Date: 03/13/2024

PREPARED BY: GROUP F-33

Adit Sandeep Virkar

Arnav Raviraj

Shivanjay Vilas Wagh

Vinay Kantilal Chavhan

Project Plan

5G-NIDD: A Comprehensive Network Intrusion Detection Dataset Generated over 5G Wireless Network

Table of Contents

[Project Description 2](#_heading=h.gjdgxs)

[Overview: 2](#_heading=h.30j0zll)

[Key Requirements:](#_heading=h.1fob9te) 3

[Deliverables:](#_heading=h.3znysh7) 4

[Acronyms and Abbreviations :](#_heading=h.2et92p0) 5

[Design and Architecture](#_heading=h.tyjcwt) 6

[Implementation Strategy](#_heading=h.3dy6vkm) 10

[High-level Work Breakdown Structure :](#_heading=h.1t3h5sf) 10

[Schedule / Timeline :](#_heading=h.4d34og8) 18

[Required Hardware :](#_heading=h.2s8eyo1) 19

[Required Software :](#_heading=h.17dp8vu) 19

[Third Party Content / Libraries :](#_heading=h.3rdcrjn) 20

[Quality:](#_heading=h.26in1rg) 21

[Other Special Considerations:](#_heading=h.lnxbz9) 21

[Process](#_heading=h.35nkun2) 22

[Process Description and justification](#_heading=h.1ksv4uv) 22

[Tools :](#_heading=h.44sinio) 22

[Roles and Responsibilities:](#_heading=h.2jxsxqh) 23

[Location of Project Artifacts:](#_heading=h.z337ya) 24

[Sponsor Communications:](#_heading=h.3j2qqm3) 24

# **Project Description**

## **Overview:**

* **Statement of work:** Our topic for the capstone project is “ Detecting intrusion in softwarized 5G networks using machine learning”. Our sponsor, Dr. Abdallah Moubayed, assigned our team a research paper, “5D-NIDD: A comprehensive network Intrusion detection dataset generated over 5G Wireless network”, which provides a detailed exposition of the creation, characteristics, and analysis of a novel dataset, 5G-NIDD, developed for network intrusion detection research in 5G environments. The dataset is distinguished by its generation from a real, operational 5G test network, offering a unique and realistic benchmark for evaluating AI/ML-based security solutions. This dataset addresses the scarcity of comprehensive, current, and realistic datasets for network security research, especially in the context of the emerging 5G networks, characterized by their complexity, scale, and diverse service offerings.
* **Final Deliverables**: Our object is to perform various data processing techniques, such as preprocessing the 5G-NIDD dataset by removing all the null and zero columns to improve the accuracy and performance of the machine learning models. With this preprocessed data, we must apply various correlation techniques to extract the relevant features and normalize the dataset with extracted features. After this, we feed our processed dataset to the machine-learning models and analyze their performance and accuracy. We have to achieve a better result than what is presented in the assigned paper.
* **Business Domain:** The business domain of our research is network security within the context of 5G wireless technology. We must address the critical need for sophisticated and proactive security mechanisms to identify and mitigate intelligent attacks on the complex and heterogeneous infrastructure of 5G networks. The increasing reliance on 5G technology across various sectors, including critical infrastructure, healthcare, and IoT, underscores the importance of this research domain
* **Technical Platform:** As our work is primarily based on a dataset, 5G-NIDD, which is generated from a real 5G test network located in Oulu, Finland, which is part of the 5G Test Network Finland (5GTN). This network provides a realistic environment for generating network traffic, both benign and malicious, under controlled conditions. The dataset includes network flows derived from various attack scenarios, such as Distributed Denial of Service (DDoS) attacks, ICMP floods, UDP floods, and HTTP floods, among others, alongside normal traffic. Tools and technologies include Argus for flow data generation, Tracewrangler for data processing, and various attack simulation tools like hping3 and GoldenEye. As per our requirement, we only have to utilize the generated dataset and feed it to the machine learning models after processing it.
* **Motivation of work:** Building upon the research described in "5G-NIDD: A Comprehensive Network Intrusion Detection Dataset Generated over 5G Wireless Network," our motivation of work focuses on several vital enhancements and expansions to further the field of network security, particularly within the rapidly evolving 5G and future 6G environments such as the expansion of threat models, dynamic and real-time data labeling, incorporation of AI/ML for active defense mechanisms, distributed detection and federated learning, testbed and dataset as a service and research beyond 5G security research.

## **Key Requirements:**

* **Functional Requirements:**
* Data Preprocessing: The solution must be capable of preprocessing the 5G-NIDD dataset by
  + Identifying and removing all null and zero-value columns to ensure data quality.
  + Applying correlation techniques to identify and extract relevant features from the dataset.
  + Normalizing the dataset based on the extracted features prepares it for machine learning analysis.
* Machine Learning Model Integration: Implement machine learning models that can be trained on the preprocessed 5G-NIDD dataset to detect network intrusions.
  + The models must support classification tasks to distinguish between benign and malicious network traffic.
  + The solution should enable model performance evaluation using accuracy, precision, recall, and F1 score metrics.
* Performance Analysis: The system must include functionality to analyze and report the performance of the machine learning models.
  + Comparison against baseline results presented in the assigned research paper to determine improvement.
  + Detailed model performance analysis under various attack scenarios mentioned in the dataset, such as DDoS, ICMP floods, UDP floods, and HTTP floods.
* **Technical Requirement:**
* Dataset Handling: The technical platform must support efficient handling of the 5G-NIDD dataset, including:
  + Secure storage and access mechanisms for the dataset.
  + Capability to process large volumes of network flow data generated from an actual 5G test network.
* Machine Learning Tools and Libraries: Utilize appropriate machine learning libraries and frameworks compatible with the dataset's characteristics and the project's goals.
  + The chosen libraries must support the preprocessing techniques, feature extraction, normalization, and model evaluation criteria outlined in the functional requirements.
* Scalability and Extensibility: Design the solution with scalability in mind to accommodate future enhancements, such as
  + Expanding the dataset with additional attack scenarios or more complex network environments.
  + Integrating additional machine learning models or deep learning approaches for enhanced detection capabilities.
* **Performance Requirement:**
  + Accuracy Improvement: The machine learning models must achieve a higher accuracy in detecting network intrusions than the baseline presented in the research paper.
  + Specific accuracy targets may be set based on preliminary analysis and the performance of existing models documented in the research paper.
  + Processing Time: The preprocessing and analysis processes should be optimized for efficiency.
  + Define maximum acceptable processing times for data preprocessing and model training phases.
  + Resource Utilization: Ensure that the solution is optimized for minimal and efficient use of computational resources, considering the complexity and volume of the 5G-NIDD dataset.

## **Deliverables:**

* Sprint 1:
  + Develop an in-depth understanding of the assigned research paper, “5G-NIDD: A Comprehensive Network Intrusion Detection Dataset Generated over 5G Wireless Network.”
  + Understanding the reference papers on 5G and 6G intrusion detection systems will help you understand the requirements to improve the performance and accuracy of machine learning models.
* Sprint 2:
  + Perform Data Preprocessing on the 5G-NIDD dataset, such as finding correlation coefficient, input sampling, and data normalization.
  + Apply boosting algorithms to the processed data and analyze the accuracy.
* Sprint 3:
  + Exploration of Feature selection on the provided 5G-NIDD dataset.
  + Improve Boosting model performance by feeding processed data into the boosting algorithm.
  + Implement ANN (Artificial Neural Networks) based model to analyze the accuracy and performance.
  + Implement a GAN (Generative Adversarial Networks) based model to analyze the accuracy and performance.
* Sprint 4 (till date):
  + Implement feature selection and input sampling techniques on the dataset.
  + Explore and Implement various boosting algorithms using multi-classification.
  + Implement a GAN-based model by applying multi-classification.
  + Implement ANN models using TensorFlow and modify the metrics.
* Final Deliverables:
  + This is a comprehensive report detailing the data preprocessing steps, feature extraction, and normalization techniques, machine learning models used, their performance analysis, and a comparison with the baseline results.
  + A software solution that includes all the code for data preprocessing, machine learning model implementation, performance analysis, and documentation for setup, configuration, and usage.

## 

## **Acronyms and Abbreviations:**

|  |  |
| --- | --- |
| **Glossary of Abbreviations and Acronyms** | **Definition** |
| 5G-NIDD | 5th Generation Network Intrusion Detection System |
| 5GTN | 5G Test Network |
| AdaBoost | Adaptive Boosting |
| AI | Artificial Intelligence |
| ANN | Artificial Neural Networks |
| CatBoost | Categorical Boosting |
| DoS | Denial of Service |
| GAN | Generative Adversarial Networks |
| IDS | Intrusion Detection System |
| ML | Machine Learning |
| ReLU | Rectified Linear Unit |
| XgBoost | Extreme Gradient Boosting |

Table 1: Acronyms and Abbreviations

# Implementation Strategy

## High-level Work Breakdown Structure:

The Work Breakdown Structure is divided into Sprints as follows.

**Sprint 1: Literature Review**

User Stories:

* Literature Review 1: Organize literature with precise categorization for easy access.
  + As a team member, I want the literature review to be well-organized and categorize studies precisely so that I can quickly locate information relevant to my task.
* Literature Review 2: Highlight critical theories and frameworks.
  + As a decision-maker, I want the literature review to highlight critical theories and frameworks relevant to our project so that we can make informed strategic decisions.
* Literature Review 3: Facilitate interdisciplinary connections.
  + As a collaborator, I want the literature review to facilitate interdisciplinary connections and highlight cross-cutting themes that can enhance the project's impact.
* Literature Review 4: Discuss the strengths and limitations of existing research.
  + As a team member, I want the literature review to include a discussion on the strengths and limitations of existing research so that we can identify areas where our project can make a unique contribution.

Tasks per User Story:

Literature Review 1:

* Task #7: Review the assigned paper.
  + Work Scope: Read and understand the assigned research paper.
  + Skillset Required: Research analysis, comprehension.
  + Expected Time: 2 days.
* Task #8: Find and read 2 related papers.
  + Work Scope: Search for and review two additional research papers on the assigned topic.
  + Skillset Required: Research skills, information retrieval.
  + Expected Time: 3 days.

Literature Review 2:

* Task #9: Find 3 related papers on different explored models.
  + Work Scope: Identify and review three research papers focusing on different models relevant to the project.
  + Skillset Required: Research skills, model understanding.
  + Expected Time: 4 days.
* Task #12: Understand the assigned research paper.
  + Work Scope: Deeply comprehend the concepts and findings presented in the assigned research paper.
  + Skillset Required: Research analysis, comprehension.
  + Expected Time: 2 days.

Literature Review 3:

* Task #11: Review the assigned paper.
  + Work Scope: Review and analyze the assigned research paper from another perspective.
  + Skillset Required: Research analysis, critical thinking.
  + Expected Time: 2 days.
* Task #14: Obtain 2 additional reference papers.
  + Work Scope: Search for and review two additional research papers from different disciplines to establish connections.
  + Skillset Required: Research skills and interdisciplinary understanding.
  + Expected Time: 3 days.

Literature Review 4:

* Task #10: Review the assigned research paper.
  + Work Scope: Review the assigned research paper and analyze its strengths and limitations.
  + Skillset Required: Research analysis, critical thinking.
  + Expected Time: 2 days.
* Task #13: Find a related research paper on a similar topic.
  + Work Scope: Search for and review another research paper discussing similar topics to identify further strengths and limitations.
  + Skillset Required: Research skills, critical analysis.
  + Expected Time: 3 days.

**Sprint 2: Model Evaluation and Optimization**

* User Stories:

Find the new model to apply to the encoded data.

* User Story 1:
  + As a team member involved in model evaluation, I want to establish robust validation strategies designed explicitly for encoded data analysis.
* User Story 2:
  + As a team member involved in model evaluation, I want to establish robust validation strategies designed explicitly for encoded data analysis.
* User Story 3:
  + As a team member involved in model interpretation, I want to develop techniques for interpreting and visualizing the output of encoded data models so that stakeholders can gain actionable insights and understand the rationale behind model predictions.
* User Story 4:
  + As a team member responsible for data preprocessing, I want to optimize the encoding process to minimize information loss and maximize data representation.

Tasks per User Story:

Tasks for User Story 1:

* - Task #22: Find and apply a new model to the current data.
  + Work Scope: Research and identify suitable models for encoded data analysis.
  + Skillset Required: Model evaluation, research.
  + Expected Time: 5 days.
* Task #24: Log the work and create documentation for it.
  + Work Scope: Document the process of finding and applying the new model.
  + Skillset Required: Documentation, data analysis.
  + Expected Time: Ongoing throughout the sprint.
* Task #28: Modify parameters and check.
  + Work Scope: Fine-tune model parameters to optimize performance.
  + Skillset Required: Model optimization, experimentation.
  + Expected Time: 3 days.

Tasks for User Story 2:

* Task #26: Find a new model other than the ones in the given research paper.
  + Work Scope: Explore alternative models beyond those provided in the research paper.
  + Skillset Required: Model evaluation, research.
  + Expected Time: 4 days.
* Task #27: Apply the new model to encoded data.
  + Work Scope: Implement the selected model on the encoded data.
  + Skillset Required: Model implementation, data analysis.
  + Expected Time: 5 days.
* Task #30: Keep the results stored in a document.
  + Work Scope: Record and store the outcomes of the model application.
  + Skillset Required: Documentation, data analysis.
  + Expected Time: Ongoing throughout the sprint.

Tasks for User Story 3:

* Task #23: Apply the model to the existing processed data.
  + Work Scope: Implement the model on the processed data.
  + Skillset Required: Model implementation, data analysis.
  + Expected Time: 4 days.
* Task #25: Modify the existing model parameters and check the results obtained.
  + Work Scope: Adjust model parameters to optimize performance.
  + Skillset Required: Model optimization and experimentation.
  + Expected Time: 3 days.
* Task #29: Create documentation and log the results.
  + Work Scope: Document the process and outcomes of the model application.
  + Skillset Required: Documentation, data analysis.
  + Expected Time: Ongoing throughout the sprint.

Tasks for User Story 4: Find the new model to apply to current encoded data 4:

* Task #19: Apply the model to currently processed data.
  + Work Scope: Implement the model on the processed data.
  + Skillset Required: Model implementation, data analysis.
  + Expected Time: 4 days.
* Task #20: Modify the current model parameters and check.
  + Work Scope: Adjust model parameters to optimize performance.
  + Skillset Required: Model optimization and experimentation.
  + Expected Time: 3 days.
* Task #21: Log the work and create documentation for it.
  + Work Scope: Document the process and outcomes of the model application.
  + Skillset Required: Documentation, data analysis.
  + Expected Time: Ongoing throughout the sprint.

**Sprint 3: Model Exploration**

**User Stories**:

* Improve boosting model performance: Enhance model performance to reduce interpretation time and increase accuracy.
  + As a team member involved in model improvement, I want to improve the model performance to reduce interpretation time and increase accuracy.
* Explore feature selection and data pre-processing: Select relevant features and data cleaning techniques.
  + As a team member responsible for data preprocessing, I want to select relevant features and data-cleaning techniques to feed the model high-quality data.
* Implement ANN model: Develop and explore ANN models.
  + As a team member involved in researching new models, I want to develop and explore ANN models to gain insights into the feasibility of neural networks for our dataset.
* Explore and implement GAN models: Develop a GAN model using adversarial techniques.
  + As a team member researching new models, I want to develop the GAN model so that we can train the model using adversarial techniques.

**Tasks per User Story:**

Improve boosting model performance**:**

* #22: Find and apply a new model to the current data.
  + Work Scope: Research and identify a new boosting model suitable for the dataset.
  + Skillset Required: Machine learning algorithms, model selection.
  + Expected Time: 3 days.
* #24: Log the work and create documentation for it.
  + Work Scope: Document the process of applying the new model and its outcomes.
  + Skillset Required: Documentation, data analysis.
  + Expected Time: Ongoing throughout the sprint.
* #28: Modify parameters and check.
  + Work Scope: Adjust model parameters to optimize performance and verify results.
  + Skillset Required: Hyperparameter tuning, model evaluation.
  + Expected Time: 2 days.

Explore feature selection and data pre-processing on the raw data**:**

* #26: Find a new model other than the ones in the given research paper.
  + Work Scope: Search and select alternative models beyond the ones in the provided research paper.
  + Skillset Required: Machine learning algorithms, model exploration.
  + Expected Time: 4 days.
* #27: Apply the new model to the current encoded data.
  + Work Scope: Implement the selected model on the encoded dataset.
  + Skillset Required: Machine learning implementation, data preprocessing.
  + Expected Time: 3 days.
* #30: Keep the results stored in a document.
  + Work Scope: Record and organize the results of applying the new model.
  + Skillset Required: Documentation, data analysis.
  + Expected Time: Ongoing throughout the sprint.

Implement the ANN model**:**

* #23: Apply the model to the existing processed data.
  + Work Scope: Implement the ANN model on the preprocessed dataset.
  + Skillset Required: Deep learning, model implementation.
  + Expected Time: 5 days.
* #25: Modify the existing model parameters and check the results obtained.
  + Work Scope: Fine-tune ANN model parameters to improve performance and validate results.
  + Skillset Required: Hyperparameter optimization, model evaluation.
  + Expected Time: 3 days.
* #29: Create documentation and Log the results.
  + Work Scope: Document the ANN model implementation process and its outcomes.
  + Skillset Required: Documentation, data analysis.
  + Expected Time: Ongoing throughout the sprint

Explore and implement GAN models**:**

* #19: Apply the model to currently processed data.
  + Work Scope: Implement the GAN model on the processed dataset.
  + Skillset Required: Deep learning, GAN implementation.
  + Expected Time: 6 days.
* #20: Modify the current model parameters and check.
  + Work Scope: Adjust GAN model parameters to optimize performance and validate results.
  + Skillset Required: Hyperparameter optimization, model evaluation.
  + Expected Time: 4 days.
* #21: Log the work and create documentation.
  + Work Scope: Document the GAN model implementation process and its outcomes.
  + Skillset Required: Documentation, data analysis.
  + Expected Time: Ongoing throughout the sprint.

**Sprint 4: More** **Model Improvement and Exploration**

**User Stories:**

* Implement feature selection and input sampling techniques:
  + As a developer, I want to implement sampling and feature selection as data preprocessing so that my teammates can use the preprocessed data and implement models to check its accuracy and performance.
* Implement various boosting models using multi-classification:
  + As a machine learning engineer, I want to explore Gradient Boosting Machines (GBM) for multiclass classification to effectively handle large datasets and complex relationships between features and classes.
* Implement a GAN-based model and apply multi-classification:
  + As a machine learning researcher, I want to design and implement a GAN architecture capable of generating realistic synthetic data across multiple classes to augment the training set for multiclass classification tasks.
* Implementation of ANN models using TensorFlow and work on modifying the metrics:
  + As a machine learning engineer, I want to design and implement a basic feedforward neural network architecture using TensorFlow, including input, hidden, and output layers, to perform multiclass classification tasks.

**Tasks for User Stories:**

Implement feature selection and input sampling techniques:

* Task #57: Perform feature selection for the top 10, 15, and 25 features.
  + Work Scope: Analyze the dataset and select top features based on relevance.
  + Skillset Required: Data preprocessing, feature engineering.
  + Expected Time: 2 days.
* Task #58: Apply input sampling techniques.
  + Work Scope: Implement various input sampling methods.
  + Skillset Required: Data preprocessing, machine learning techniques.
  + Expected Time: 3 days.
* Task #37: Document and log the results.
  + Work Scope: Record findings and outcomes of feature selection and sampling.
  + Skillset Required: Documentation, data analysis.
  + Expected Time: Ongoing throughout the sprint.

Implement various boosting models using multiclassification:

* Task #60: Conduct hyperparameter tuning and metric experimentation.
  + Work Scope: Fine-tune model parameters and evaluate performance metrics.
  + Skillset Required: Machine learning, hyperparameter optimization.
  + Expected Time: 5 days.
* Task #61: Apply multiclassification to existing models.
  + Work Scope: Extend boosting models to support multiclass classification.
  + Skillset Required: Machine learning algorithms, multiclass techniques.
  + Expected Time: 4 days.
* Task #62: Log the work and create new documentation.
  + Work Scope: Document the implementation process and outcomes.
  + Skillset Required: Documentation, data analysis.
  + Expected Time: Ongoing throughout the sprint.

Implement a GAN-based model and apply multiclassification:

* Task #63: Implement GAN based on multiclassification.
  + Work Scope: Develop GAN architecture for multiclass classification tasks.
  + Skillset Required: Deep learning, GAN implementation.
  + Expected Time: 7 days.
* Task #64: Apply classifiers other than CATBoost to the GAN-based models.
  + Work Scope: Experiment with various classifiers for GAN-generated data.
  + Skillset Required: Machine learning algorithms, GANs.
  + Expected Time: 6 days.
* Task #65: Create a document to store results.
  + Work Scope: Compile and organize results and findings.
  + Skillset Required: Documentation, data analysis.
  + Expected Time: Ongoing throughout the sprint.

Implementation of ANN models using TensorFlow and work on modifying the metrics:

* Task #66: Enhance the TensorFlow framework.
  + Work Scope: Improve existing TensorFlow architecture for ANN models.
  + Skillset Required: Deep learning, TensorFlow.
  + Expected Time: 4 days.
* Task #67: Experiment with parameters and metrics.
  + Work Scope: Explore different parameters and performance metrics for ANN models.
  + Skillset Required: Machine learning, experimentation.
  + Expected Time: 5 days.
* Task #68: Store all results in a document.
  + Work Scope: Compile and organize experimental results and metrics.
  + Skillset Required: Documentation, data analysis.
  + Expected Time: Ongoing throughout the sprint.

**Analysis and Discussion:**

We delve into implementing and analyzing machine learning models applied to our processed data. We discuss the performance and characteristics of each model, shedding light on their strengths and limitations.

**A. Boosting Algorithms:**

* **XGBoost:**

XGBoost, or eXtreme Gradient Boosting, is a robust and versatile algorithm widely used in machine learning and data science applications. Despite its slightly slower training time than a competitor, XGBoost exhibited superior accuracy and robustness. However, concerns regarding potential overfitting were noted.

* **CatBoost:**

CatBoost, known for its seamless handling of categorical variables and GPU acceleration support, performed comparably to XGBoost in terms of accuracy. Its ability to handle categorical features without preprocessing makes it well-suited for our classification task.

* **ADABoost:**

ADABoost, implemented for multi-class classification, demonstrated its effectiveness after preprocessing steps. Its ability to handle categorical variables using one-hot encoding and performance metrics such as precision, recall, and F1-score were evaluated.

B**. Artificial Neural Network (ANN):**

* **Fully Connected ANN:**

A fully connected artificial neural network architecture with multiple hidden layers and ReLU activation functions was designed. The Adam optimizer was chosen for efficient training, although further research is needed to fine-tune hyperparameters and prevent overfitting.

* **TensorFlow and Keras Implementation:**

Another ANN implementation utilizing TensorFlow and Keras for binary classification was developed. This implementation underwent preprocessing steps, including handling missing values and normalization. The model architecture, training parameters, and performance metrics were specified.

**C. GAN-Based Network**:

Generative Adversarial Networks (GANs) revolutionize data generation by pitting two neural networks against each other: the generator and the discriminator. This adversarial framework allows for generating realistic synthetic data, augmenting the original dataset for downstream tasks. The GAN implementation involved preprocessing steps, constructing the GAN architecture, training the generator and discriminator iteratively, and evaluating the efficacy of GAN-generated data in enhancing model performance.

This analysis provides insights into the performance and characteristics of the implemented machine learning models, guiding future research and development efforts. The table below shows the performance of each model, including those given in the reference research paper. [1]

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **Accuracy** | **Training Time(s)** | **Prediction Time(s)** |
| Decision Tree | 0.991678249 | 3.880316758 | 0.038118291 |
| Random Forest | 0.994812031 | 11.7032125 | 0.37769897 |
| KNN | 0.994835607 | 2.148859835 | 282.0652506 |
| Naive Bayes | 0.409909614 | 1.292910719 | 0.227512336 |
| MLP | 0.994992694 | 316.8805176 | 0.230028152 |
| XGBoost | 100 | 5.2565107345581055 | 1.236060619354248 |
| AdaBoost | 0.995578447 | 212.34 sec | 1.7 |
| CatBoost | 0.999757379 | 256.52 | 0.39 |
| GAN | 0.9996874717285281 | 75.77 | 0.16 |
| ANN(pytorch) | 0.61 | 51852.08 | 0.4864246845 |
| ANN(TensorFlow) | 0.63 | 1853.68 | 0.54 |

**Table 2: Model Accuracy**

**Schedule / Timeline:**

**Key Deliverables:**

* Improve current research on detecting attacks in 5G networks.
* Apply different data processing techniques and newer models to current research.
* Explore options to advance multi-classification models.
* Create final documentation and code solution.

**Milestones:**

* Read the given research paper and complete the literature review. This is mapped to Key Deliverables.
* Explore new models and apply them to the currently processed data. This is mapped to Key Deliverable 1.
* Work on ANN, GAN, and Boosting-based models. This is mapped to Key Deliverable 1.
* Apply sample-based techniques and work on data processing. This is mapped to Key Deliverable 2.
* Work on advancements in research and apply them to current models. This is mapped to Key Deliverable 3.
* Finalize and consolidate the work and focus on documentation. This is mapped to Key Deliverable 4.

**Timeline:**

**Sprint 1 – Research (23rd Jan 2024 – 6th Feb 2024):**

The goal of Sprint 1 was to complete the literature review, focusing on Milestone M1.

**Sprint 2 – Model Exploration (7th Feb 2024 – 20th Feb 2024):**

The goal of Sprint 2 was to find new models and apply them to the currently processed data, focusing on Milestone M2.

**Sprint 3 – Model Implementation (21st Feb 2024 – 3rd Mar 2024**):

The goal of Sprint 3 was to work on ANN, GAN, and Boosting models and apply them to the currently processed data, focusing on Milestone M3.

**Spring Break (4th March 2024 – 10th March 2024):**

The team utilized this break to explore other models that could be applied individually and to review other related research papers.

**Sprint 4 – Data Processing (11th March 2024 – 25th March 2024):**

The goal of Sprint 4 is to apply sample-based techniques and work on data processing technique options, focusing on Milestone M4.

**Sprint 5 - Advancement (26th March 2024 – 9th April 2024):**

The goal of Sprint 5 is to work on multiclassification models and explore advancements in current research, focusing on Milestone M5.

**Sprint 6 – Consolidation (10th April 2024 – 23rd April 2024):**

The goal of Sprint 6 is to consolidate the work and focus on the documentation part of the research, focusing on Milestone M6.

## Required Hardware:

An NVIDIA graphics card is required to develop and train the machine-learning model using CUDA programming. This card facilitates the acceleration of computations necessary for training the model efficiently.

It's important to note that besides the NVIDIA Graphics Card, no other special hardware is required for the project's development and testing.

## Required Software :

The critical software for building the application includes integrated development environments (IDEs) such as PyCharm (Version 17.0.3), Jupyter Notebook, or Google Collab for running the machine learning models. Additionally, GPU support is required to run the artificial neural network (ANN) machine learning models efficiently. The platform-independent software solution can run on various operating systems, including iOS, Windows, or Linux. No cloud server is necessary to run the application.

**List of Required Software**:

* PyCharm (Version 17.0.3) or Jupyter Notebook: These are the preferred IDEs for running the machine learning models and managing the project codebase.
* GPU Support: This is required to run the ANN machine learning models efficiently. This can include NVIDIA graphics cards with CUDA support.
* Google Colab: Google Colab is also a preferred option for running machine learning models collaboratively, leveraging its cloud-based infrastructure and access to GPU resources.
* Operating System Compatibility: The software solution is compatible with multiple operating systems, including iOS, Windows, and Linux, ensuring flexibility in deployment.

**Notes**:

* PyCharm or Jupyter Notebook is the primary development environment for coding and executing machine learning models.
* GPU support is essential for accelerating the training process of ANN models.
* The software solution is designed to be platform-independent, allowing deployment on various operating systems without needing a cloud server.

## Third Party Content / Libraries :

The project requires several third-party libraries and content to facilitate its completion. Each library serves a specific purpose in data processing, machine learning model development, and visualization. Adhering to the licensing terms for all content and libraries used in the project is essential. Additionally, the appropriate library versions are specified to ensure compatibility and consistency.

**List of Required Libraries**:

* pandas: A powerful data manipulation library for data preprocessing and analysis.
* scikit-learn (sklearn): A comprehensive machine learning library providing various algorithms for classification, regression, clustering, and more.
* matplotlib: A plotting library for creating static, interactive, and animated visualizations in Python.
* numpy: A fundamental package for scientific computing supporting large, multi-dimensional arrays and matrices.
* xgboost: An optimized gradient boosting library for machine learning tasks.
* seaborn: A data visualization library based on matplotlib, providing a high-level interface for drawing attractive statistical graphics.
* torch (PyTorch): An open-source machine learning library for developing deep learning models, including neural networks.
* CatBoost: A gradient-boosting library optimized for categorical features that is often used in tabular data.
* tensorflow: An open-source machine learning framework for developing and training deep learning models, including neural networks.
* 10. time: A standard Python library for handling time-related operations and measurements.

**Licensing Terms**:

* Each library's licensing terms should be reviewed and adhered to as per the project requirements.

**Special Requirements**:

* Ensure that all third-party libraries are appropriately installed and are compatible with the project's Python environment.
* Regularly update the libraries to the latest stable versions to leverage new features and bug fixes.

## Quality:

* Accuracy: The primary quality goal is to enhance the accuracy of machine learning models for network intrusion detection. This will be measured using metrics such as classification accuracy, precision, recall, and F1-score, which quantify the ability of the models to classify instances of intrusions and regular network traffic correctly.
* Robustness: Another vital quality aspect is the robustness of the models, which refers to their ability to maintain high performance across different datasets and in the presence of noisy or adversarial inputs. This will be evaluated by cross-validation on diverse datasets and testing the models against various network attacks.
* Efficiency: Efficiency measures the computational resources the models require for training and inference. Metrics such as training time, memory usage, and inference speed will be used to assess the efficiency of the models.

## Other Special Considerations:

* Security and Privacy: Since the project deals with network data, ensuring its security and privacy is crucial. Complying with relevant data protection regulations, like the General Data Protection Regulation (GDPR) in the European Union or other local laws, is essential. Any team working on the project should be aware of these requirements and how they impact the handling and processing of dataset features.
* Ethical Use of AI: The development and deployment of AI models should follow ethical guidelines to prevent biases and unfair treatment of individuals or groups. This includes the ethical use of data, the potential consequences of model decisions, and the transparency of AI processes.
* Continuous Learning and Updating: Cyber threats are constantly evolving; thus, the models must be capable of updating and learning from new threats. A mechanism for continuous learning, possibly through online learning or periodic retraining with new data, should be considered.
* Model Explainability: The models should be accurate and explainable. Explainability in machine learning refers to describing how a model made a particular decision, which is crucial for trust and transparency, especially when dealing with security.

## Process

## Process Description and justification:

 For our project, we have chosen an adapted Agile development process specifically tailored to our team's and the project's requirements. This process involves using Taiga as our primary project management tool, supplemented by Google Drive for document storage and YouTube for sharing sprint review meetings.

 The adapted Agile process is justified due to its flexibility, iterative nature, and focus on collaboration, which aligns with the dynamic requirements of our machine learning project. We can effectively plan and track our progress throughout each sprint by breaking down our work into manageable user stories and tasks within Taiga. Additionally, Taiga's support for Agile methodologies ensures that our team can easily prioritize tasks, adjust workflows as needed, and respond to project scope or requirements changes.

 Furthermore, integrating Google Drive allows us to organize and share project documentation seamlessly, enhancing collaboration and ensuring accessibility for all team members. This aligns with the sponsor's requirement for regular updates.

 The chosen development process meets the sponsor's requirements by facilitating efficient project management, fostering collaboration, and enabling transparent communication. It suits our team's needs, allowing us to adapt to changing requirements and deliver high-quality results within the project timeline.

## Tools:

For the project focused on improving ML models for network intrusion detection, several non-hardware tools are essential for development and software process management:

* Python Programming Language: Python is the primary programming language for developing machine learning models due to its extensive libraries and frameworks, such as scikit-learn, TensorFlow, PyTorch, and XGBoost.
* Jupyter Notebooks and Google Colabs: Jupyter Notebooks and Google Collabs are invaluable for exploratory data analysis, prototyping models, and documenting the development process. They provide an interactive environment for running code, visualizing data, and writing explanatory text.
* Data Analysis and Visualization Libraries: Pandas, NumPy, sckitlearn, and matplotlib are indispensable for data preprocessing, analysis, and visualization. They facilitate tasks like data cleaning, feature engineering, and generating visualizations to gain insights into the dataset.
* XGBoost, AdaBoost, CatBoost, and GAN Libraries: Specific libraries like XGBoost and GAN frameworks (e.g., TensorFlow or PyTorch) are crucial for implementing and fine-tuning the XGBoost algorithm and generative adversarial networks (GANs) for improving model accuracy and robustness.
* Version Control Systems: Tools like Git and GitHub are essential for version control, collaboration, and managing codebase changes across team members. They enable tracking of project history, branching, merging, and resolving conflicts.
* Integrated Development Environments (IDEs): IDEs such as PyCharm, VS Code, SublimeText, or JupyterLab provide sophisticated development environments with features like code autocompletion, syntax highlighting, debugging, and project management, enhancing developer productivity.
* Taiga: Taiga is chosen for project management due to its intuitive interface and support for Agile methodologies. It facilitates efficient collaboration and task tracking for the team's iterative development process. Its flexibility and integration capabilities further enhance productivity and streamline project workflows.
* Zoom and YouTube: Zoom and YouTube are used for recording meetings due to Zoom's ease of use and comprehensive recording features. YouTube provides a platform for sharing recorded meetings, ensuring accessibility and transparency for stakeholders. One meeting is uploaded per sprint to maintain an organized and accessible archive of project progress.

By effectively leveraging these non-hardware tools, the development team can streamline the development process, improve collaboration, and deliver high-quality ML models for network intrusion detection that meet the project objectives and stakeholder requirements.

## Roles and Responsibilities :

In our project development process, we have defined specific roles and responsibilities for each team member to ensure the smooth execution of tasks. While roles may rotate throughout the project, this transition will be managed effectively to maintain continuity and efficiency. It's crucial to note that every team member, regardless of their assigned role, contributes significantly to developing software deliverables. These roles provide clarity and allow individuals to take ownership of specific aspects of the project, fostering accountability and teamwork.

**Sprint 1:**

Roles:

* Scrum Master: Arnav Raviraj
* Product Advocate: Dr. Abdallah Moubayed
* Developers: Adit Sandeep Virkar, Vinay Kantilal Chavhan, Shivanjay Vilas Wagh

**Sprint 2:**

Roles:

* Scrum Master: Vinay Kantilal Chavhan
* Product Advocate: Dr. Abdallah Moubayed
* Developers: Adit Sandeep Virkar, Arnav Raviraj, Shivanjay Vilas Wagh

**Sprint 3:**

Roles:

* Scrum Master: Shivanjay Vilas Wagh
* Product Advocate: Dr. Abdallah Moubayed
* Developers: Adit Sandeep Virkar, Arnav Raviraj, Vinay Kantilal Chavhan

**Sprint 4:**

Roles:

* Scrum Master: Adit Sandeep Virkar
* Product Advocate: Dr. Abdallah Moubayed
* Developers: Shivanjay Vilas Wagh, Arnav Raviraj, Vinay Kantilal Chavhan

## Location of Project Artifacts :

Regarding project organization, our choice encompasses a combination of platforms to optimize usability, reliability, and change management. GitHub is our primary repository for version control and collaborative coding efforts, offering robust features for code management, issue tracking, and pull requests. To access our project materials, including codebase and documentation, please click <https://github.com/Araviraj8/SER_517_F_33>.

Additionally, we utilize Google Drive to store and share project documentation, meetings, and other non-code assets. Its intuitive interface and real-time collaboration capabilities enhance usability and ensure easy access to essential project resources for all team members. For supplementary project documents and materials, please access our Google Drive folder: <https://drive.google.com/drive/folders/17fXt0y-jq5i_r6ynpcHEslSKF3EHksBE>.

We rely on Taiga to facilitate project planning, task management, and agile workflows. This platform enables us to organize user stories, track tasks, and prioritize features, fostering efficient project execution and seamless change management. For project management-related tasks and updates, please visit our Taiga project board at <https://tree.taiga.io/project/araviraj8-ser517-f-33/backlog>.

Overall, this multifaceted approach to project organization ensures that our team can effectively collaborate, maintain reliability in our deliverables, and adapt to changes throughout the project lifecycle.

## Sponsor Communications :

Our sponsor meetings are scheduled bi-weekly on Tuesdays at the Polytechnic campus. Due to the spring break, one of our meetings has been rescheduled to the following Tuesday. Below are the dates and details of the sponsor meetings:

**Sponsor Meeting 1:**

Date: Jan 23, Tuesday, 1:30 pm

Our sponsor gave us a research paper entitled '5G-NIDD: A Comprehensive Network Intrusion Detection Dataset Generated over 5G Wireless Network'. They introduced the problem statement, described the dataset's generation process, and highlighted its importance. Our task was to investigate new machine-learning models. In the following meeting, we were assigned to conduct a literature review, familiarize ourselves with the project and the paper, and so forth. The overall objective was to enhance the models examined in the paper and conduct an analysis.

**Sponsor Meeting 2:**

Date: Feb 6, Tuesday, 1:30 pm

We handed in our literature review and received feedback. During our discussion, we focused on the XGBoost model, known for its superior performance compared to Random Forest (RF) and Artificial Neural Networks (ANN) examined in the original paper. Our suggestion involves leveraging XGBoost to enhance the model's performance across various metrics such as accuracy, inference time, and training time. Additionally, we delved into topics like adversarial machine learning and federated learning (FL), the latter being a distributed approach for deploying intrusion detection models. Furthermore, we addressed concerns regarding noise attacks and poisoning attacks during the conversation.

**Sponsor Meeting 3:**

Date: Feb 20, Tuesday, 1:30 pm

During the meeting, we engaged in a comprehensive discussion with the Sponsor regarding the application of our model. We presented our analysis, focusing on the utilization of boosting models. The Sponsor suggested exploring the potential of GAN-based and ANN-based algorithms, prompting us to consider various classification techniques for the test set and to experiment with different data processing methods.

Furthermore, we discussed the possibility of applying alternative data processing techniques to the raw data, including various feature selection strategies, to enhance performance. The Sponsor guided us in prioritizing fixing one model and implementing ANN and GAN-based models before our next meeting, scheduled after spring break.

**Sponsor Meeting 4:**

Date: March 12, Tuesday, 1:30 pm

We analyzed various boosting, ANN, and GAN-based models for binary classification. The sponsor has recommended expanding the analysis from binary to multi-class classification, broadening the scope of the study to handle more than two classes in classification tasks. Additionally, they suggest employing input sampling techniques to introduce variations in the training data. Hyperparameter tuning is also emphasized to fine-tune the model's parameters for optimal performance, with techniques such as grid search or random search being helpful in this context. Finally, the sponsor suggests exploring various evaluation metrics to assess model performance accurately in multi-class classification scenarios, including but not limited to accuracy, precision, recall, and F1-score.

# 

# 