A Minor Project Synopsis on

Network Anomaly detection using efficient Machine Learning techniques

Submitted to Manipal University, Jaipur

Towards the partial fulfillment for the Award of the Degree of

BACHELOR OF TECHNOLOGY

In Computers Science and Engineering

2023-2024

By

Pradyumna Singh

209301309

Ansh Ujlayan

209301050

MANIPAL UNIVERSITY
JAIPUR

Under the guidance of

Lav Upadhyay

Department of Computer Science and Engineering

School of Computer Science and Engineering

Manipal University Jaipur

Jaipur, Rajasthan

INTRODUCTION

In today's world, the reliance on technology and network systems has become ubiquitous. With an ever-increasing number of devices connected to these systems, it's essential to have the ability to identify and respond to any unusual events in real-time. That's where our project comes in. We aim to develop a cutting-edge machine learning solution that can effectively and efficiently detect network issues as they arise.

The primary focus of this project is to find and address any network problems before they escalate into bigger problems that might cause significant damage. Our team will conduct thorough research on various machine learning algorithms and techniques to determine the most suitable approach to solving this issue. The outcome of this research will be documented in a comprehensive research paper, which will provide valuable insights into the problem and how our solution can help.

Aside from the research component, we will also be developing a practical application that leverages the findings from our research. This application will allow network administrators to quickly identify any network issues, allowing them to respond to them promptly. The goal of this project is to make a real difference in the world of technology by helping to ensure the smooth and uninterrupted operation of network systems.

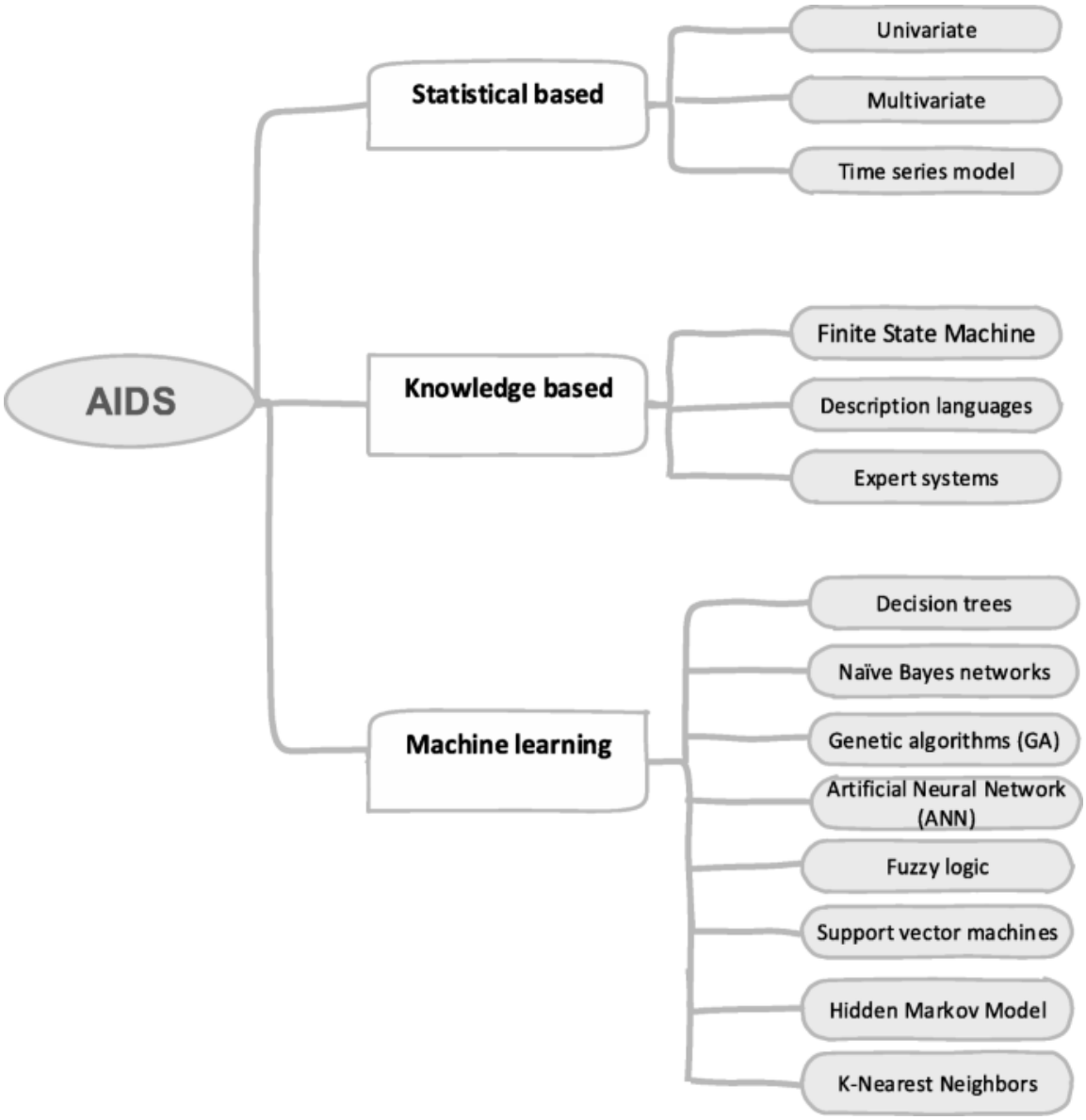
MOTIVATION

Network anomalies can seriously harm a system, resulting in security breaches, data loss, and system degradation. Therefore, it is important to detect these anomalies in real time. Current methods for detecting network anomalies are limited and can be time-consuming. This project aims to overcome this limitation by applying effective machine learning techniques to detect anomalous networks. The use of machine learning algorithms in network anomaly detection will provide a faster and more accurate method for detecting network anomalies.

PROJECT OBJECTIVE

The objective of this project is to develop a sophisticated machine learning model that can detect network anomalies with high accuracy. The model will be designed to consider various factors such as network traffic, system logs, and other relevant data, to provide a comprehensive understanding of network behavior. This is a crucial step towards ensuring the security and stability of network systems.

Existing Methods:



The project involves using effective machine learning techniques for network anomaly detection. This requires a thorough understanding of the CICIDS2018 dataset, which contains a large number of network traffic features and labels for various types of network attacks, including botnet attacks.

The project aims to use the MLP model for detecting botnet attacks, which is a type of artificial neural network that is well-suited for classification tasks. The implementation of the MLP model involves selecting appropriate hyperparameters and applying L2 regularization to prevent overfitting.

The project also involves evaluating the performance of the MLP model using standard metrics such as AUC, accuracy, precision, and recall. A perfect score of 1 for AUC indicates that the model can accurately distinguish between positive and negative instances, and a high level of accuracy, precision, and recall indicates that the model is effective in detecting botnet attacks.

However, the project also acknowledges the shortcomings of the study, which include a lack of detail and the use of only one classifier for evaluation. To address these issues, future research in this area could focus on providing more comprehensive and detailed evaluations of machine learning models for network anomaly detection, as well as exploring the use of other classifiers and techniques for improving performance.

Authors	Proposed model(s)	Computing environment	Accuracy	Precision	Recall	AUC
Atefinia and Ahmadi [14]	Modular Deep Neural Network	Python, Scikit-learn	100.00	100.00	100.00	n/a
Fitni and Ramli [22]	Logistic Regression + Decision Tree + Gradient Boosting	GPU (on some PCs)	98.80	98.80	97.10	0.94
Huancayo Ramos et al. [25]	Random Forest, Decision Tree	Python, Scikit-learn	99.99	100.00	99.99	n/a
Kanimozhi and Jacob [26]	MLP	Python, Scikit-learn	100.00	100.00	100.00	1
Karatas et al. [28]	Adaboost	GPU, Keras, Python, Scikit-learn, Tensorflow	99.69	99.70	99.69	n/a

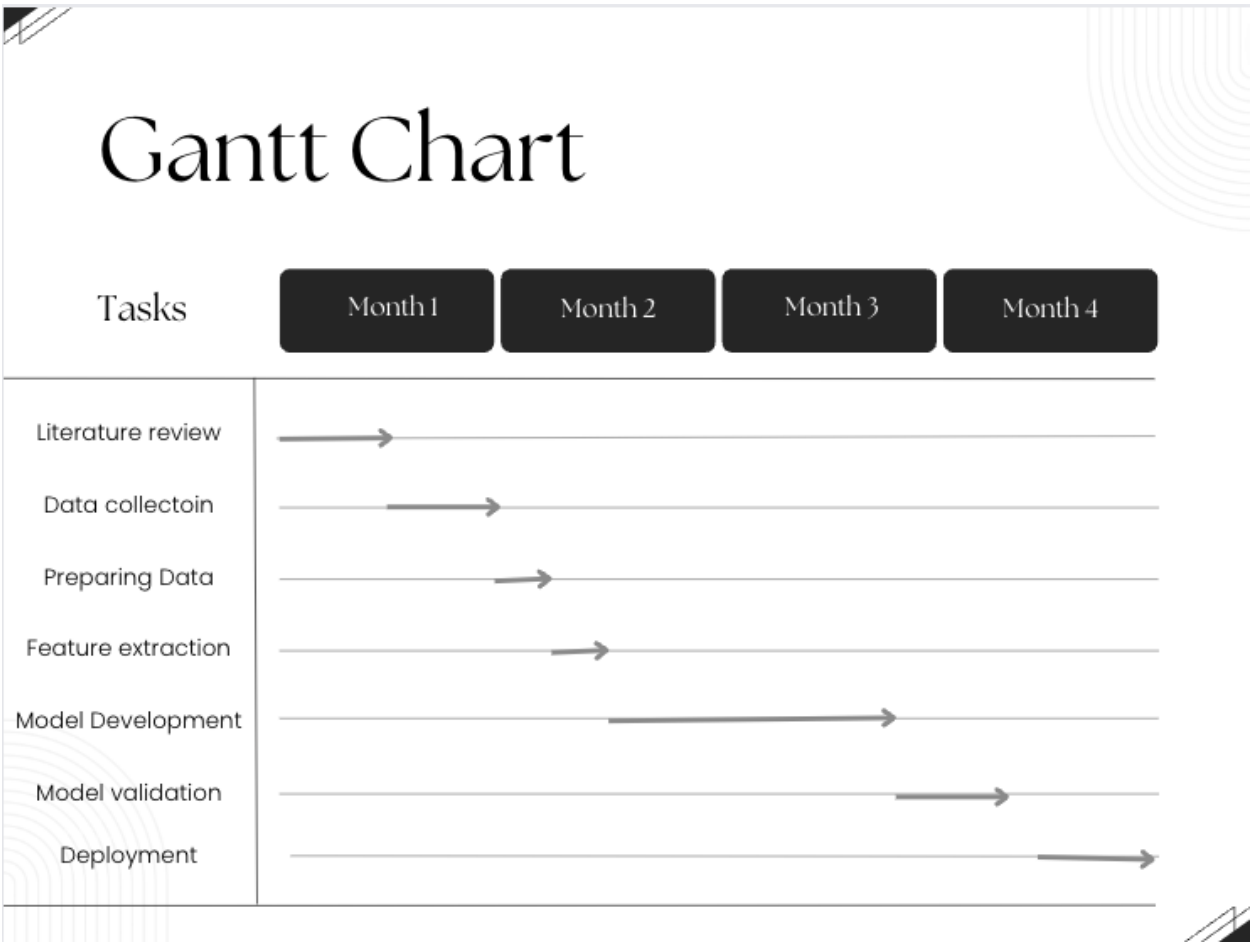
Kanimozhi and Jacob proposed a Multilayer Perceptron (MLP) model implemented with Python and Scikit-learn. This model achieved 100% accuracy, precision, and recall scores. The other models discussed in the given data are Atefinia and Ahmadi's Modular Deep Neural Network, Fitni and Ramli's ensemble model using Logistic Regression, Decision Tree, and Gradient Boosting, and Huancayo Ramos et al.'s comparison of RF and Decision Tree classifiers. All models have high accuracy, precision, and recall scores, but there are some limitations. For instance, Atefinia and Ahmadi did not evaluate the performance of the proposed model with multiple models, while Fitni and Ramli's ensemble model could benefit from using other gradient-boosting classifiers. Meanwhile, the study by Huancayo Ramos et al. did not test other classifiers beyond RF and Decision Tree. Overall, Kanimozhi and Jacob's MLP model seems to be the best model among the mentioned models, based on its perfect accuracy, precision, and recall scores.

METHODOLOGY

To achieve our objective, we will follow a structured methodology that involves several key steps.

1. Literature review: To gain a better understanding of how intrusion detection systems work, we will begin by conducting a literature review to survey existing research in the field.
2. Data collection: The second step will be to collect relevant data such as network traffic, system logs, and other relevant data.
3. Preparing data for analysis: Preprocess the collected data by removing irrelevant and missing values to ensure data quality and integrity.
4. Feature extraction: Next, we will extract relevant features from the collected data that will be used for anomaly detection. This step is crucial as it helps to narrow down the data to the most relevant information, making it easier for the machine learning model to detect anomalies.
5. Model development: We will then develop a machine learning model that can effectively detect network anomalies. This will involve training the model on large amounts of data and fine-tuning it to ensure that it provides accurate results.
6. Model validation: Once the model has been developed, we will validate it using real-world data. This step will help us determine the effectiveness of the model and make any necessary adjustments to ensure its accuracy.
7. Deployment: The final step involves implementing the model in a network system, where it can be used to detect anomalies and improve the overall system security.

The project timeline will be presented using a Gantt chart and specific timelines, allowing us to track our progress and ensure that the project stays on track. The use of these tools will also allow us to make any necessary adjustments to our timeline as needed, ensuring that we deliver the best possible solution for detecting network anomalies.



FACILITIES REQUIRED FOR PROPOSED WORK

Carrying out this project requires access to a range of software and hardware tools that will allow us to collect, process, analyze, and visualize data, as well as develop and deploy a machine learning model.

Software Requirements:
The software required for this project includes programming languages such as Python and R, which are widely used in the field of data science and machine learning. These languages will be used to write code for collecting, processing, and analyzing data, as well as for developing and deploying the machine learning model.

In addition to programming languages, we will also need access to machine learning libraries such as TensorFlow and scikit-learn. These libraries provide a range of pre-built algorithms and tools that can be used to develop machine learning models, saving us time and effort compared to developing these algorithms from scratch.

Data visualization is also an important aspect of this project, and we will use tools such as Matplotlib and Seaborn to visualize data and understand the behavior of our machine learning model.

Hardware Requirements:
In terms of hardware, we will need access to a computer with enough RAM, storage, and processing power. This will ensure that we can handle the large amounts of data that we will be working with, and that we can perform the necessary computations in a reasonable amount of time.

In conclusion, access to the right software and hardware tools is essential for successfully carrying out this project. By having these tools at our disposal, we will be able to collect, process, analyze, and visualize data, as well as develop and deploy a machine learning model that can detect network anomalies with high accuracy and in real-time.

REFERENCES

1. https://www.unb.ca/cic/datasets/ids-2018.html
2. https://www.unb.ca/cic/datasets/ids-2017.html
3. https://journalofbigdata.springeropen.com/articles/10.1186/s40537-021-00475-1
4. https://www.usenix.org/legacy/event/sysml07/tech/full_papers/ahmed/ahmed_html/sysml07CR_07.html
5. https://journalofbigdata.springeropen.com/articles/10.1186/s40537-020-00382-x#Bib1
6. https://www.sciencedirect.com/science/article/pii/S1877050920307961
7. https://www.sciencedirect.com/science/article/pii/S1877050920311121
8. https://www.sciencedirect.com/science/article/abs/pii/S1568494620302416