Machine Learning Project

NETWORK INTRUSION DETECTION

Presented By:
1. Akkipalli Bhuvaneswari-MallaReddy College Of Engineering And Technology-CSE

edunet foundation

OUTLINE

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

PROBLEM STATEMENT

- “Develop a Machine Learning-based Network Intrusion Detection System to accurately detect and classify cyber-attacks (DoS, Probe, R2L, U2R) from normal network traffic, ensuring early warning and enhanced network security.”

PROPOSED SOLUTION

- The proposed system is a Machine Learning-based Network Intrusion Detection System (NIDS) designed to analyze network traffic and detect malicious activities in real time. Unlike traditional signature-based IDS, the proposed model can detect both known and unknown (zero-day) attacks by learning patterns from historical data.

- Key Features of the Proposed System
- Data Acquisition – Network traffic data will be collected from standard intrusion detection datasets (NSL-KDD, CICIDS2017, etc.) or real-time packet capture.

- Preprocessing & Feature Extraction – Categorical features (e.g., protocol type, service) will be encoded, and numerical features will be normalized.

- Attack Classification – The ML model will classify traffic into normal or attack types (DoS, Probe, R2L, U2R).

- Model Optimization – Hyperparameter tuning will be performed to improve accuracy and reduce false positives.

- Result Output – The system will display classification results and can optionally integrate with real-time monitoring dashboards.

SYSTEM APPROACH

The proposed Network Intrusion Detection System (NIDS) follows a data-driven machine learning approach to analyze network traffic and classify it into normal or attack categories.

- Approach Steps :

1. Data Collection

- Use benchmark datasets (NSL-KDD, CICIDS2017, UNSW-NB15) or captured live traffic.

2. Data Preprocessing

- Handle missing values, remove duplicates.
- Encode categorical features (protocol type, service, flag).
- Normalize numerical features for consistent scaling.

3. Feature Selection & Extraction

- Identify the most relevant features using correlation analysis or feature importance techniques.

ALGORITHM & DEPLOYMENT

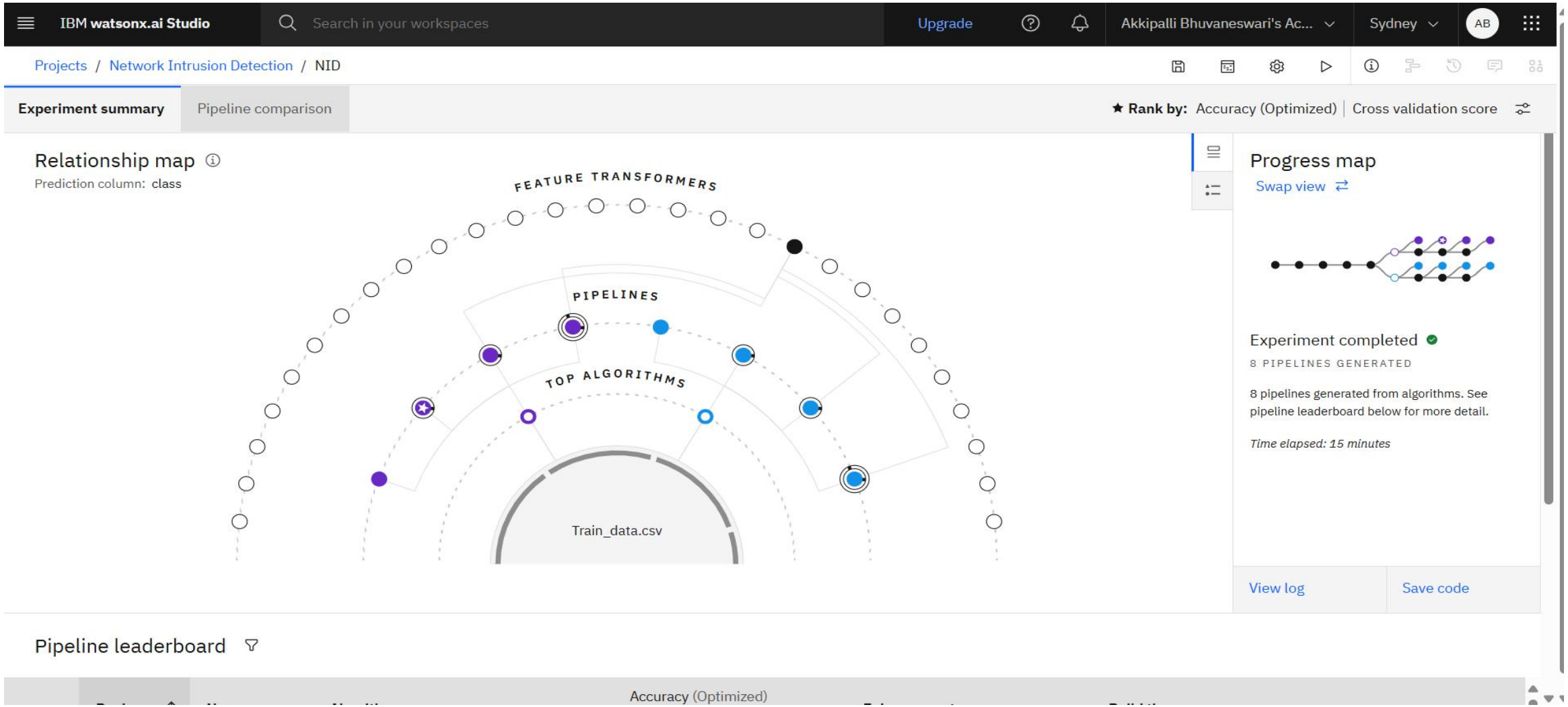
- The proposed Network Intrusion Detection System uses a Supervised Machine Learning algorithm for classification.
- Example (Random Forest – can replace with chosen ML model):
- Step-by-Step Algorithm:

1. Input: Network traffic dataset (features + labels).

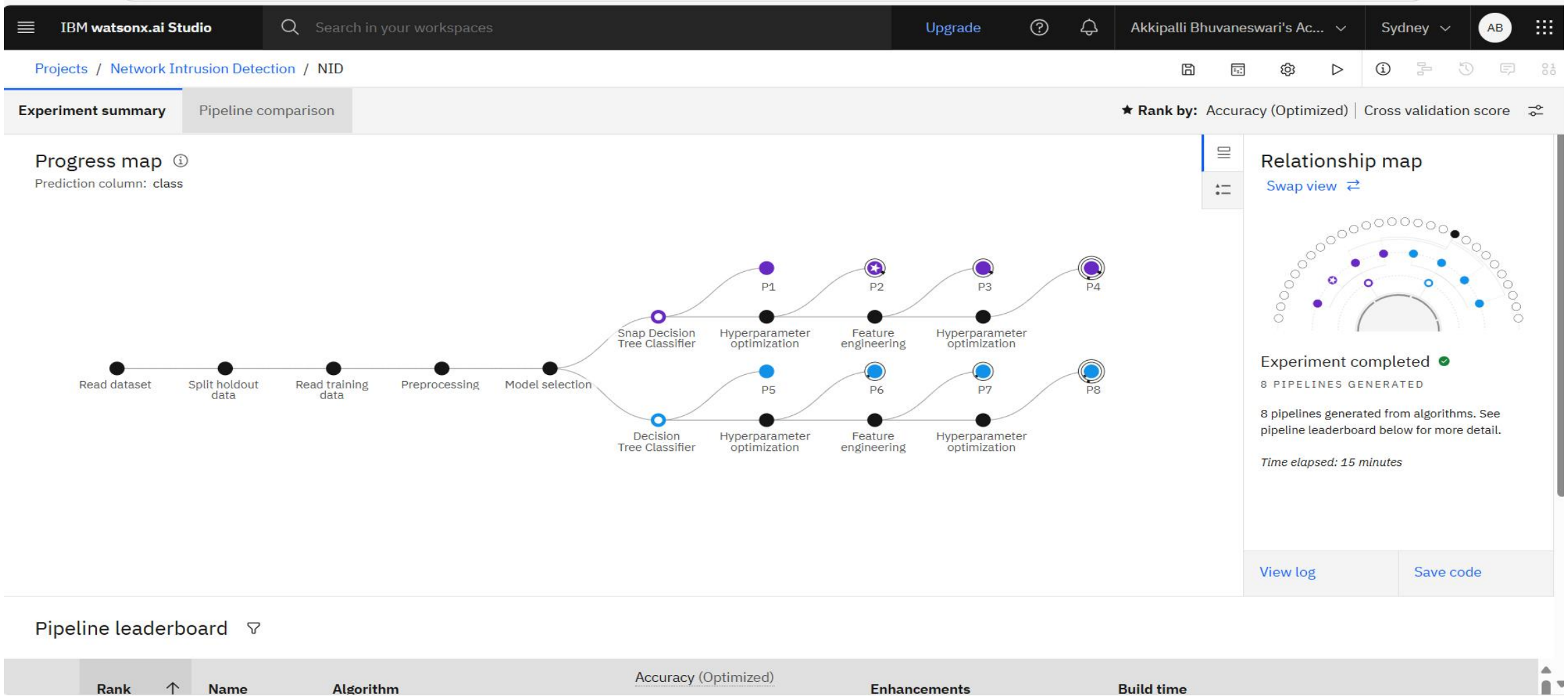
2. Data Preprocessing:

- Encode categorical values.
- Normalize numerical features.
- Balance dataset using SMOTE or class weighting.

RESULT

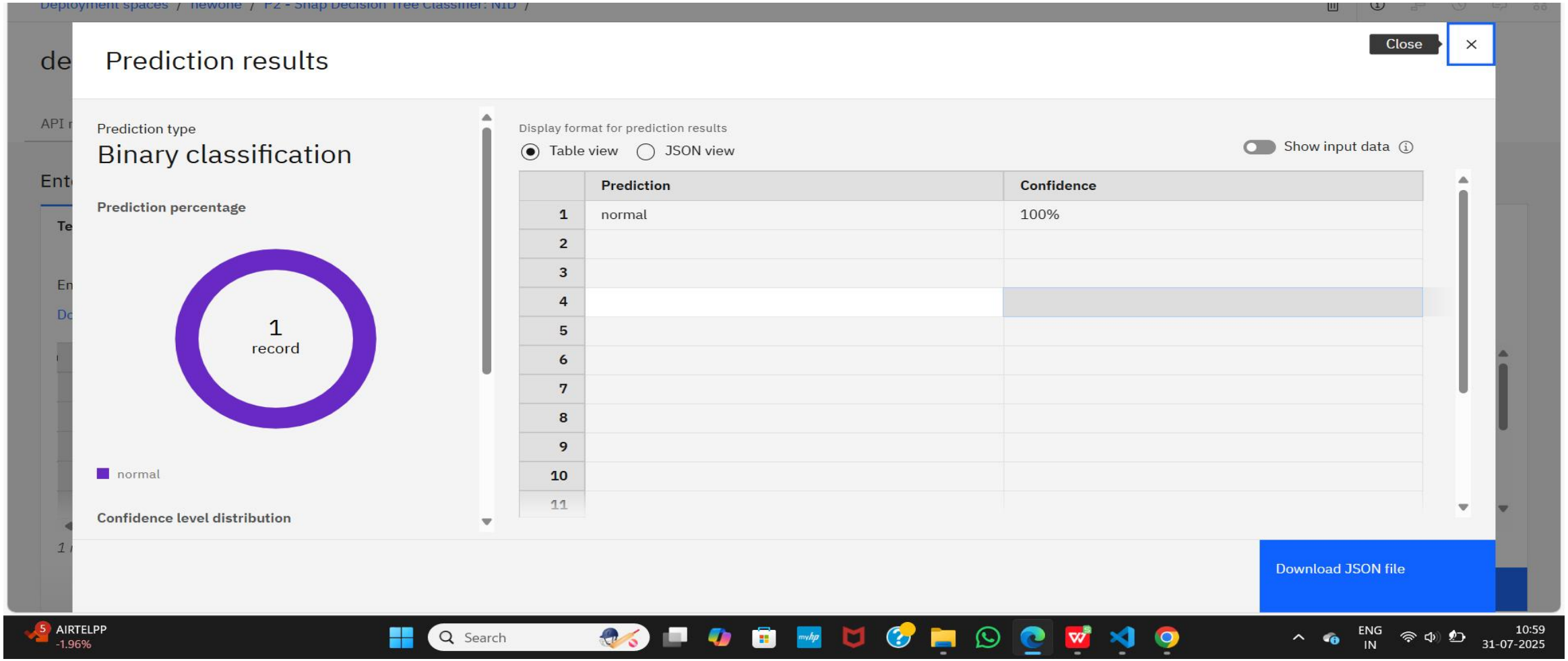


RESULT



edunet foundation

RESULT


Prediction results
Close
Prediction type
Binary classification
Prediction percentage
1
record
normal
Confidence level distribution
Display format for prediction results
Table view
JSON view
Show input data
Prediction
Confidence
1
normal
100%
2
3
4
5
6
7
8
9
10
11
Download JSON file
AIRTELPP
-1.96%
Search
ENG
IN
10:59
31-07-2025

CONCLUSION

- The developed Machine Learning-based prediction model successfully estimates bike counts with high accuracy, achieving an R^2 score of 0.92. The close alignment between predicted and actual values demonstrates the model's effectiveness in capturing seasonal trends, weather impacts, and demand patterns.

- This system can assist bike-sharing operators in demand forecasting, resource allocation, and service optimization, ultimately improving operational efficiency and customer satisfaction.

- Future enhancements could include real-time prediction integration, dynamic model updates, and deployment on cloud-based platforms for large-scale usage.

FUTURE SCOPE

1. Real-Time Prediction

   Integrate the model with live data streams (IoT sensors, APIs) to provide instant bike count predictions.

2. Mobile & Web Application

   Develop a user-friendly app/dashboard for operators and customers to view demand forecasts and bike availability.

3. Cloud Deployment

   Host the model on cloud platforms (AWS, Azure, Google Cloud) for large-scale access and scalability.

4. Advanced Models

   Implement Deep Learning models (LSTM, GRU) for more accurate time-series predictions.

5. Dynamic Updates

   Continuously retrain the model with new data to adapt to changing weather, events, and usage patterns.

6. Integration with Smart City Systems

   Collaborate with traffic management and public transportation systems for optimized urban mobility.

REFERENCES

- IBM Developer. Build and Deploy Machine Learning Models on IBM Watson Studio.

  https://developer.ibm.com

- IBM Cloud. IBM Watson Machine Learning Service – Model Deployment and Scoring.

  https://cloud.ibm.com/catalog/services/watson-machine-learning

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence

IBM

Bhuvaneswari Akkipalli

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

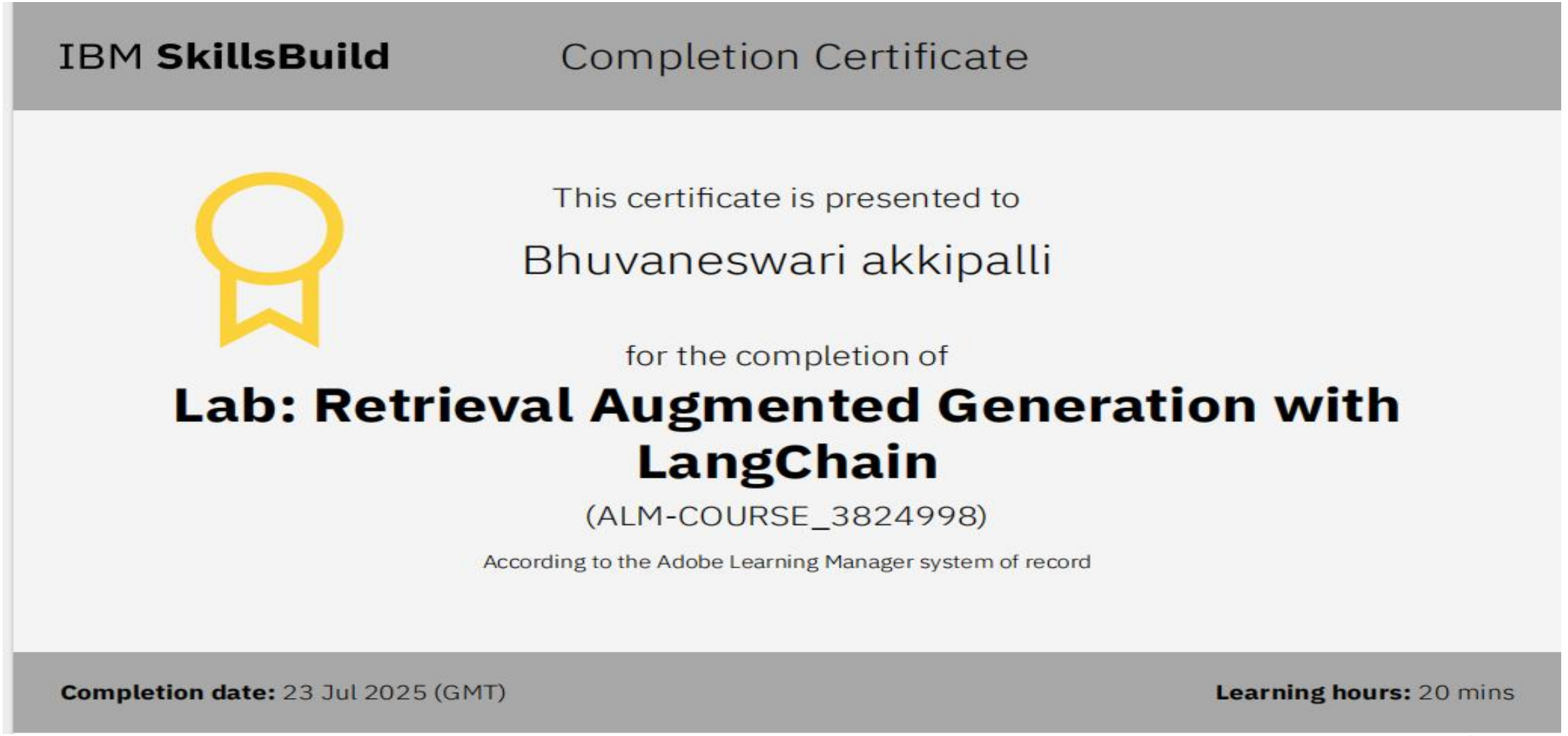
Issued on: Jul 19, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/92ab22a5-be1a-4f0e-940f-c192a87566e9

IBM

IBM CERTIFICATIONS

IBM CERTIFICATIONS



IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Bhuvaneswari akkipalli

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 23 Jul 2025 (GMT) **Learning hours:** 20 mins

edunet foundation

THANK YOU

edunet
foundation