Network Traffic Analysis Using Machine Learning

BITS ZC4999T: Capstone Project

by

Rushil Gupta

202117bh082

Capstone Project work carried out at

HCL Tech, Lucknow

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE
PILANI (RAJASTHAN)

July,2025

Mid Semester Progress Report

Details of work done till date: -

The project's primary objective is to leverage machine learning to detect and classify malicious network traffic. Over the past ten weeks, the focus was on establishing a robust data pipeline, performing in-depth analysis, and building a strong foundation for model development.

Phase 1: Foundation and Data Acquisition (Weeks 1-4)
The initial phase of the project was dedicated to literature review, environment setup, and a deep dive into the dataset. A comprehensive review of existing ML-based IDS approaches helped inform the project's direction, highlighting the importance of robust preprocessing and comparative model evaluation.

The development environment was configured using Python, with essential libraries such as Pandas for data manipulation, Matplotlib and Seaborn for visualization, and Scikit-learn for machine learning. The primary dataset, networktraffic.csv, was acquired and loaded from Kaggle, a popular data science platform. Initial inspection revealed a rich but complex dataset containing over 20 features describing network connections, including protocol types, connection states, and byte counts.

A significant challenge identified during this phase was data quality. The raw data contained numerous non-standard placeholders (e.g., '-') and inconsistent data types. A substantial effort was made to develop a resilient preprocessing script. This involved:

- Replacing all placeholder values with NaN (Not a Number) to enable standardized handling.
- Identifying and separating columns into categorical and numerical types.
- Implementing an imputation strategy to handle the newly created missing values. For categorical features, the most frequent value (mode) was used, while for numerical features, the median was chosen to mitigate the effect of outliers. This imputation step was critical to prevent data loss and ensure the integrity of the dataset for subsequent analysis.

Phase 2: Exploratory Data Analysis and Feature Engineering (Weeks 5-7)
With a clean dataset, the project moved into a rigorous Exploratory Data Analysis (EDA) phase to uncover patterns and relationships. The first step was to visualize the distribution of the target variable, 'label'. The analysis showed an imbalanced dataset, with 'Benign' traffic instances outnumbering 'Malicious' ones. This was a crucial finding that would influence model evaluation metrics in later stages, as accuracy alone could be misleading.

A key deliverable from this phase was the creation of a correlation heatmap. This visualization mapped the statistical relationship between each feature and the target variable. It provided valuable insights, revealing that features like duration, orig_bytes, and certain connection states (conn_state) had a stronger correlation with malicious activity than others. This analysis formed the empirical basis for the feature selection process.

The final part of this phase involved feature engineering, where the cleaned, imputed data was transformed into a format suitable for machine learning. This was achieved through one-hot encoding, which converted categorical string values (like proto='tcp') into a numerical format that the models can process.

Phase 3: Feature Selection and Baseline Modeling (Weeks 8-10)
Building on the EDA, a systematic feature selection process was implemented. While the heatmap provided initial clues, a more sophisticated approach was used to quantify feature importance. A Random Forest model was trained on the entire feature set specifically to leverage its feature_importances_ attribute. This technique provided a ranked list of features based on their contribution to the model's predictive power. A threshold was set to select only the most impactful features, effectively reducing the dimensionality of the dataset from over 40 (post-encoding) to a more manageable and potent subset of 15-20 features. This step is vital for improving model performance and reducing computational overhead.

With a refined set of features, the final weeks of this reporting period were focused on training and comparing several baseline machine learning models. The goal was to establish a performance benchmark and understand which algorithmic approaches are best suited to this classification problem. Several models were implemented and evaluated including:

- Logistic Regression: A simple, interpretable linear model.
- Decision Tree: A non-linear model that provides clear decision rules.
- Random Forest: An ensemble model known for its high accuracy and robustness.
- LightGBM: A powerful gradient-boosting framework optimized for speed and performance.

Each model was trained on the selected features and evaluated based on its accuracy. This comparative analysis revealed that ensemble methods like Random Forest and LightGBM significantly outperformed the simpler models, achieving preliminary accuracies well above 90%. This confirms that the engineered features are highly predictive and sets a strong foundation for the next phase of the project: model optimization and deployment.

Plan of work yet to be done: -

Having successfully established a data pipeline, selected key features, and benchmarked several baseline models, the focus will now shift on testing newer models, comparing their metrics and choosing a final model, refining the chosen model, integrating it into a practical system, and preparing the final project deliverables.

Phase 4: Model Optimization and Final Evaluation (Weeks 11-12)

Once the comparative analysis of several models (both baseline and ensemble) is complete and the best-performing model architecture is identified, the next critical step would be to fine-tune its performance. This phase would be dedicated to hyperparameter tuning, a process of systematically adjusting the model's internal settings to achieve optimal results.

Instead of using the default parameters, techniques like Grid Search or Randomized Search will be employed. These methods will automatically iterate through various combinations of parameters to find the configuration that yields the highest performance.

The evaluation will also be expanded beyond simple accuracy. Given the class imbalance discovered during EDA, the final model will be rigorously assessed using a comprehensive set of metrics, including Precision, Recall, and the F1-Score. This ensures the model is not only accurate but also effective at correctly identifying malicious traffic (high recall) without raising too many false alarms (high precision).

Key Deliverables for this Phase:

- An optimized, fine-tuned version of the selected machine learning model.
- A detailed performance report comparing the baseline model with the tuned model, highlighting improvements in Precision, Recall, and F1-Score.
- Visualization of the final model's confusion matrix.

Phase 5: System Integration and Prototype Alerting (Weeks 13-14)

This phase will focus on translating the trained model into a practical, proof-of-concept application. The primary goal would be to build a prototype alerting system that simulates how the IDS would function in a real-world scenario.

A Python script will be developed to serve as the core of this system. This script will be designed to:

- Load the final, optimized machine learning model.
- Ingest new, unseen network traffic data (simulated from the test set).
- Apply the same preprocessing and feature engineering steps used during training to the new data.
- Feed the processed data into the model to generate a prediction ('Benign' or 'Malicious').
- Implement a conditional logic that, upon detecting a 'Malicious' prediction, generates a clear and actionable security alert to the console.

This integration will demonstrate the end-to-end functionality of the project, from raw data processing to actionable security intelligence.

Key Deliverables for this Phase:

- A fully documented Python script capable of real-time prediction and alerting on new data samples.
- A demonstration of the system successfully identifying malicious traffic from the test set and triggering the appropriate alerts.

Phase 6: Testing, Final Documentation and Project Submission (Weeks 15-16)
The final two weeks would be dedicated to last-minute checks of the entire project functioning, and the consolidation and presentation of the entire project. This involves thorough edge case testing and compiling all research, code, results, and analyses into a comprehensive final package.

The main activity will be writing the final project report. This document will be a formal record of the project, detailing the problem statement, the methodology employed, the challenges encountered during data processing, the results of the model evaluation and optimization, and a conclusion summarizing the project's outcomes. A critical section on future work will also be included, suggesting potential improvements such as deploying the model in a live environment or exploring more advanced deep learning architectures.

Alongside the report, all source code will be cleaned, commented, and organized for submission. Finally, a presentation will be prepared for the final review, summarizing the project's key achievements and demonstrating the prototype system.

Key Deliverables for this Phase:

- Thorough testing of edge cases present in the code.
- A comprehensive final project report.
- The complete, well-documented source code for the project.
- Presentation slides for the final project and mentor review.

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
WORK-INTEGRATED LEARNING PROGRAMMES DIVISION

BITS ZC499T: Capstone Project Mid-Semester Progress Evaluation Sheet

ID No. : **202117bh082**

NAME OF THE STUDENT : **Rushil Gupta**

EMAIL ADDRESS : **202117bh082@wilp.bits-pilani.ac.in**

NAME OF THE MENTOR : **Mohammed Mutheeb Parveez**

Capstone PROJECT TITLE : **Network Traffic Analysis Using Machine Learning**

Details of work done till date: Attached with this report
(with reference to Outline)

Plan of work yet to be done: Attached with this report

EVALUATION

CAPSTONE PROJECT PROGRESS EVALUATION *(Please put a tick (✓) mark in the appropriate box)*

EC No.	Component	Excellent	Good	Fair	Poor
1.	Capstone Project Outline	✓			
2.	Work Progress & Achievements	✓			
3.	Initiative and Originality	✓			
4.	Documentation & Expression	✓			
5.	Research & Innovation		✓		
6.	Relevance to the work environment		✓		

	Mentor	**Additional Examiner**
Name	Mohammed Mutheeb Parveez	Kalash Dutt
Qualification	MBA	BCA
Designation	Group Manager	Senior Software Engineer
Employing Orgn and Location	HCL Tech, Bangalore	HCL Tech, Noida
Phone No. (with STD Code)	+919902395353	+918565968856
Email Address	mohd.parveez@hcltech.com	kalash.dutt@hcltech.com
Signature	Mohd Parveez	Kalash Dutt
Date	21-Aug-2025	21-Aug-2025