CAPSTONE PROJECT

PROBLEM STATEMENT NO.40 – NETWORK INTRUSION DETECTION

Presented By: Mahak_Kumrawat-Shri Govindram Seksaria Institute Of Technology And Science-Electronics And Telecommunications

OUTLINE

- **Problem Statement** (Should not include solution)
- **Proposed System/Solution**
- **System Development Approach** (Technology Used)
- **Algorithm & Deployment**
- **Result (Output Image)**
- **Conclusion**
- **Future Scope**
- **References**

PROBLEM STATEMENT

- 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

- Proposed System: Network Intrusion Detection using Machine LearningThe proposed system aims to detect and classify various cyber-attacks (e.g., DoS, Probe, R2L, U2R) by analyzing network traffic using machine learning to secure communication networks.
- Data Collection:Use labeled Kaggle dataset containing network traffic features and attack categories.
- Data Preprocessing:Clean data, encode categorical features, and normalize inputs for better model performance.
- Machine Learning Algorithm:Train an XGBoost classifier to distinguish between normal and malicious traffic.
- Deployment:Deploy the trained model on IBM Watson Machine Learning for real-time inference.
- Evaluation:Evaluate using accuracy, F1-score, and AUC to ensure robust attack detection.
- Result:Achieved high accuracy in identifying network intrusions with potential for real-time security monitoring.

SYSTEM APPROACH

- System Requirements
- **Processor**: Intel i5 / AMD Ryzen 5 or better
- **RAM**: Minimum 8 GB (16 GB recommended)
- **Storage**: At least 10 GB free disk space
- **Operating System**: Windows 10/11, Linux (Ubuntu), or macOS
- **Internet**: Stable connection for IBM Cloud services
- **GPU** *(optional)*: NVIDIA GPU for faster training (if using deep learning)
- **Cloud Platform**: IBM Cloud Lite (Free tier is sufficient)

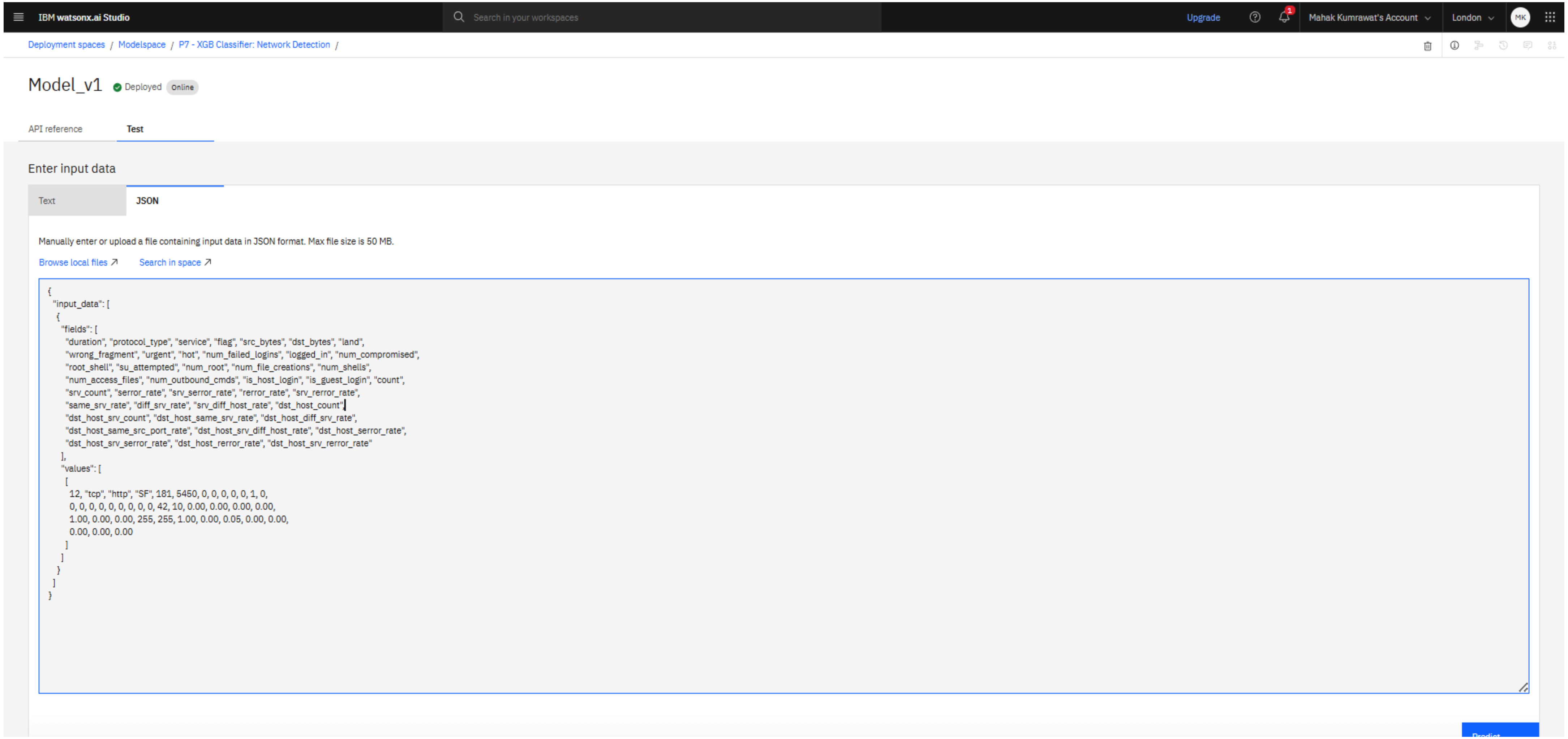
SYSTEM APPROACH

- Required Python Libraries
- pandas – Data manipulation and CSV handling
- numpy – Numerical computations
- scikit-learn – Data preprocessing, model building, evaluation
- xgboost – ML algorithm for intrusion detection (XGBoost Classifier)
- matplotlib – Data visualization (basic plots)
- seaborn – Advanced visualizations (heatmaps, distribution plots)
- joblib – Save and load trained models
- IBM Watson ai studio
- IBM Watson Auto AI

ALGORITHM & DEPLOYMENT

- Algorithm Selection: XGBoost Classifier was chosen for its speed, accuracy, and ability to handle tabular data with class imbalance—ideal for multi-class intrusion detection.
- Data Input: Features include network parameters like protocol type, service, source/destination bytes, failed logins, and connection counts. The target is the attack category (Normal, DoS, Probe, R2L, U2R).
- Training Process: Data was preprocessed, split into training/testing sets, and tuned using cross-validation and grid search. Evaluation metrics include accuracy, precision, recall, and F1-score.
- Prediction Process: The trained model classifies real-time network traffic to detect and label potential attacks.

RESULT


IBM watsonx.ai Studio
Search in your workspaces
Upgrade
Mahak Kumrawat's Account
London
MK
Deployment spaces / Modelspace / P7 - XGB Classifier: Network Detection /
API reference
Test
Text
JSON

Model_v1 Deployed Online

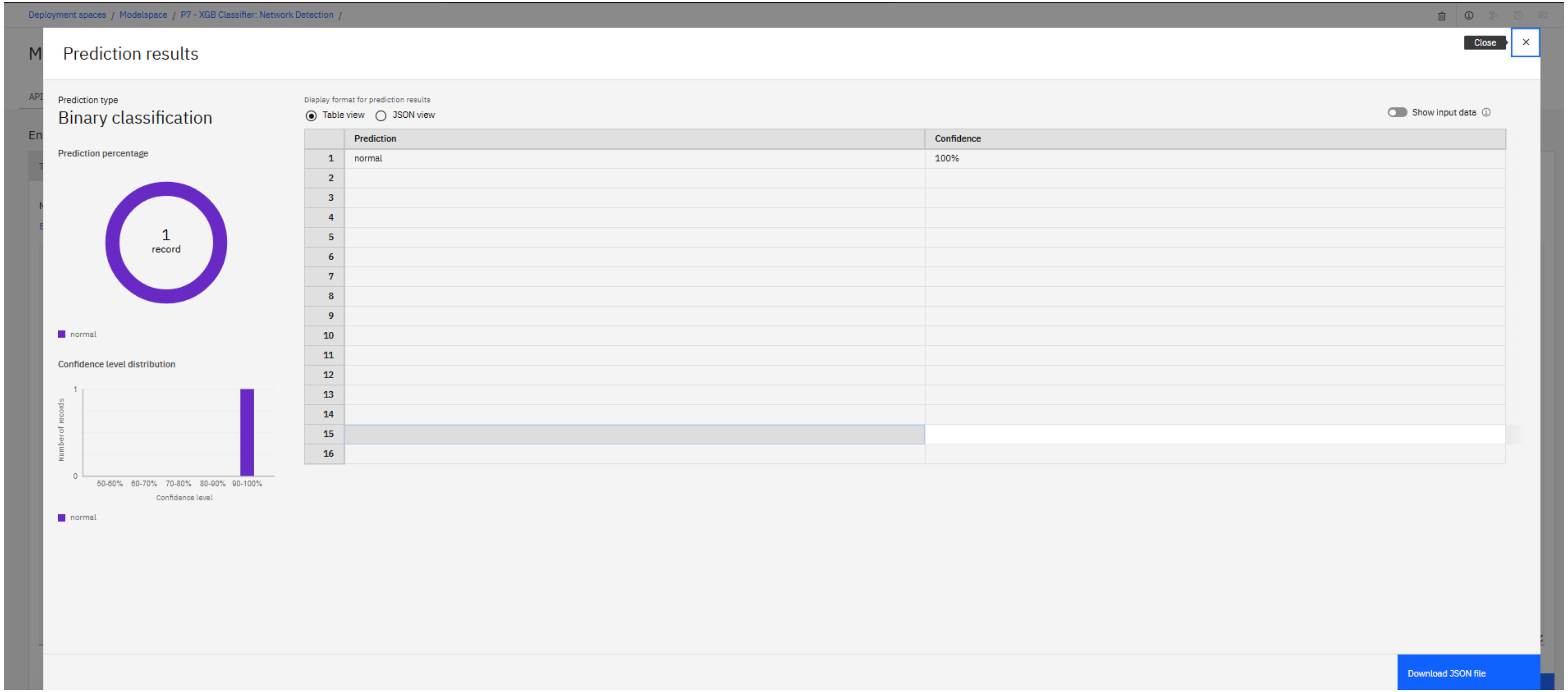
Enter input data

Manually enter or upload a file containing input data in JSON format. Max file size is 50 MB.

Browse local files ↗ Search in space ↗

```
{
 "input_data": [
  {
   "fields": [
    "duration", "protocol_type", "service", "flag", "src_bytes", "dst_bytes", "land",
    "wrong_fragment", "urgent", "hot", "num_failed_logins", "logged_in", "num_compromised",
    "root_shell", "su_attempted", "num_root", "num_file_creations", "num_shells",
    "num_access_files", "num_outbound_cmds", "is_host_login", "is_guest_login", "count",
    "srv_count", "serror_rate", "srv_serror_rate", "rerror_rate", "srv_rerror_rate",
    "same_srv_rate", "diff_srv_rate", "srv_diff_host_rate", "dst_host_count",
    "dst_host_srv_count", "dst_host_same_srv_rate", "dst_host_diff_srv_rate",
    "dst_host_same_src_port_rate", "dst_host_srv_diff_host_rate", "dst_host_serror_rate",
    "dst_host_srv_serror_rate", "dst_host_rerror_rate", "dst_host_srv_rerror_rate"
   ],
   "values": [
    [
     12, "tcp", "http", "SF", 181, 5450, 0, 0, 0, 0, 0, 1, 0,
     0, 0, 0, 0, 0, 0, 0, 0, 0, 42, 10, 0.00, 0.00, 0.00, 0.00,
     1.00, 0.00, 0.00, 255, 255, 1.00, 0.00, 0.05, 0.00, 0.00,
     0.00, 0.00, 0.00
    ]
   ]
  }
 ]
}
```

RESULT



Deployment spaces / Modelspace / P7 - XGB Classifier: Network Detection /

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		
12		
13		
14		
15		
16		

Download JSON file

CONCLUSION

- Findings: XGBoost achieved high accuracy in detecting and classifying network intrusions. Effectively handled multiple attack types (DoS, Probe, R2L, U2R).
- Challenges: Imbalanced dataset for rare attacks. Complex preprocessing of features.
- Importance: Accurate intrusion detection is essential for securing networks—just like accurate bike demand prediction ensures availability in rental systems.

FUTURE SCOPE

- Improvements
- Use deep learning for better detection of rare attacks.
- Enable real-time deployment and model updates.

.

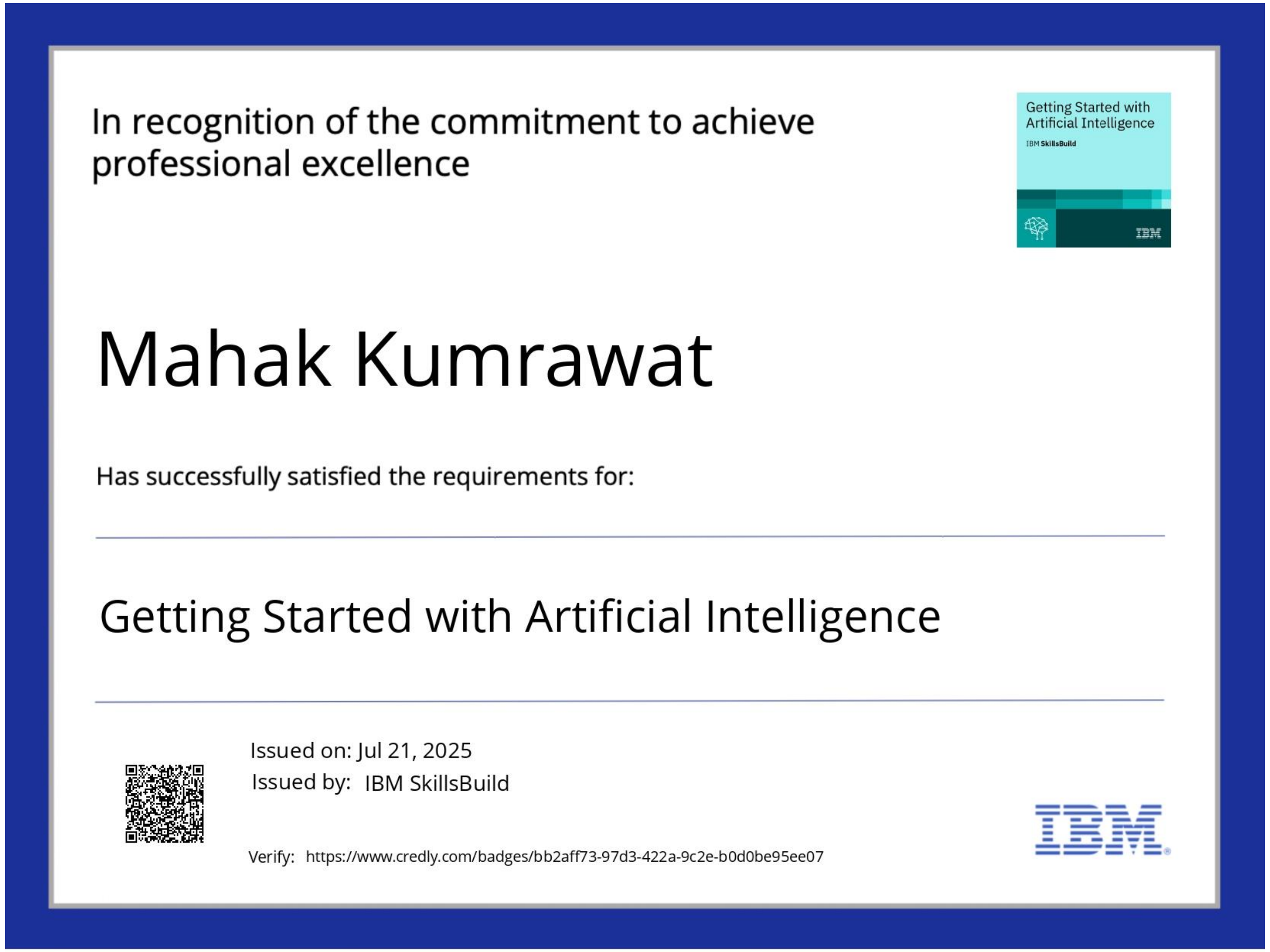
REFERENCES

- Scikit-learn documentation. (n.d.). *Machine Learning in Python*. https://scikit-learn.org/
→ For data preprocessing, model training, and evaluation best practices.

- IBM Watson Machine Learning. (n.d.). *Model deployment and management on IBM Cloud*. https://www.ibm.com/cloud/machine-learning
→ For potential deployment of the model on cloud infrastructure.

- Dua, D., & Graff, C. (2019). *UCI Machine Learning Repository: KDD Cup 1999 Data*. https://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
→ Source of benchmark dataset for network intrusion detection.

IBM CERTIFICATIONS



In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence
IBM SkillsBuild
IBM

Mahak Kumrawat

Has successfully satisfied the requirements for:

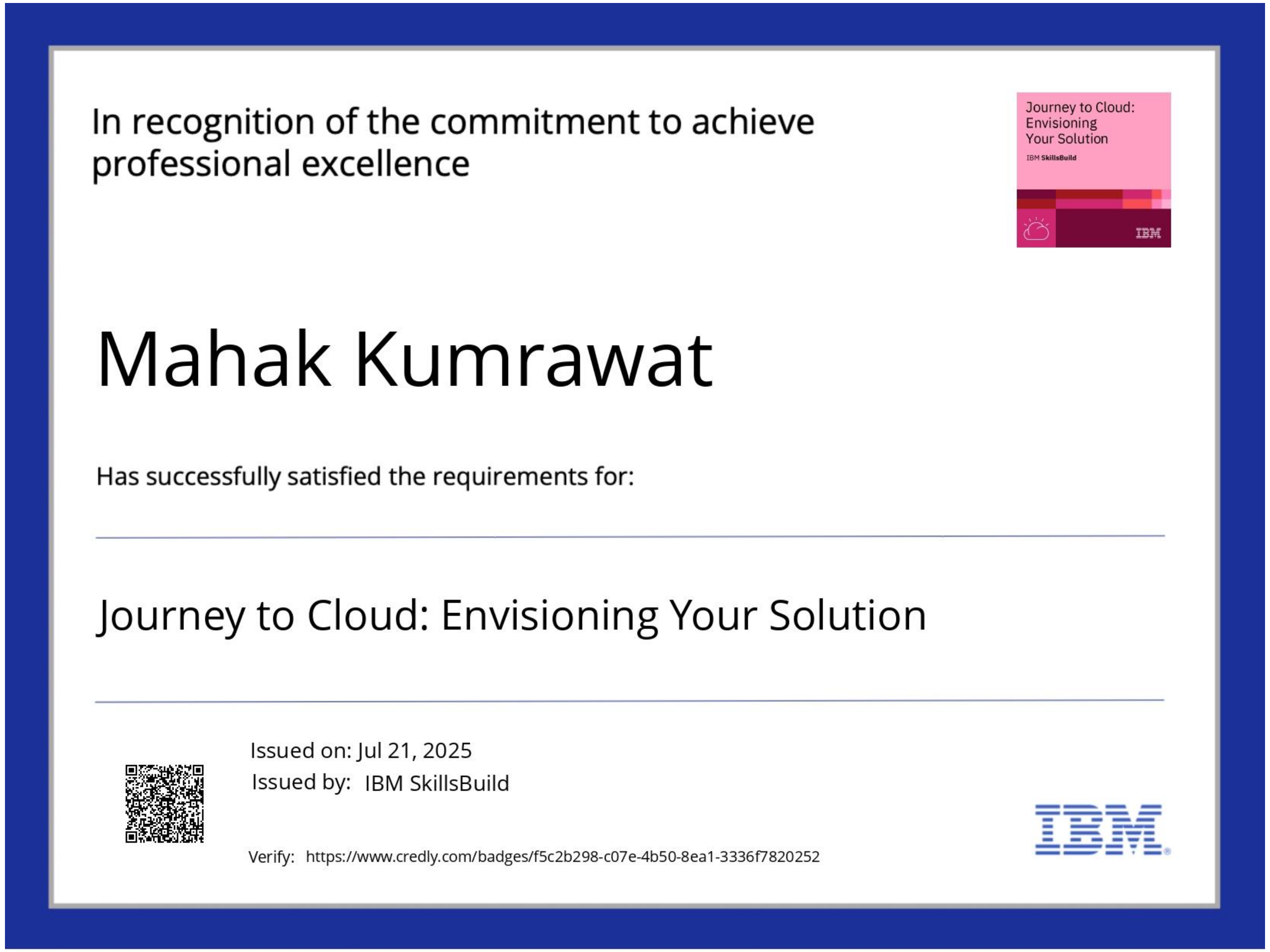
Getting Started with Artificial Intelligence

Issued on: Jul 21, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/bb2aff73-97d3-422a-9c2e-b0d0be95ee07

IBM

IBM CERTIFICATIONS



Journey to Cloud:
Envisioning
Your Solution
IBM SkillsBuild
IBM

In recognition of the commitment to achieve professional excellence

Mahak Kumrawat

Has successfully satisfied the requirements for:

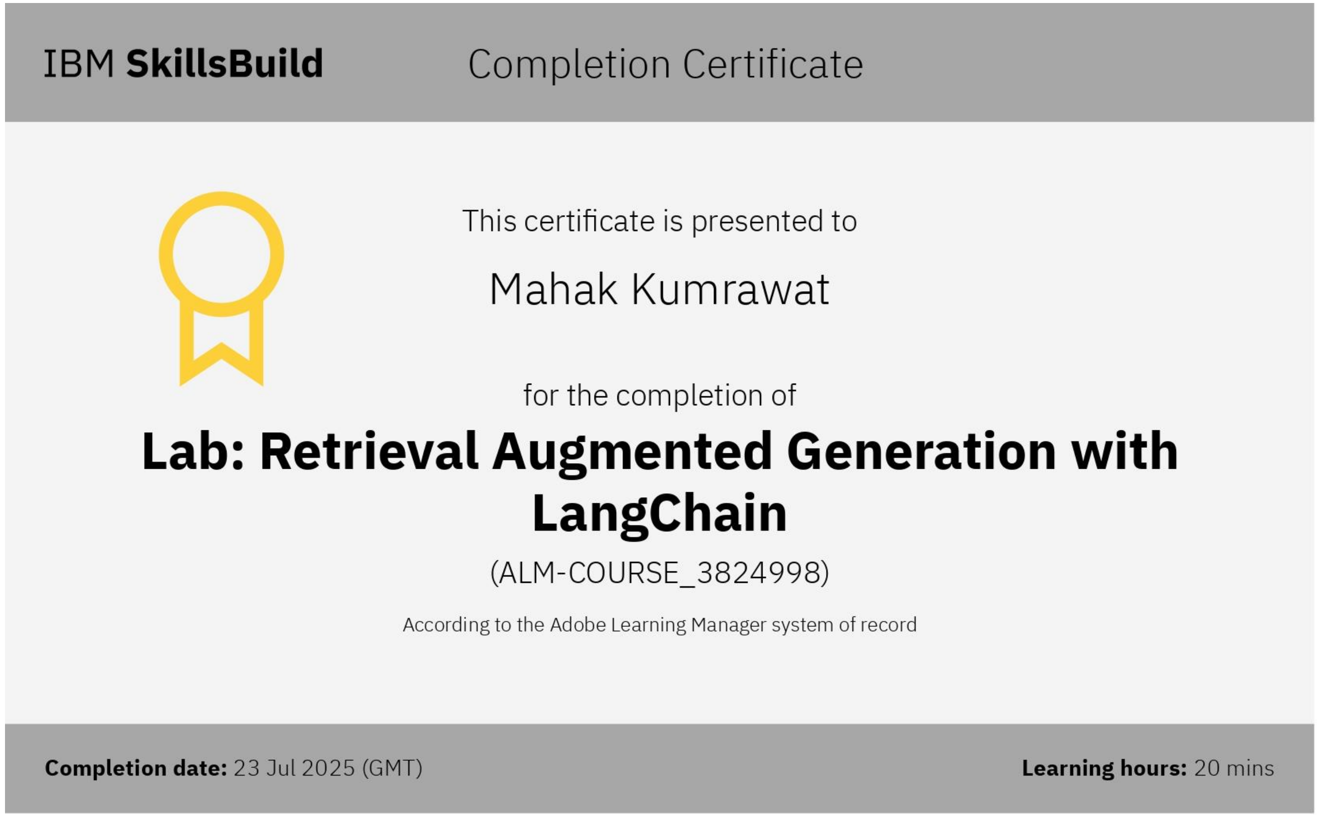
Journey to Cloud: Envisioning Your Solution

Issued on: Jul 21, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/f5c2b298-c07e-4b50-8ea1-3336f7820252

IBM

IBM CERTIFICATIONS



IBM **SkillsBuild**

Completion Certificate

This certificate is presented to

Mahak Kumrawat

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

THANK YOU

edunet
foundation