CAPSTONE PROJECT

NETWORK INTRUSION DETECTION SYSTEM (NIDS) USING MACHINE LEARNING ON IBM CLOUD

Presented By:
1. Akhil Kannan-Amrita Vishwa Vidyapeetham, Amritapuri-B.Tech CSE

edunet foundation

OUTLINE

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

edunet foundation

PROBLEM STATEMENT

In the modern digital landscape, communication networks are constantly exposed to sophisticated cyber-attacks. It is crucial to have a system that can effectively identify and classify these malicious activities to secure the network. The challenge lies in analyzing vast amounts of network traffic data to accurately distinguish between normal and anomalous behavior, providing an early warning of potential threats. The ultimate goal is to ensure the security and integrity of communication networks by proactively detecting a wide variety of cyber-attacks, including DoS, Probe, R2L, and U2R attacks.

PROPOSED SOLUTION

The proposed system aims to address the challenge of network security by creating an intelligent intrusion detection model capable of effectively identifying and classifying malicious activities. This solution leverages machine learning to detect anomalous patterns accurately. The solution will consist of the following components:

- **Data Collection:**
  - **Gather historical network traffic data from the Network Intrusion Detection dataset available on Kaggle (specifically, Train_data.csv).**
  - **This dataset includes various features such as connection duration, protocol type, service, and data bytes, which are used to build a comprehensive view of network activity.**
- **Data Preprocessing:**
  - **Clean and prepare the raw data by handling missing values, outliers, and inconsistencies.**
  - **Utilize automated feature engineering and data transformation techniques within IBM AutoAI to prepare both numerical and categorical features into a format suitable for model training.**
- **Machine Learning Algorithm:**
  - **Implement a machine learning algorithm, specifically the P2 - Snap Decision Tree Classifier, to predict the type of network activity. This model was chosen by AutoAI as the best-performing pipeline.**
  - **Use the class column from the Train_data.csv file as the target variable, where the model is trained to classify connections as either normal or an anomaly.**
- **Prediction:**
  - **The trained model, named IntrusionNet, will make predictions on new, unseen network traffic data (Test_data.csv) to classify it as normal or anomalous.**
  - **This allows the system to provide an immediate alert of a potential intrusion, thereby securing the network.**
- **Deployment:**
  - **Deploy the machine learning model to a scalable and reliable platform, such as an IBM Cloud Deployment Space.**
  - **This deployment creates a REST API endpoint for real-time inference, allowing the model to be integrated with other network monitoring applications.**
- **Evaluation:**
  - **Assess the model's performance using appropriate metrics for imbalanced classification problems, such as Accuracy, Precision, Recall, and F1-Score.**
  - **The model's performance is fine-tuned based on these metrics to ensure high-quality and reliable threat detection.**

SYSTEM APPROACH

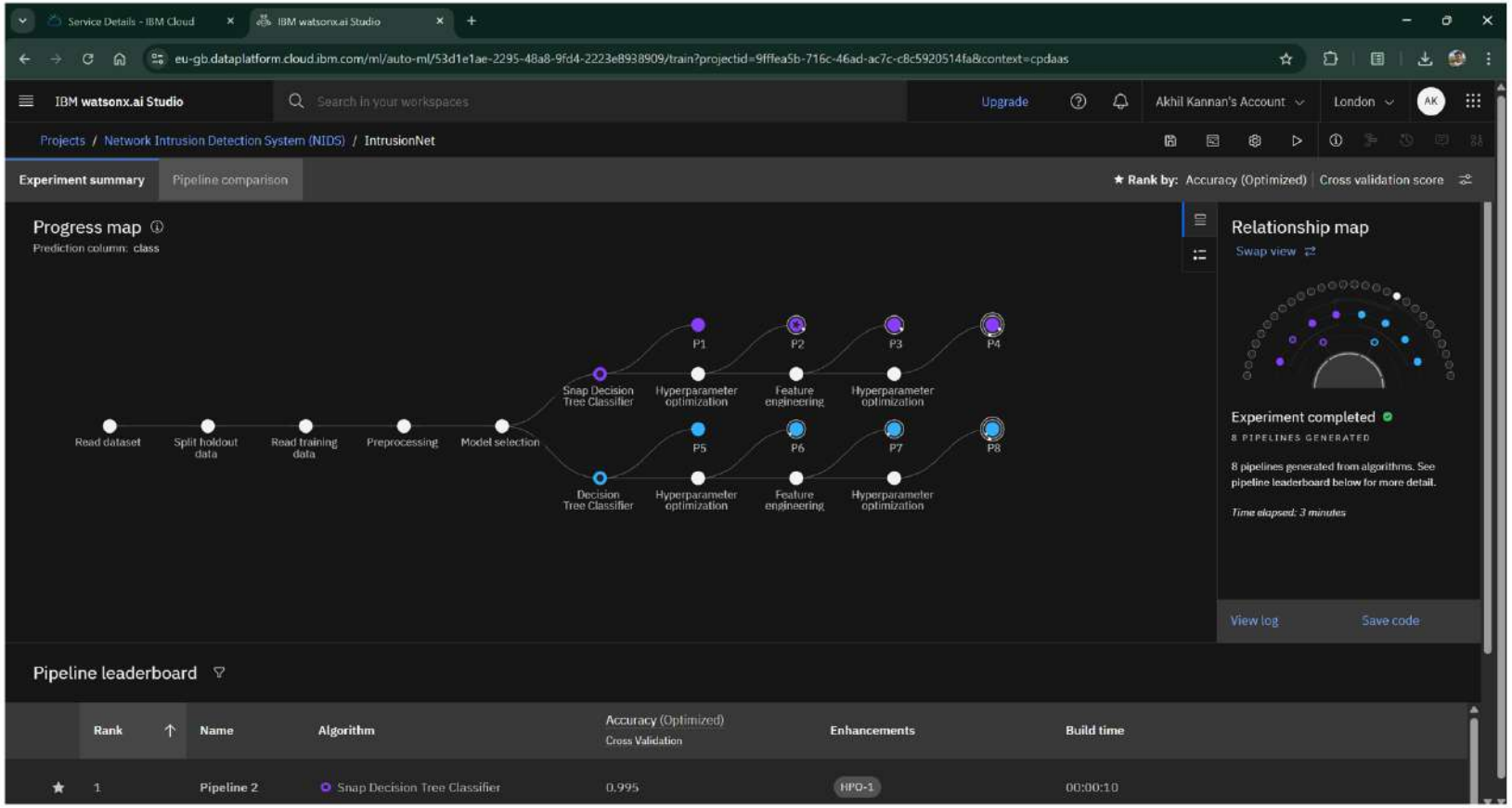
This section outlines the overall strategy and methodology for developing and implementing the Network Intrusion Detection System.

- System Requirements
  - **Platform:** A robust cloud environment for data science and model deployment. The project is built entirely on **IBM Cloud Lite services**.
  - **Tools:** The primary tool is **IBM Watson Studio**, which provides the environment for notebook development and AutoAI.
  - **Data Storage:** Data is managed using **IBM Cloud Object Storage** to store the Kaggle dataset.
  - **Deployment:** The final model is deployed to an **IBM Cloud Deployment Space** to create a functional API endpoint.

ALGORITHM & DEPLOYMENT

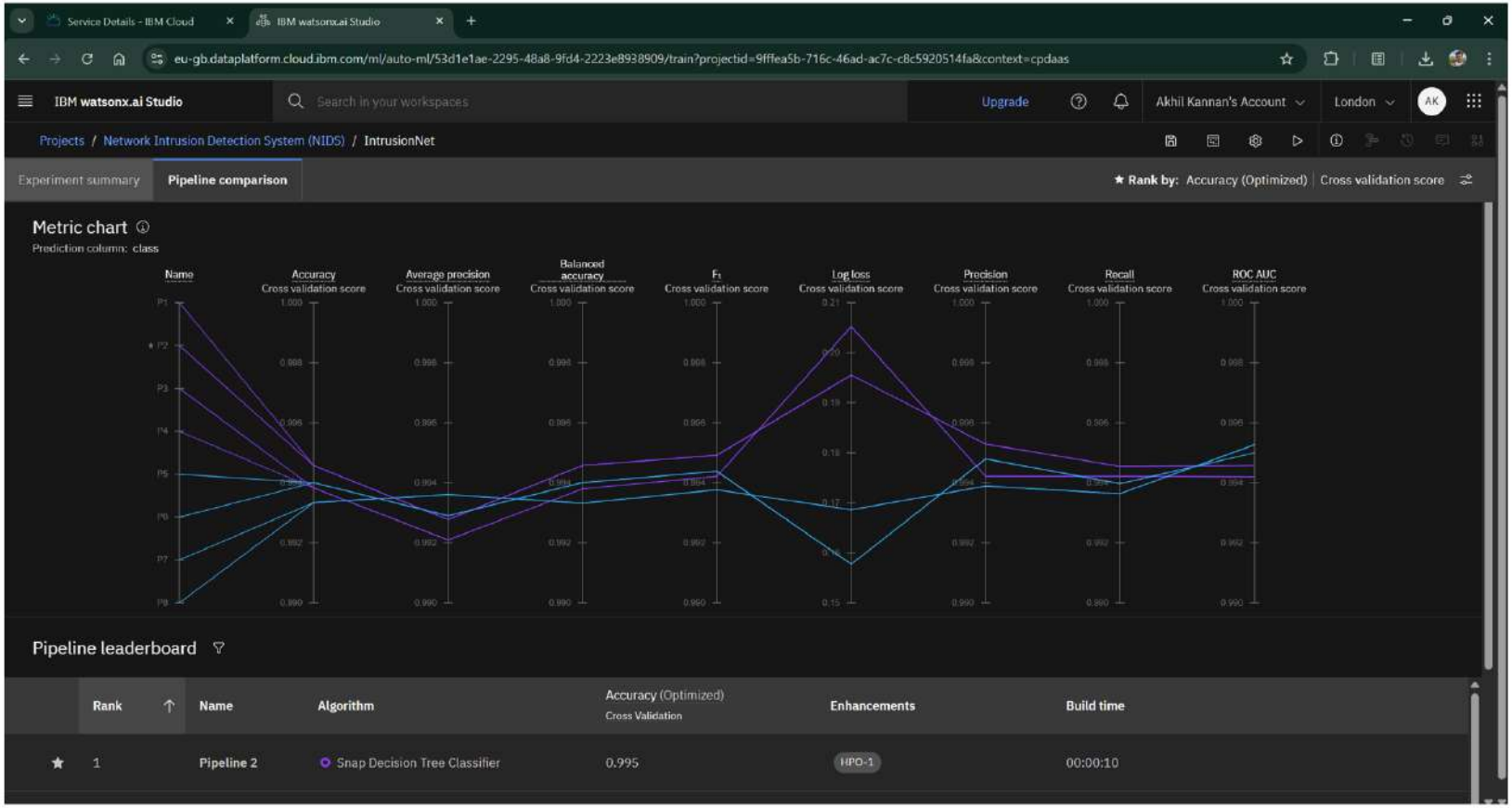
- In this section, we describe the machine learning model chosen and how it was deployed.
- **Algorithm Selection:**
  - The core of our solution is the P2 - Snap Decision Tree Classifier, which was automatically selected and optimized by IBM's AutoAI.
  - This algorithm was chosen by AutoAI due to its superior performance on the dataset during the automated evaluation process, providing a high level of accuracy and a clear decision-making process.
- **Data Input:**
  - The algorithm uses 41 features from the KDD Cup 1999 dataset.
  - Key features include duration, protocol_type, service, flag, and various statistics on connection errors and host traffic.
- **Training Process:**
  - The model was trained using the labeled Train_data.csv file, with the class column serving as the target variable.
  - AutoAI automatically handled cross-validation and hyperparameter tuning, ensuring the model was robust and well-optimized for the classification task.
- **Prediction & Deployment Process:**
  - The trained model, named IntrusionNet, makes predictions by classifying new network traffic data from the Test_data.csv file as either normal or anomaly.
  - The model was deployed to a dedicated IBM Cloud Deployment Space, providing a REST API endpoint that can be used to make real-time predictions for early intrusion warnings.

RESULT



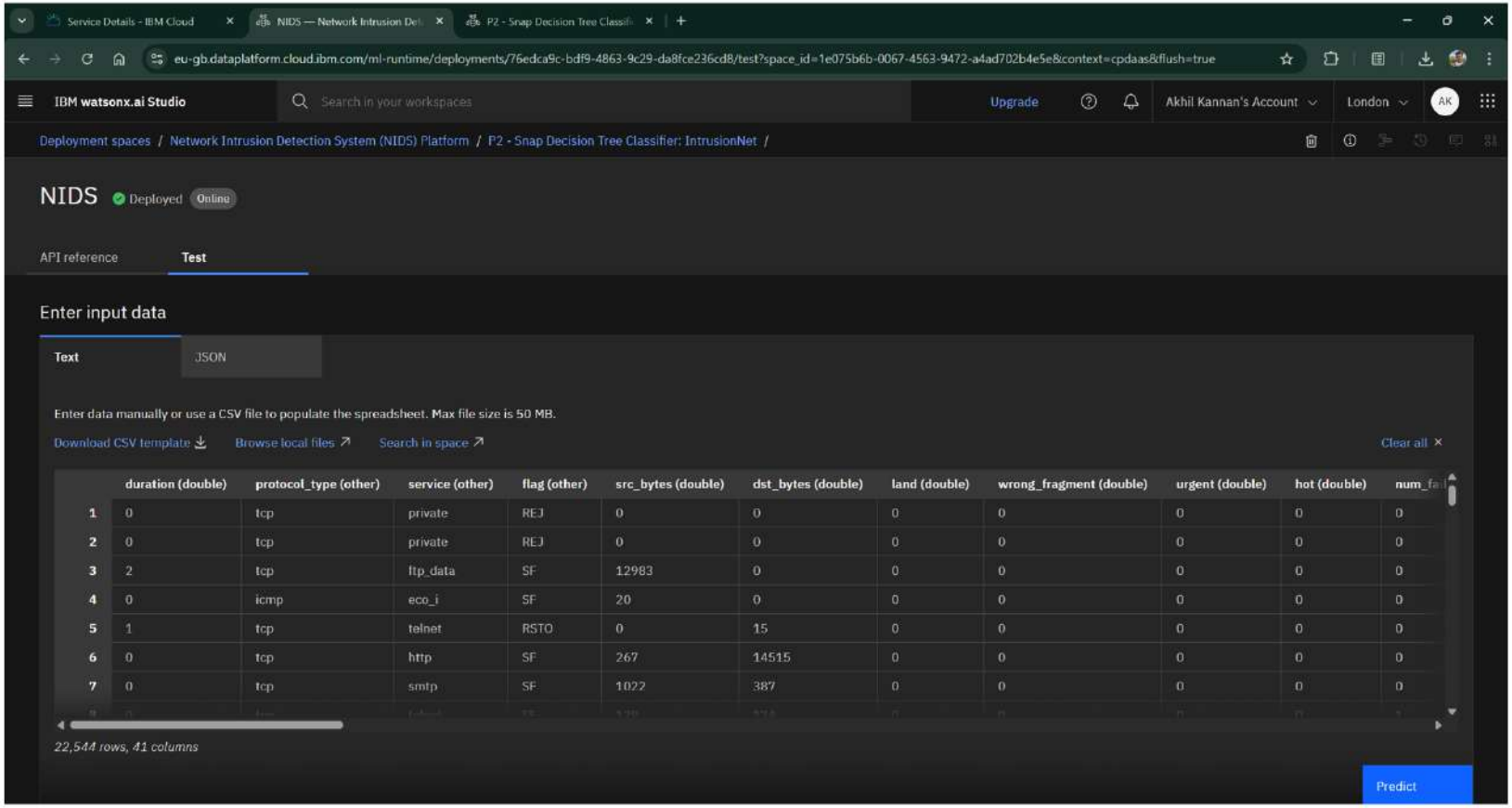
edunet foundation

RESULT



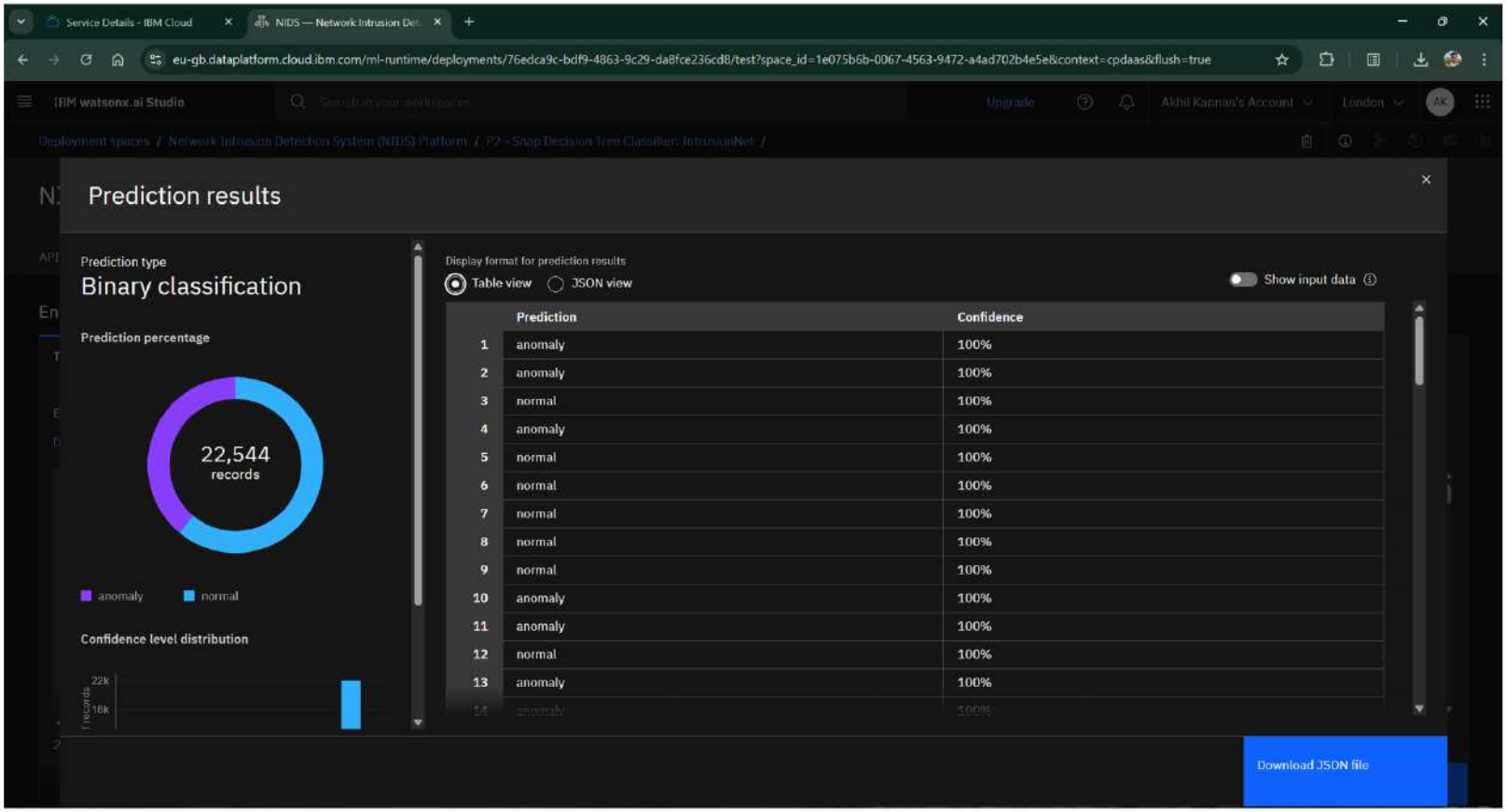
edunet foundation

RESULT



edunet foundation

RESULT



edunet foundation

CONCLUSION

- This project successfully developed and deployed a robust Network Intrusion Detection System using machine learning on the IBM Cloud platform. By leveraging IBM Watson Studio's AutoAI feature, we were able to efficiently preprocess the KDD Cup 1999 dataset and train a high-performing model without manual intervention. The final model, IntrusionNet (P2 - Snap Decision Tree Classifier), demonstrated its effectiveness in distinguishing between normal and malicious network traffic. The solution provides a valuable proof-of-concept for using AI to enhance network security by providing early warnings of potential cyber-attacks, thereby safeguarding communication networks.

FUTURE SCOPE

- The system can be enhanced and expanded in several ways. Firstly, retraining the model on more recent network traffic datasets, such as the CIC-IDS-2017, would enable it to detect modern and sophisticated attack vectors. The system could be expanded to handle real-time streaming data directly from a network, providing continuous monitoring and immediate alerts. Furthermore, integrating the model with an automated response system could allow for instant mitigation of detected threats. Finally, exploring advanced machine learning techniques, such as deep learning models or ensemble methods, could potentially further optimize model performance and accuracy.

REFERENCES

- KDD Cup 1999 Dataset: The official dataset used for the project, available on Kaggle.
- IBM Watson Studio Documentation: Guides and tutorials on using Watson Studio, AutoAI, and Deployment Spaces.
- Machine Learning Algorithms: Academic papers and resources on the specific algorithms used (e.g., Decision Trees, Gradient Boosting).
- IBM SkillsBuild: The official platform for the internship, providing resources and educational materials on AI and machine learning.

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence

IBM

Akhil Kannan

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 17, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/08a521b1-44fb-4aaf-a07d-b29128aa0620

IBM

edunet foundation

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Journey to Cloud: Envisioning Your Solution

IBM

Akhil Kannan

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: Jul 18, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/1606d6d1-95c2-416a-90cb-42b2c29cab86

IBM

edunet foundation

IBM CERTIFICATIONS

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Akhil Kannan

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 24 Jul 2025 (GMT) **Learning hours:** 20 mins

edunet foundation

THANK YOU

edunet
foundation