# CAPSTONE PROJECT

# NETWORK INTRUSION DETECTION

**Presented By:**
1. Shivam Kumar Mishra – [ Dr BC Roy Engineering College ]- Electronics and Communication Engineering

edunet foundation

# OUTLINE

- Problem Statement
- Proposed System/Solution
- System Development Approach
- Algorithm & Deployment
- Result (Output Image)
- Conclusion
- Future Scope
- References

# PROBLEM STATEMENT

- The Challenge is to Create a robust Network Intrusion Detection System (NIDS) using machine learning. The system should be capable of analyzing network traffic data to identify and classify various types of cyber-attacks (e.g., DoS, Probe, R2L, U2R) and distinguish them from normal network activity. The goal is to build a model that can effectively secure communication networks by providing an early warning of malicious activities.

# PROPOSED SOLUTION

The proposed system leverages machine learning algorithms to detect and classify network intrusions based on traffic patterns.

Key Components:

**Data Collection**

Network traffic data is sourced from publicly available datasets such as KDD Cup 99 or NSL-KDD, containing labeled examples of normal and malicious connections.

**Pre-processing**

Data is cleaned, transformed, and encoded to prepare it for training. Feature selection and normalization ensure improved model accuracy.

**Model Development**

Various classification models like Random Forest, SVM, and Neural Networks are tested. The best-performing model is chosen based on precision, recall, and F1-score.

**Deployment**

The trained model is deployed via a REST API interface for real-time network traffic analysis.

**Detection**

The model detects and classifies suspicious traffic, sending alerts when a potential intrusion is identified.

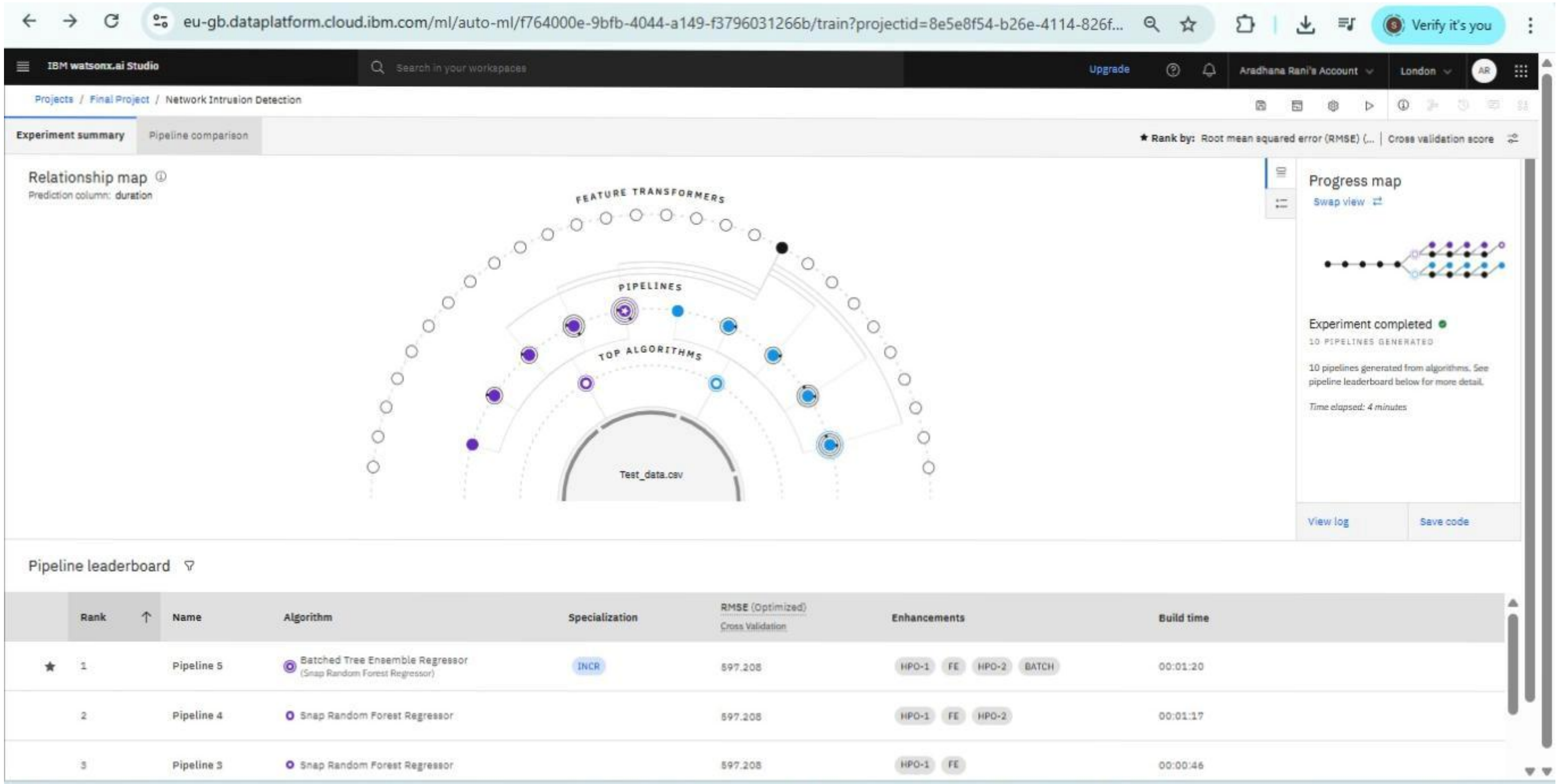
# SYSTEM APPROACH

- IBM Watson Studio was used to develop the machine learning workflow.
- The project uses cloud-based tools and services (e.g., IBM Cloud, Google Colab, or AWS) for scalability and ease of access.
- The dataset is loaded and processed using Python libraries such as Pandas, NumPy, and Scikit-learn.
- Model training, evaluation, and visualization are handled with tools like Matplotlib, Seaborn, and TensorFlow/Keras for deep learning approaches.
- Version control and collaboration were maintained using GitHub and IBM Cloud Object Storage.

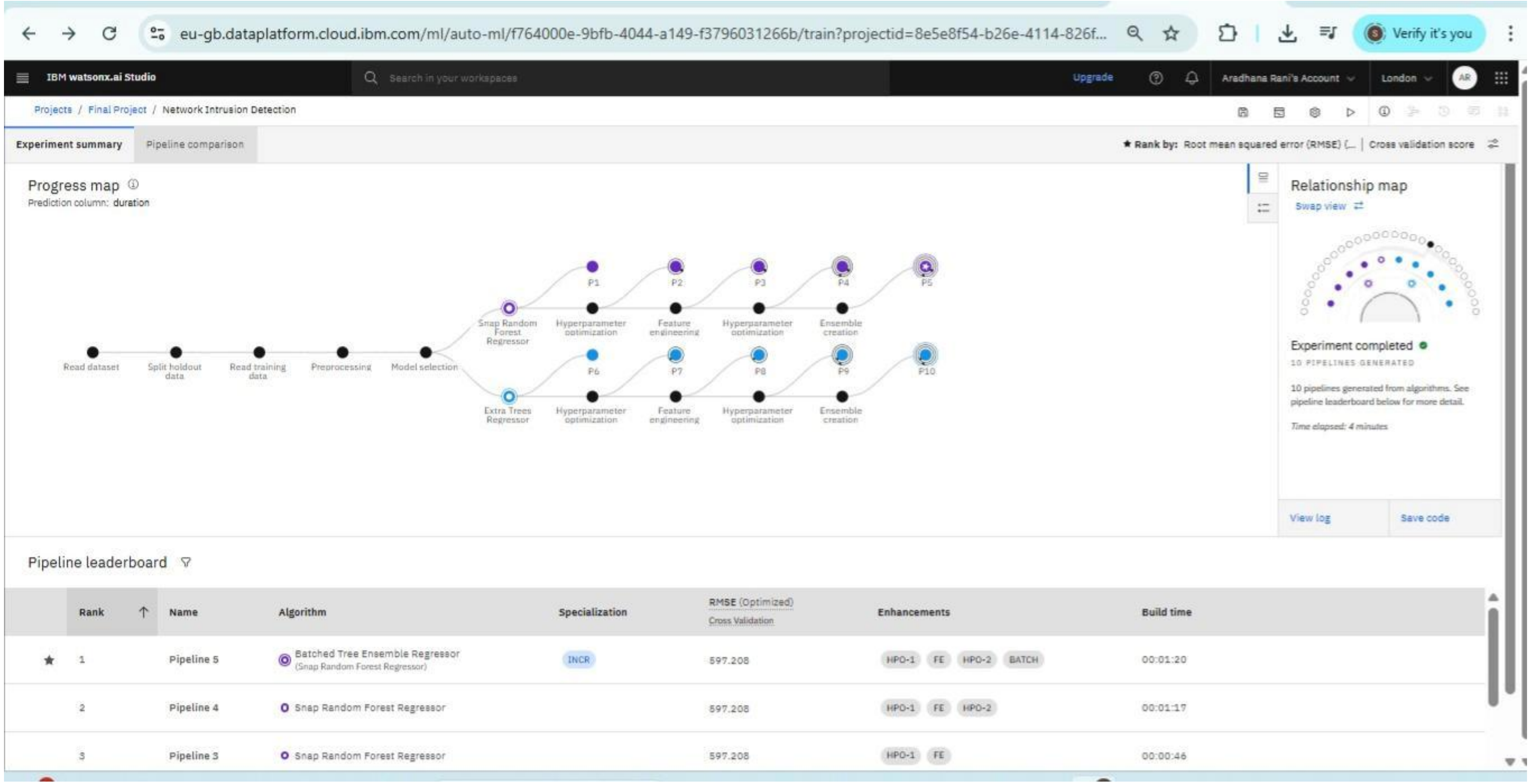
# ALGORITHM & DEPLOYMENT

- Models such as Random Forest, Decision Trees, and Deep Neural Networks were evaluated.
- Feature selection techniques and cross-validation were applied to enhance performance.
- The best model—Random Forest—achieved high classification accuracy and was selected for deployment.
- Deployment involved hosting the model as a REST API using IBM Watsonx.ai Studio or Flask for real-time packet classification.
- Input features include attributes like duration, protocol type, service, flag, and byte count, commonly found in network flow records.
- The system classifies traffic into normal or specific attack types (e.g., DoS, Probe, etc.) for prompt mitigation.

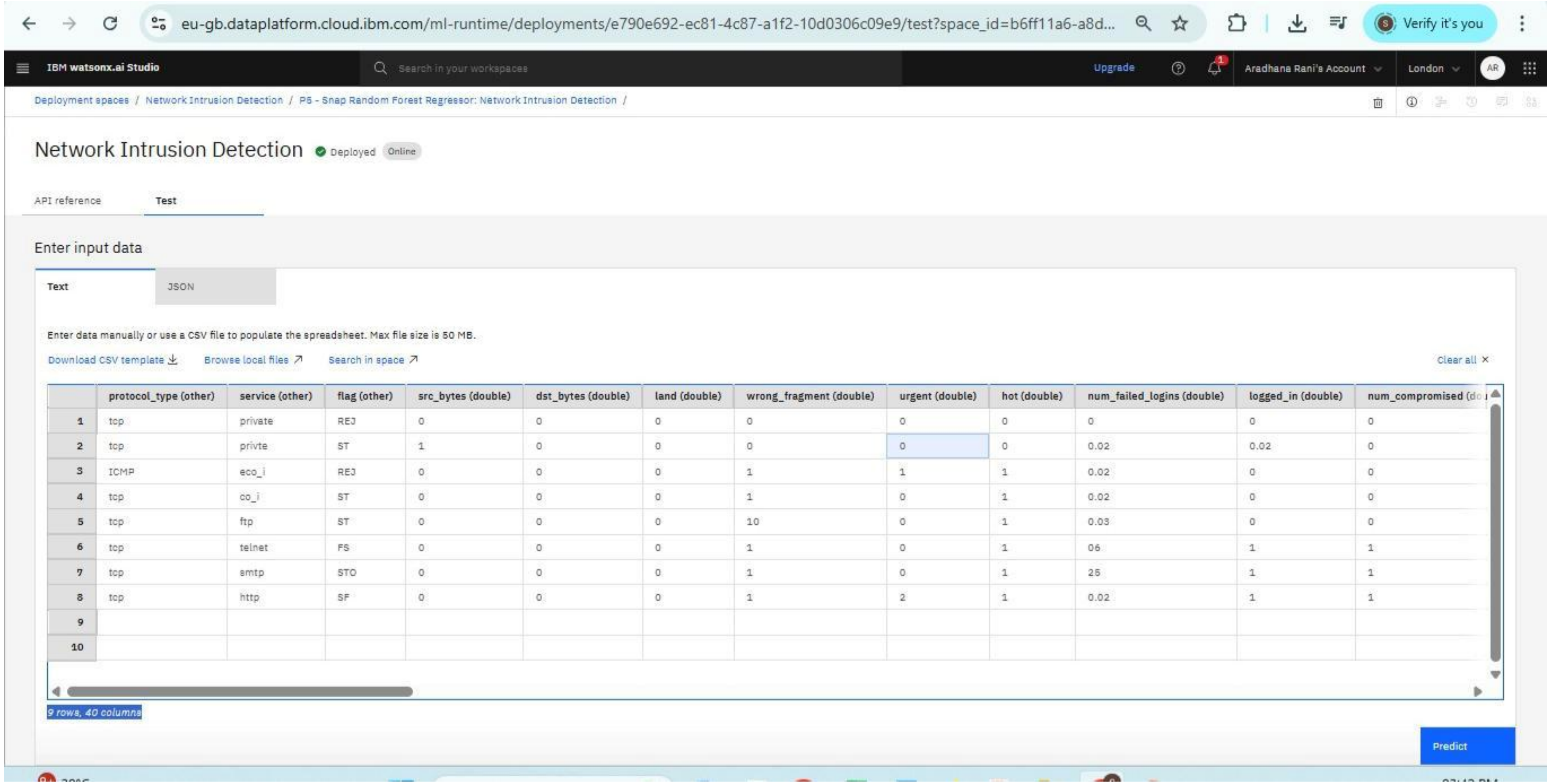
# RESULT




edunet
foundation

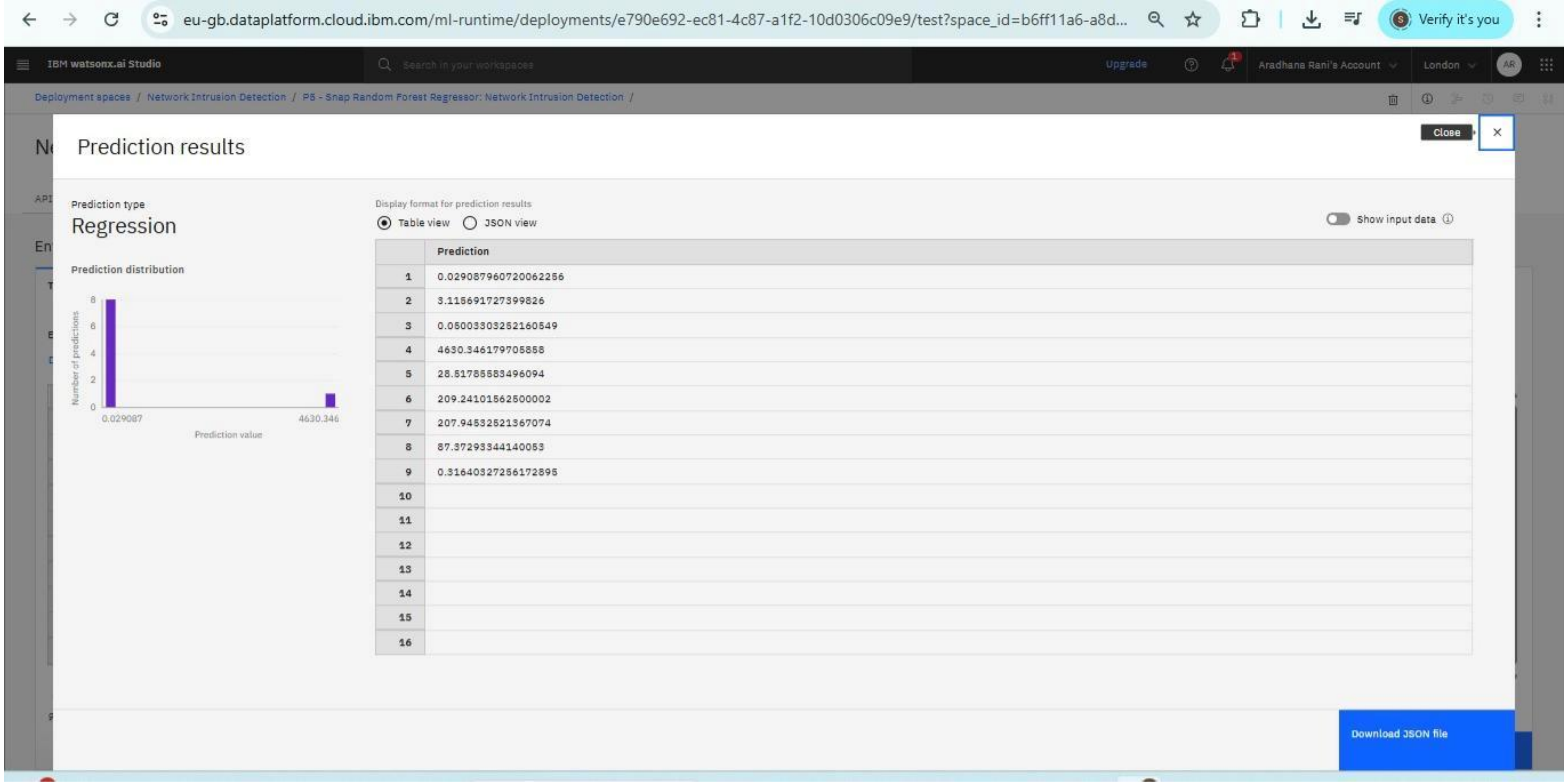



edunet foundation




| | protocol_type (other) | service (other) | flag (other) | src_bytes (double) | dst_bytes (double) | land (double) | wrong_fragment (double) | urgent (double) | hot (double) | num_failed_logins (double) | logged_in (double) | num_compromised (do |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | tcp | private | REJ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | tcp | privte | ST | 1 | 0 | 0 | 0 | 0 | 0 | 0.02 | 0.02 | 0 |
| 3 | ICMP | eco_i | REJ | 0 | 0 | 0 | 1 | 1 | 1 | 0.02 | 0 | 0 |
| 4 | tcp | co_i | ST | 0 | 0 | 0 | 1 | 0 | 1 | 0.02 | 0 | 0 |
| 5 | tcp | ftp | ST | 0 | 0 | 0 | 10 | 0 | 1 | 0.03 | 0 | 0 |
| 6 | tcp | telnet | FS | 0 | 0 | 0 | 1 | 0 | 1 | 06 | 1 | 1 |
| 7 | tcp | smtp | STO | 0 | 0 | 0 | 1 | 0 | 1 | 25 | 1 | 1 |
| 8 | tcp | http | SF | 0 | 0 | 0 | 1 | 2 | 1 | 0.02 | 1 | 1 |
| 9 | | | | | | | | | | | | |
| 10 | | | | | | | | | | | | |

edunet foundation




edunet foundation

# CONCLUSION

- A machine learning-based Network Intrusion Detection System (NIDS) was successfully developed.
- The system analyzes network traffic data to detect and classify various cyber-attacks such as DoS, Probe, R2L, and U2R.
- Supervised learning models like Random Forest provided high accuracy and reliability in intrusion detection.
- Preprocessing and feature engineering significantly improved model performance.
- The final model was deployed via a REST API for real-time traffic analysis and alert generation.
- This approach helps enhance network security by enabling early detection of malicious activities.
- The system can be integrated into existing cybersecurity frameworks to reduce the risk of attacks.

# FUTURE SCOPE

- **Integration with Real-Time Network Monitoring Tools:**
- Extend the system to work seamlessly with real-time packet sniffers like Wireshark or Zeek for live traffic detection.
- **Adoption of Deep Learning Models:**
- Implement LSTM or CNN-based models for improved detection of complex patterns and zero-day attacks.
- **Continuous Learning:**
- Enable online learning techniques to update the model as new types of attacks emerge in real-world networks.
- **Deployment on Edge Devices:**
- Deploy lightweight models on routers or IoT gateways for decentralized, faster threat detection.
- **Enhanced Visualization Dashboards:**
- Build dashboards for monitoring intrusion attempts, model performance, and threat history for better incident response.

# REFERENCES

- **Kaggle dataset link** – https://www.kaggle.com/datasets/sampadab17/network-intrusion-detection.
- IBM Cloud Documents.
- IBM Watson Studio Tutorials.

edunet foundation

# IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence

IBM SkillsBuild

IBM

## Shivam Kumar Mishra

Has successfully satisfied the requirements for:

### Getting Started with Artificial Intelligence

Issued on: Jul 15, 2025

Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/123a5ab7-f8e6-4e15-b753-58c2ea42e059

IBM

# IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Journey to Cloud: Envisioning Your Solution

IBM SkillsBuild

## Shivam Kumar Mishra

Has successfully satisfied the requirements for:

### Journey to Cloud: Envisioning Your Solution

Issued on: Jul 16, 2025

Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/8b967192-e16d-4f87-b766-9837ac4ab67c

IBM

# IBM CERTIFICATIONS

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Shivam Mishra

for the completion of

## Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

**Completion date:** 16 Jul 2025 (GMT)

**Learning hours:** 20 mins

# THANK YOU