, encoding categorical labels, and scaling numerical features. Model performance was assessed using accuracy, F1-score, precision, and recall.

This report aims to determine whether the methodology and results presented in the original paper can be successfully reproduced and to identify potential challenges in the replication process.

II. Methodology

In this section I describe the steps taken to preprocess the dataset, implement machine learning classifiers, and evaluate their performance. The intention was to follow the methodology described in the paper “Malware Dataset Generation and Evaluation” [2], but due to the lack of innumerous instructions certain decisions were made independently based on previous knowledge.

A. Data Preprocessing

The dataset TUANDROMD was loaded into a DataFrame for analysis and the following preprocessing steps were performed:

- Handling Missing Values: Missing values were replaced with the median of the respective column to maintain data consistency.
- Label Encoding: The dataset ”Label” column that classifies each sample as either malware or goodware was converted into numerical values, using `LabelEncoder` from `sklearn.preprocessing`:
  - Malware = 1
  - Goodware = 0
- Unexpected Classification: While encoding, an unexpected class with value 2 was identified, with only one occurrence. After a manual inspection I found that row 2535 was composed by empty values. The row was removed to ensure data integrity.
- Feature Normalization: To ensure that all features had the same scale, the `StandardScaler` from `sklearn.preprocessing` was applied.

This preprocessing phase is not mentioned in the paper, so the decisions regarding missing values, label encoding, and feature scaling were made independently.

A pie chart visualization Fig. 1 was created to represent the class distribution of malware and goodware.

TABLE I
CLASS DISTRIBUTION IN
THE TUANDROMD DATASET

Label	Count
Malware	3565
Goodware	899

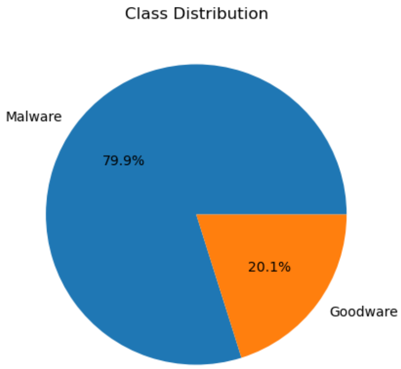

Fig. 1. Class Distribution in the TUANDROMD Dataset.

B. Dataset Splitting

In the paper the authors state that the dataset is ”not perfectly balanced”, but do not specify how they handled class imbalance. As shown in Fig. 1, the dataset is highly unbalanced. To mitigate this I applied class weighthing in classifiers that support it, such as Random Forest and Extra Trees (`class_weight='balanced'`). No additional balancing techniques were applied.

It is also never mentioned if the dataset was splitted into training and testing so, independently, I used `train_test_split` to divide it into training (80%) and testing (20%) sets.

A learning curve analysis was later be used to evaluate how model performance changes with different amounts of training data.

C. Implementation of Machine Learning Classifiers

The five classifiers evaluated in the original paper were implemented using the Scikit-learn and XGBoost libraries: **Random Forest**, **Extra Trees**, **AdaBoost**, **XGBoost**, **Gradient Boosting**.

Since the paper does not specify hyperparameters for these models, the parameters were chosen with the help of ChatGPT and existing best practices in the field: It is referred the use of 10-Fold Stratified Cross-Validation, so this was the approach to ensure that both malware and goodware samples were well-represented in training and testing splits. Performance was assessed using the following metrics:

- **Accuracy** - Overall correctness of classification.
- **F1 Score** - Balances precision and recall.
- **Precision** - Measures how many predicted malware samples are actually malware.
- **Recall** - Measures how many actual malware samples were correctly identified.

D. Performance Evaluation and Visualization

After training and evaluating the classifiers, results were visualized using various plots to analyze the models' effectiveness. A bar chart was used to compare all the metrics across classifiers. Confusion matrices were created to show the number of correctly and incorrectly classified samples for each class. The ROC curves illustrate the trade-off between the true positive rate and the false positive rate, and learning curves were used to analyze how model performance changes with different amounts of training data.

E. Results

To assess reproducibility, the classification results obtained in this study were compared with those reported in the original paper. Table II presents a comparison of the accuracy values for each classifier.

TABLE II
COMPARISON OF CLASSIFIER ACCURACY WITH PAPER RESULTS

Classifier	Reported Accuracy	Obtained Accuracy
Random Forest	98.7%	99.55%
Extra Trees	98.8%	99.55%
AdaBoost	97.9%	97.98%
XGBoost	97.8%	99.44%
Gradient Boosting	97.4%	98.66%

The results I obtained are higher than those reported in the original paper across all classifiers. These improvements may be a result of the existing variations of the dataset presented in the paper to the one I had access, and also some different processes taken.

III. CONCLUSION

Reproducing machine learning methodologies requires transparency and detailed documentation of every step in the process. Given the limited information provided in the original paper, it was impossible to fully replicate the methodology and results presented by the authors. Key details regarding data preprocessing, train-test splitting, and hyperparameter selection were not addressed, making independent implementation necessary. Additionally, the dataset used in this study presents some differences from the one referenced in the paper.

To ensure the transparency of machine learning experiments, authors should provide clear descriptions of preprocessing steps, data splitting strategies, and model configurations. In this case, only accuracy values were reported, and no additional performance metrics such as F1-score, precision, and recall were given.

For reproducibility to be effective, research papers should offer complete datasets, parameter settings, and a full documented methodology. Future work in this area should emphasize transparency to enable fair comparisons and the validation of results.

References

[1] S. Sawla, “Ensuring Reproducibility in ML: Why and How,” LinkedIn, 2023. [Online]. Available: https://www.linkedin.com/pulse/ensuring-reproducibility-ml-why-how-srishti-sawla. [Accessed: Mar. 2, 2025].

[2] D. K. Borah, S. Sharma, P. Hazarika, S. K. Dutta, and B. K. Bhattacharyya, “Malware Dataset Generation and Evaluation,” IEEE Xplore, 2020. [Online]. Available: https://sci-hub.se/downloads/2021-05-28/7c/borah2020.pdf. [Accessed: Mar. 2, 2025].