Project Title The Design and Implementation of an Intrusion Detection System Using Machine Learning Techniques

By Usiwoghene Ekebor

Introduction

Intrusion detection systems protect networks from malicious access. This project focuses on Denial-of-Service attacks in Wireless Sensor Networks, whose devices faces energy and memory limits. Using the WSN-DS dataset, four machine learning models were tested. Results show accurate detection of Blackhole, Grayhole, Flooding, and TDMA attacks, which could be implemented in real time.

Aims & Objectives

The aim of this project is to design and implementable machine learning-based intrusion detection system that accurately detects modern cyberattacks in TSNs, smart homes, and WSNs with improved real-time performance and low false positives.

Objective

- Develop a model with a labelled dataset containing normal and attack traffic
- Evaluate and compare machine learning (ML) algorithms for intrusion detection performance.
- Implement feature selection and oversampling techniques to handle data imbalance.
- Test the IDS in a simulated or real-time environment.
- Minimise false positives while maintaining high accuracy and low latency detection.

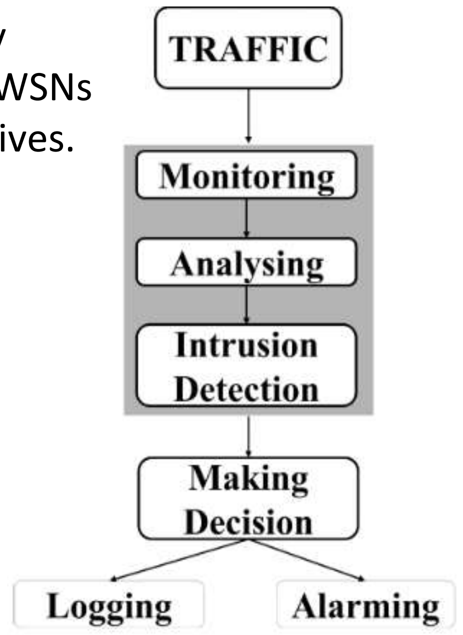


Fig 1. Intrusion Detection System flowchart

Methods

This project used the WSN-DS dataset from Security Engineering Lab (SEL) of Prince Sultan University which has 19 distinct features and 374,661 records covering four DoS attacks and normal traffic. The models of this project were trained in Google Colab Premium IDE with 12GB RAM and 250GB storage utilising Python libraries such as NumPy, Matplotlib, Sklearn, and TensorFlow/Keras for ANN and CNN etc. The develpment process involve the loading of the dataset which was uploaded to google drive, preprocess the set and trained selected ML models and finally evaluate and compare results amongst the trained models.

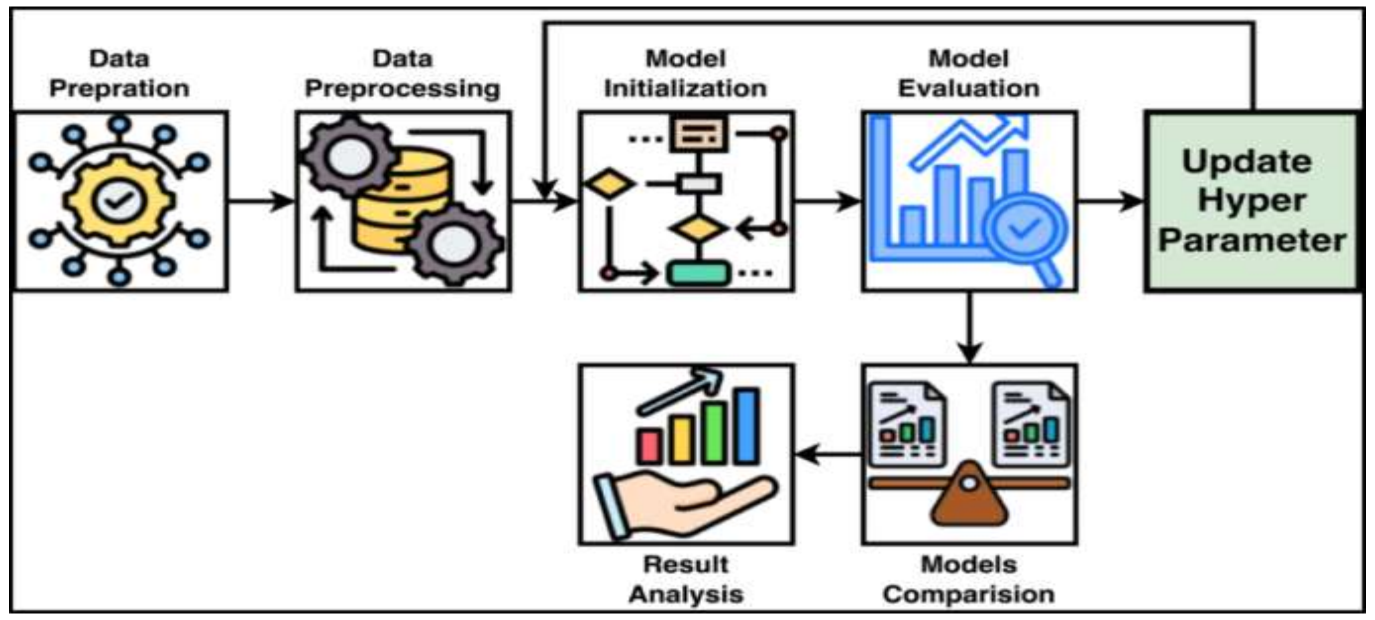


Fig. 2 Methodology Flowchart

Preprocessing Techniques

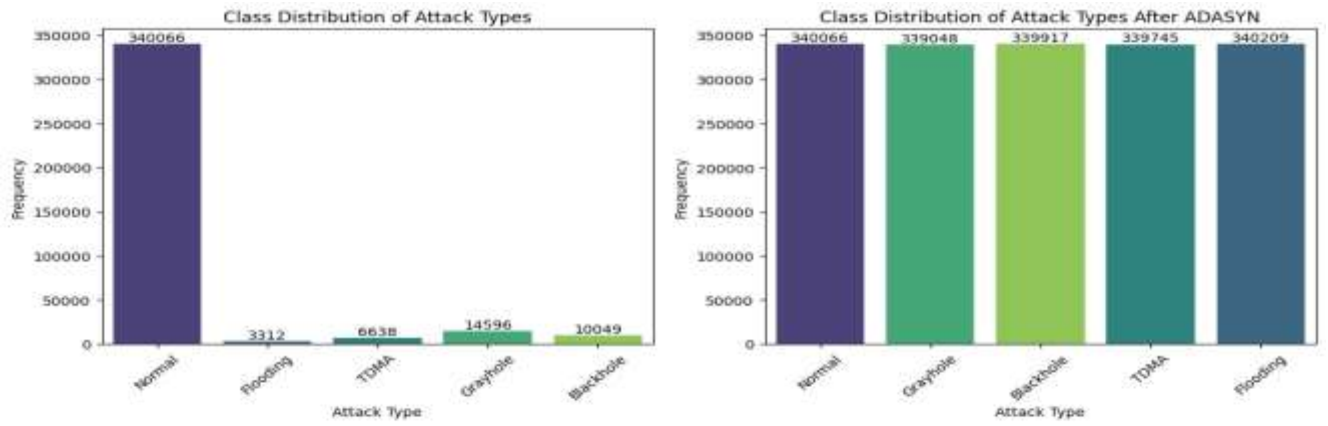


Fig 3 Class distribution before and after oversampling with ADASYN

The dataset was first prepared by calculating correlations, removing irrelevant columns, and handling imbalance with ADASYN. It was then split into 70% training and 30% testing. Mutual information was applied to reduce 18 features to 13 important ones. These selected features were scaled and prepared for model training.

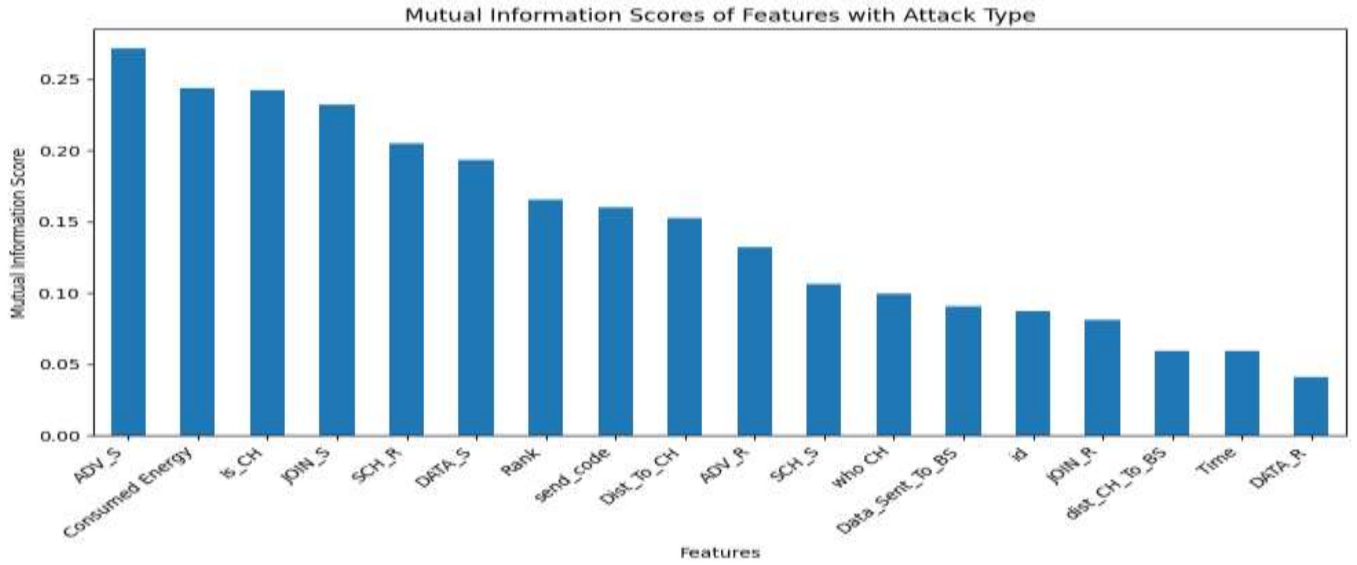


Fig. 4 Mutual information score of feature importance

Four ML models, ANN, CNN, DT, and XGB, were trained on the processed dataset and achieved strong accuracy scores. To further improve detection, two feature engineering methods, PCA and LDA, were applied. Retraining with these techniques produced even higher accuracy for some ML models.

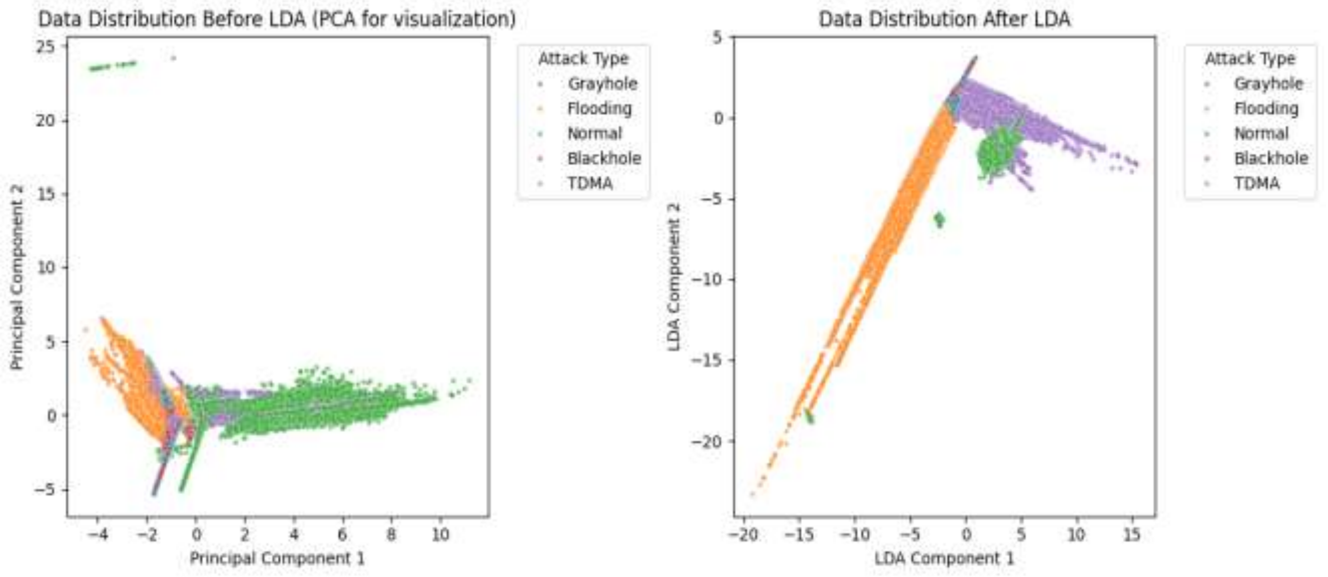


Fig. 5 Visualization the data distribution of PCA and LDA

Predicted Outcomes

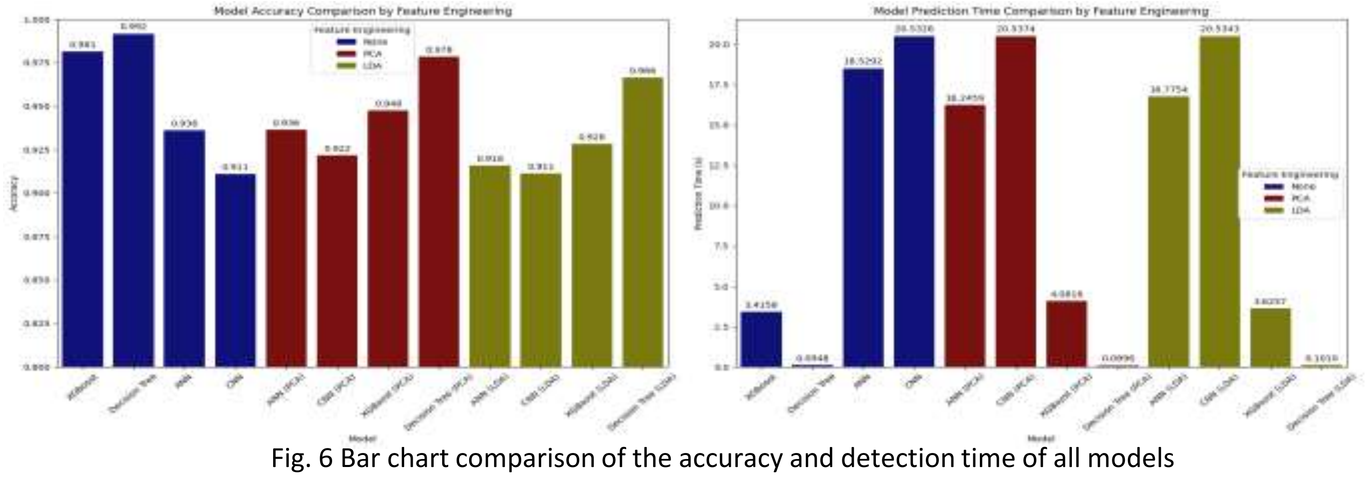


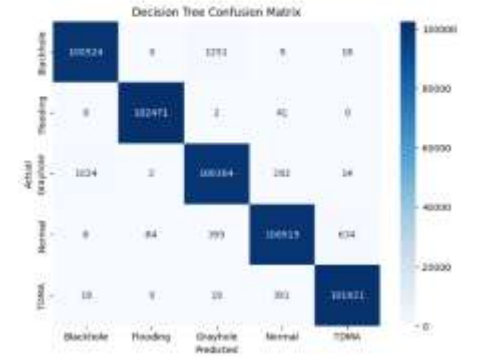
Fig. 6 Bar chart comparison of the accuracy and detection time of all models

The models trained in this project achieved high accuracy across all instances with very low detection times than earlier studies. Fares et al. (2025) reported detection times of 29, 27, and 33 seconds on models trained, while the highest detection time in this project is 20.5 seconds with strong accuracy across all the models.

Consequently, each model was trained three times with different preprocessing techniques, and the best result for each trained model is highlighted in the bar chart above. ANN (PCA) achieved 93% accuracy in 16.2 seconds. CNN (PCA) scored 92% with 20.5 seconds. XGB and DT reached 98% and 99% accuracy with very fast detection times of 3.4 and 0.09 seconds.

Lesson Learned

The project showed that careful feature selection and balancing improve accuracy and detection speed. While ANN and CNN benefit from PCA feature engineering, XGB and DT perform best with mutual information feature selection and ADASYN for oversampling.



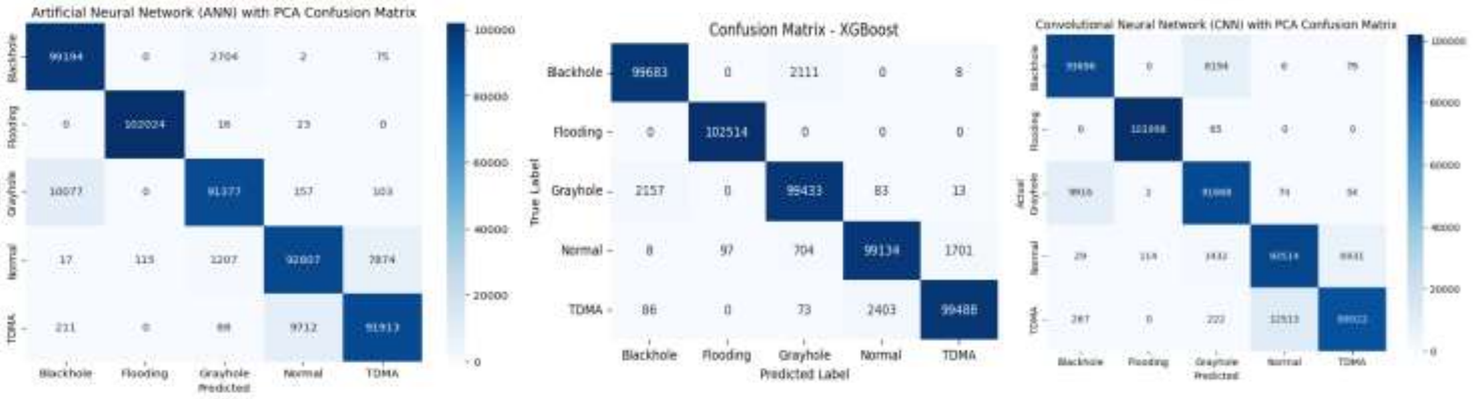


Fig. 7 Confusion matrix of best result of ANN, CNN, XGB and DT

Acknowledgements

Firstly, I thank God Almighty for his strength and wisdom throughout this project. Also, my sincere gratitude goes to my project supervisor, M. S. Mekala, for his support and valuable guidance which made the entire project simple and easy to understand. Finally, I want to extend my special appreciation to the Security Engineering Lab (SEL) team at Prince Sultan University, Saudi Arabia, for generously sharing the WSN-DS dataset, which played an important role in the successful development and completion of this project.