IBM SKILLS BUILD & AICTE PROJECT

NETWORK INTRUSION DETECTION SYSTEM (NIDS) USING MACHINE LEARNING

Presented By:

1. Student Name- SHACHIN P R
2. College Name- GOVERNMENT COLLEGE OF ENGINEERING,ERODE
3. Department- CSE (I-YEAR)

edunet foundation

OUTLINE

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

PROBLEM STATEMENT

- Network Intrusion Detection The Challenge: 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

To address this challenge, we propose a machine learning-based NIDS built using IBM AutoAI. The system automatically selects the best model pipeline to classify network traffic and detect intrusions.

Key components include:

- **Data Collection:**
  Used a labeled dataset with 22,544 records containing network traffic features and attack labels from Kaggle's Network Intrusion Detection Dataset

- **Data Preprocessing:**
  AutoAI handled feature transformation, encoding, and missing value treatment automatically.

- **Model Training:**
  AutoAI generated 8 pipelines and selected the best-performing one (Snap Decision Tree Classifier) with 99.5% accuracy.

- **Prediction Output:**
  The model classifies traffic as either anomaly or normal with high confidence (90–100%).

SYSTEM APPROACH

- **Platform Used:**
  IBM watsonx.ai Studio (AutoAI + Deployment)

- **Steps Followed:**
  Uploaded dataset and selected target column (class)
  AutoAI generated multiple pipelines and evaluated them
  Logs confirmed smooth execution in under 3 minutes
  Model deployed directly via IBM watsonx.ai Studio—no manual coding required

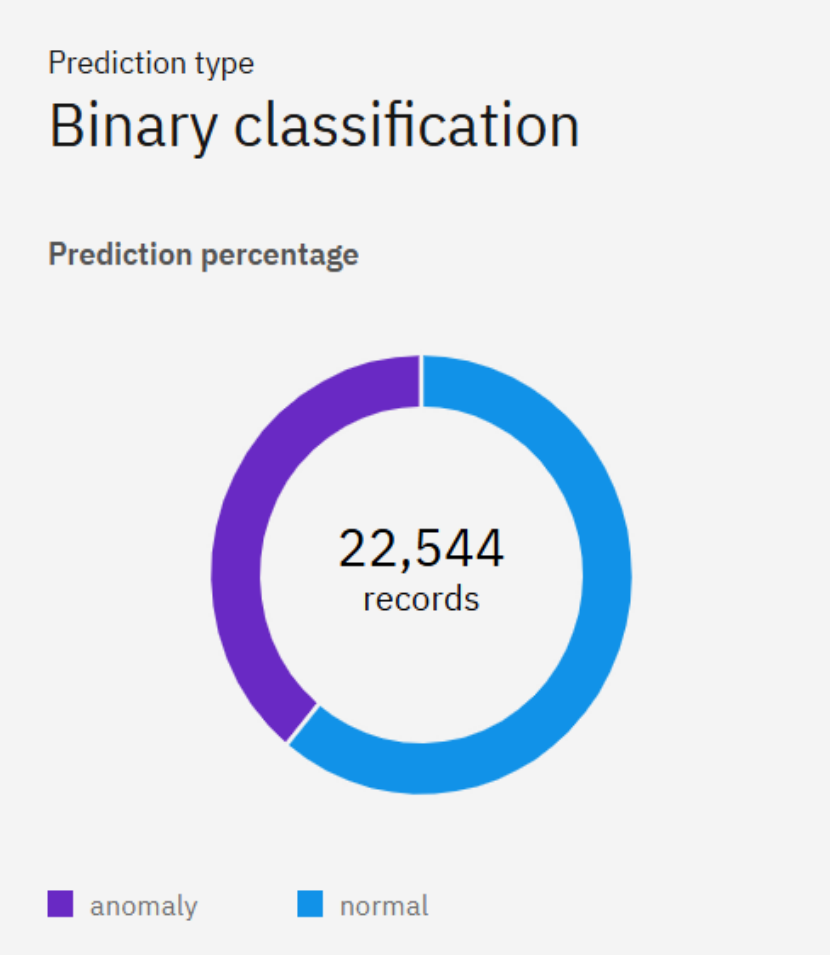
- **System Requirements:**
  IBM Cloud Lite account
  AutoAI tool access
  IBM watsonx.ai Runtime (Python 3.x) for model execution and deployment
  IBM watsonx.ai Deployment interface

- **Libraries Required (for integration only):**
  IBM Cloud SDKs (optional for API integration)
  IAM token setup for secure access

ALGORITHM & DEPLOYMENT

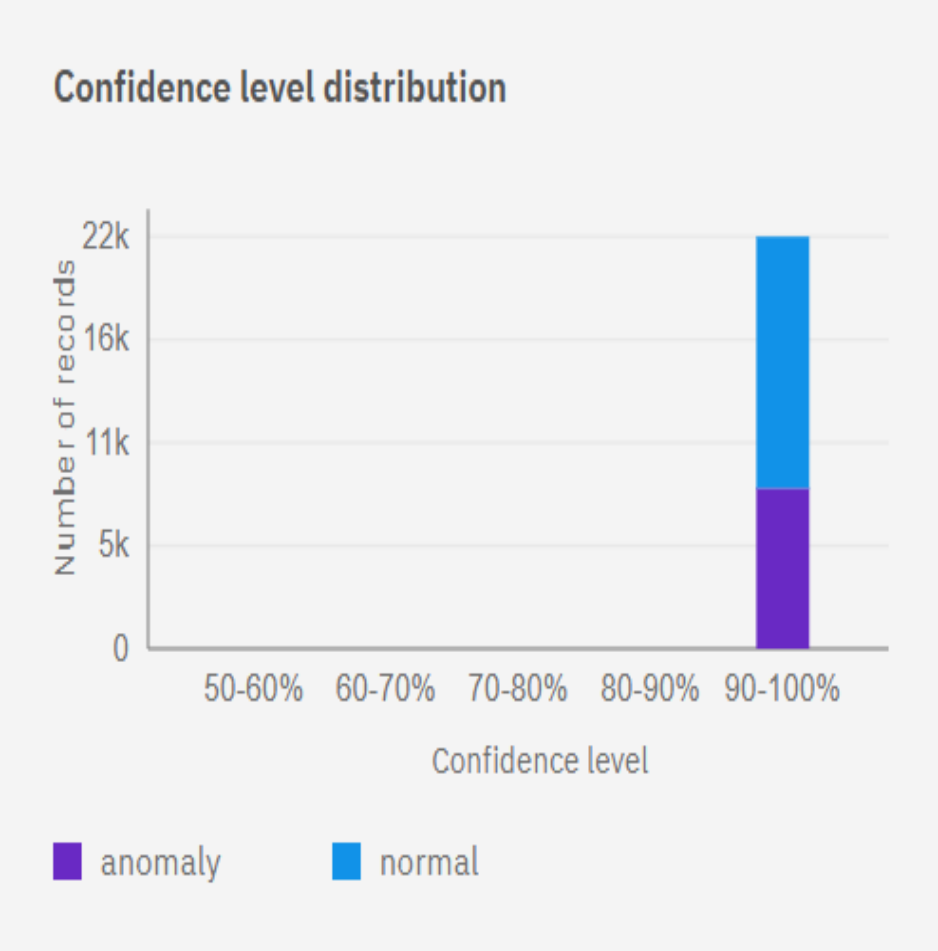
- **Algorithm Selected:**
  Snap Decision Tree Classifier (AutoAI-selected)
- **Why This Algorithm:**
  High accuracy (99.5%)
  Fast training time (9 seconds)
  Interpretable and efficient for real-time detection
- **Training & Testing:**
  Kaggle's Network Intrusion Detection Dataset
  Train_data.csv (125,973 rows) & Test_data.csv (22,544 rows)
  Target column: class (normal vs attack)
- **Training Process:**
  AutoAI used 10% of training data for initial selection, then optimized top pipelines using hyperparameter tuning.
- **Deployment Strategy:**
  Model deployed using IBM watsonx.ai Studio's built-in deployment feature
  Generated public and private scoring endpoints
  No Flask or Python required—ready-to-use REST API provided by IBM
  Supports integration via cURL, Java, JavaScript, Python, Scala

RESULT



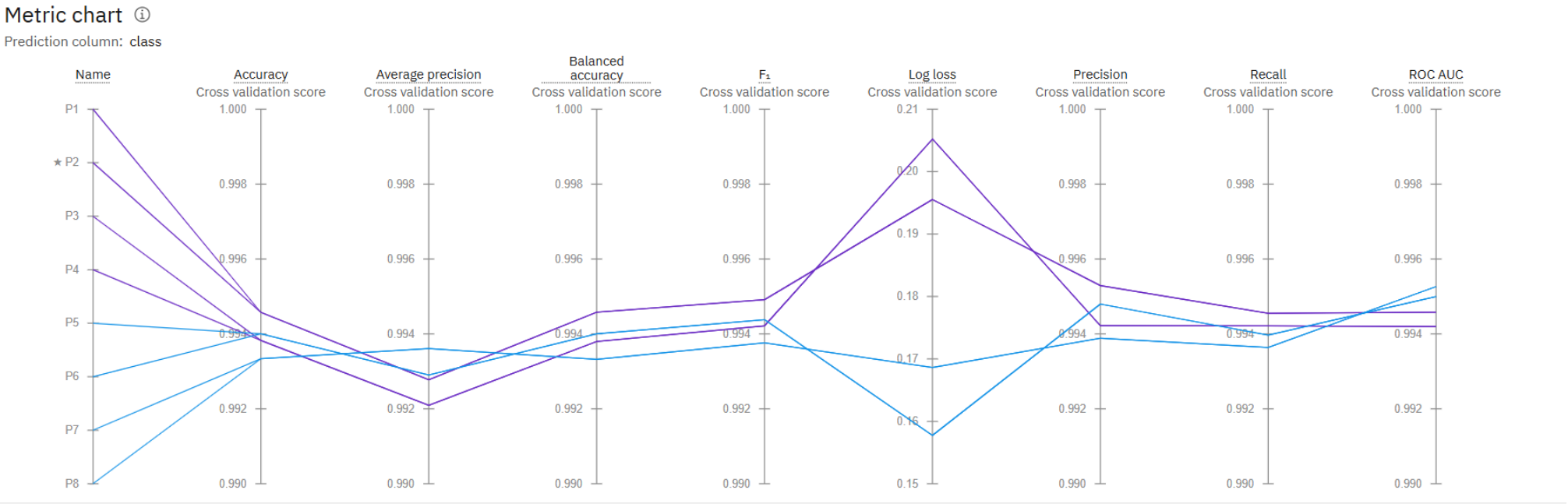
- **Prediction Type:** Binary classification (Anomaly vs. Normal)
- **Total Records Analyzed:** 22,544
- **Class Distribution:**
  - **Anomaly (Purple):**
    Represents malicious or suspicious network activity
  - **Normal (Blue):**
    Represents safe, expected traffic
- **Model Confidence**:
  High—most predictions fall in the 90–100% confidence range
- **Insight**:
  The model effectively distinguishes between normal and anomalous traffic, supporting its use in real-time intrusion detection
- **Visualization Purpose**:
  Confirms balanced classification and strong model generalization

RESULT



- **Chart Title:** Confidence Level Distribution
- **Confidence Range**: All predictions fall within the 90–100% confidence bracket
- **Class Breakdown:**
  - **Normal (Blue):** ~17,000 records
  - **Anomaly (Purple):** ~5,000 records
- **Interpretation:**
  The model shows high reliability in its predictions, with no records falling below 90% confidence
- **Significance:**
  This strong confidence distribution supports the model's robustness and suitability for real-time deployment in network intrusion detection
- **Visual Insight:**
  The bar chart confirms that both anomaly and normal classifications are made with high certainty, reducing false positives and negatives

RESULT



- **Best Pipeline Selected**: ⭐ P2 – Snap Decision Tree Classifier
- **Build Time**: ~9 seconds
- **Confidence Range**: 90–100% for most predictions

.

CONCLUSION

- **The proposed NIDS successfully detects network intrusions using machine learning. IBM AutoAI and watsonx.ai Studio streamlined the entire process—from data preprocessing to model selection and deployment—resulting in a highly accurate and efficient classifier. The Snap Decision Tree model is now deployed via IBM's cloud-native interface, offering secure and scalable access through REST APIs**

FUTURE SCOPE

1.Integrate Real-Time Traffic Monitoring and Prediction

Enhance the system to ingest and analyze live network traffic for immediate threat detection.

2.Expand to Multi-Class Classification for Attack Types

Move beyond binary classification to identify specific attack categories like DoS, Probe, R2L, and U2R.

3.Deploy on Edge Devices for Faster Response

Optimize models for lightweight deployment on routers, gateways, or Raspberry Pi for low-latency detection.

4.Use Advanced Models like LSTM for Sequential Traffic Analysis

Incorporate deep learning models to capture temporal patterns in network flows and improve accuracy.

4.Build a Dashboard Using Streamlit for Live Visualization

Create an interactive frontend to monitor predictions, traffic stats, and alerts in real time.

5.Integrate Threat Intelligence Feeds

Enrich detection by correlating traffic with known malicious IPs, domains, and signatures.

6.Enable Auto-Retraining with New Data

Set up pipelines to periodically retrain models using fresh traffic data to adapt to evolving threats.

7.Benchmark Against Industry Standards (e.g., CICIDS, UNSW-NB15)

Validate model performance using diverse, real-world datasets for robustness.

8.Collaborate with SOC Teams for Real-World Feedback

Integrate the system into Security Operations Centers to gather insights and improve usability.

REFERENCES

- 1. IBM watsonx.ai Studio

  cloud.ibm.com/catalog/services/watsonxai-studio
- 2. Kaggle NIDS Dataset

  https://www.kaggle.com/datasets/sampadab17/network-intrusion-detection
- 3. NSL-KDD Dataset Overview

  ijert.org
- 4. AutoAI Anomaly Detection Tutorial

  ibm.com/docs
- 5.Concept of NIDS using AIML

  https://youtu.be/BITK2pPcQaQ?si=IxQHtuz5Ly4PT3_A

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence

IBM SkillsBuild

IBM

SHACHIN P R

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 16, 2025

Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/128546df-ef34-404c-89db-51dfc690b923

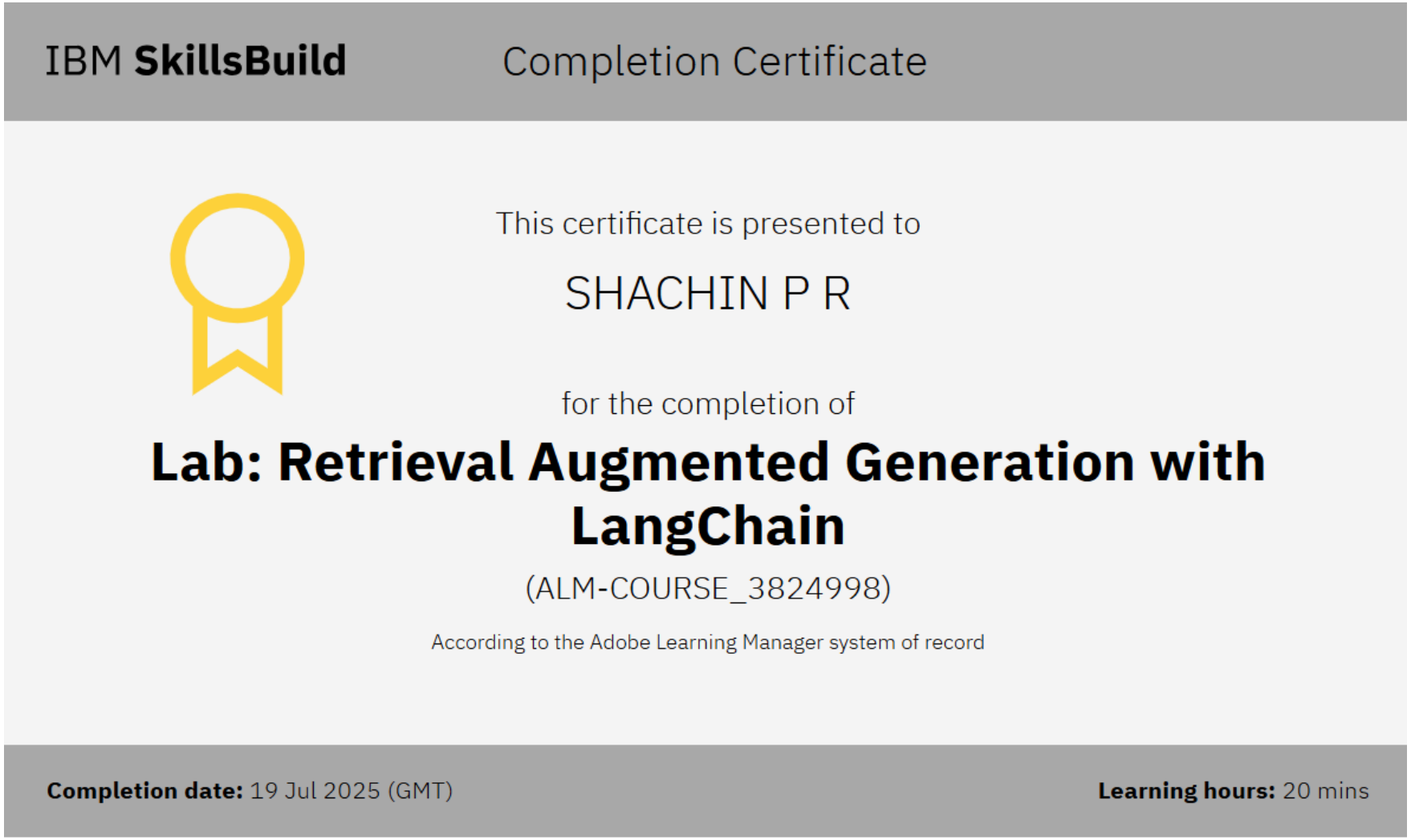
IBM

edunet foundation

IBM CERTIFICATIONS

edunet foundation

IBM CERTIFICATIONS



IBM **SkillsBuild** Completion Certificate

This certificate is presented to

SHACHIN P R

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 19 Jul 2025 (GMT) **Learning hours:** 20 mins

edunet foundation

GITHUB LINK

GITHUB REPOSITORIES LINK:

https://github.com/shachinpr29/NIDS-AIML

PPT DOWNLOAD LINK:

https://github.com/shachinpr29/NIDS-AIML/blob/main/ProjectTemplate%20(1).pptx

THANK YOU

edunet foundation