CAPSTONE PROJECT

NETWORK INTRUSION DETECTION SYSTEM USING MACHINE LEARNING

Presented By:
Gundrasam Pavan Kumar Reddy
Vellore institution of technology Amaravati
Department: Computer Science and Engineering

edunet
foundation

OUTLINE

PROBLEM STATEMENT

Why NIDS matters:

- Modern networks face a barrage of sophisticated attacks—floods of traffic (DoS), stealthy probes, unauthorized logins (R2L), and privilege escalations (U2R)
- Existing signature-based tools struggle with novel or obfuscated threats
- We need an adaptive, data-driven system that spots anomalies in real time and classifies them accurately.

PROPOSED SOLUTION

The proposed system aims to address the challenge of detecting malicious network activity and classifying it from normal network behavior. This supports the early identification of security threats, helping to protect networks from cyberattacks. The solution will include the following components:

Data Collection

- Collect network traffic data with details such as protocol type, service, packet size, and connection duration.
- Use labeled data indicating whether connections were normal or suspicious (anomalous).
- Incorporate metadata about network sessions that might help reveal attack patterns.

Data Preprocessing

- Clean and preprocess the collected data to handle missing values, inconsistencies, and irrelevant fields.
- Apply encoding to categorical features such as protocol type and service names.
- Perform feature engineering to create meaningful indicators (e.g., connection rates, error counts) that can improve detection accuracy.

PROPOSED SOLUTION

Machine Learning Algorithm

- Implement a machine learning algorithm, such as a Random Forest–based ensemble classifier, to distinguish between normal and abnormal traffic.
- Use hyperparameter optimization and feature engineering to achieve the best performance.
- Optionally, if detailed attack labels are available, extend the classification to categories such as DoS, Probe, R2L, and U2R.

Deployment

- Deploy the trained model as a web service or API on IBM watsonx.ai to allow real-time predictions.
- Provide a simple interface for cybersecurity teams to submit network session data and get immediate classification results.
- Ensure the deployed system is scalable, secure, and integrates easily with existing network monitoring tools.

SYSTEM APPROACH

- The system aims to detect network intrusions in real time using machine learning. We start by analyzing a labeled dataset of network traffic and preprocessing it to clean and transform the data. Using IBM watsonx.ai, we train a Snap Random Forest Classifier with optimized features and parameters.
- Once trained, the model is deployed as an API to classify incoming traffic as normal or anomalous. Real-time predictions help flag threats quickly, enabling proactive security actions. The system is scalable, accurate, and continuously improvable with new data—making it ideal for modern network defense.

ALGORITHM & DEPLOYMENT

Algorithm Selection:

- For this project, I selected the Snap Random Forest Classifier, a tree-based ensemble machine learning model. This algorithm was chosen because of its strong performance in classification tasks, particularly when dealing with complex, high-dimensional data like network traffic logs. Random Forests are also less prone to overfitting and can handle both numerical and categorical features, making them well-suited for detecting anomalies in varied connection patterns.

Data Input:

The model was trained on labeled network traffic data with features such as:

- Duration of connection
- Protocol type
- Source and destination bytes
- Connection flags
- Count of connections to the same host/service and other flow-related attributes

These features capture both statistical and behavioral aspects of traffic, allowing the model to learn normal vs. abnormal activity patterns.

ALGORITHM & DEPLOYMENT

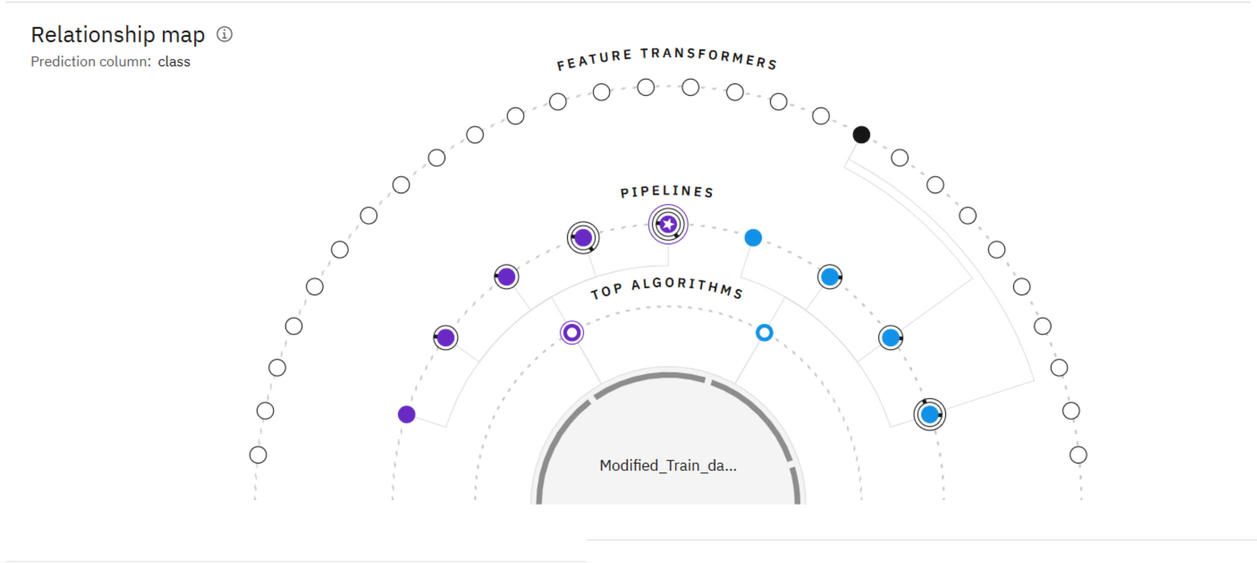
Training Process:

- The dataset was uploaded and preprocessed using watsonx.ai's AutoAI, which automatically handled:
- Feature selection and transformation
- Data normalization
- Class balancing
- The AutoAI pipeline also performed hyperparameter optimization and cross-validation to ensure the best version of the Random Forest model was selected based on accuracy, F1-score, and ROC-AUC. The training was done using historical traffic data labeled as either "normal" or "anomaly."

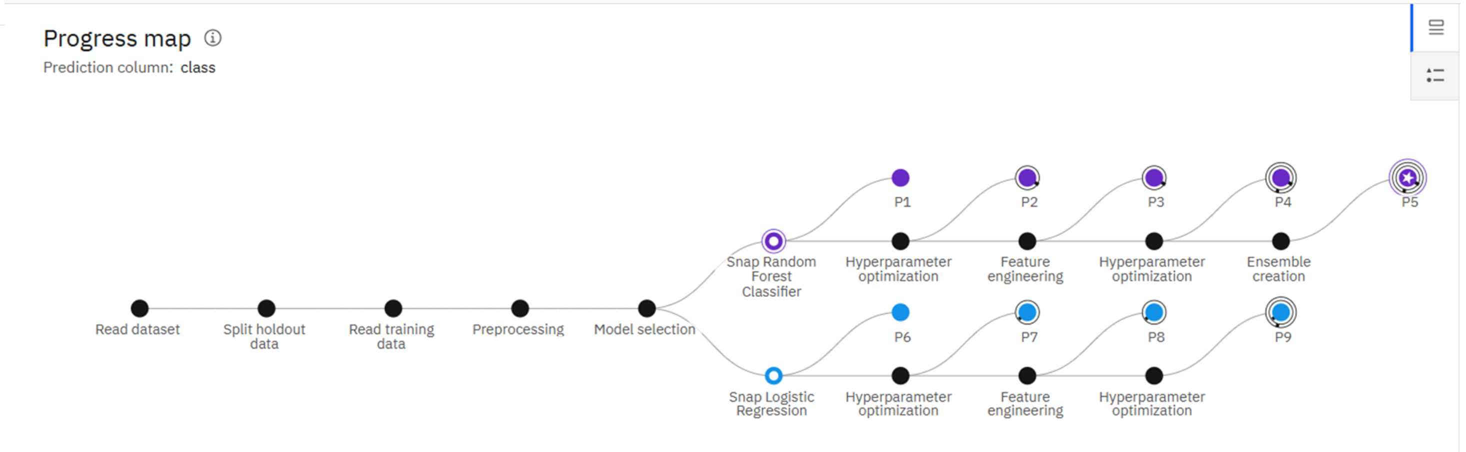
Prediction Process:

- Once deployed, the model uses live or batch input data to predict whether a given traffic flow is normal or potentially malicious. Inputs are passed as a structured JSON payload, and predictions are returned in real time via the model's REST API endpoint. This allows the system to be integrated into larger security frameworks for live threat detection and alerting.

RESULT

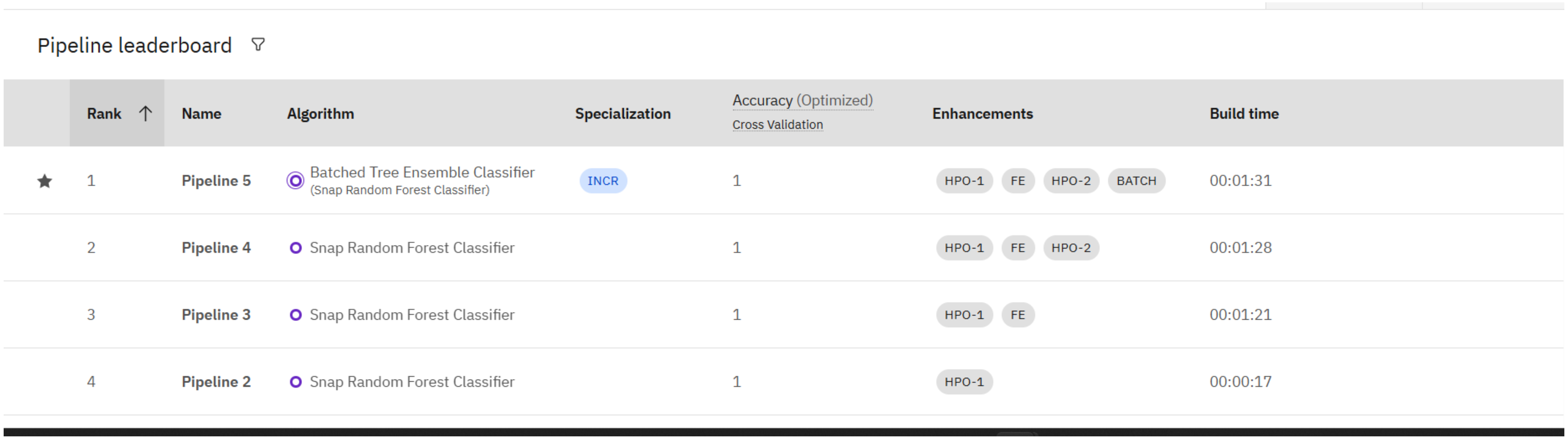


Relationship map
Prediction column: class
FEATURE TRANSFORMERS
PIPELINES
TOP ALGORITHMS
Modified_Train_da...



Progress map
Prediction column: class
Read dataset
Split holdout data
Read training data
Preprocessing
Model selection
Snap Random Forest Classifier
Hyperparameter optimization
Feature engineering
Hyperparameter optimization
Ensemble creation
P1
P2
P3
P4
P5
Snap Logistic Regression
Hyperparameter optimization
Feature engineering
Hyperparameter optimization
P6
P7
P8
P9

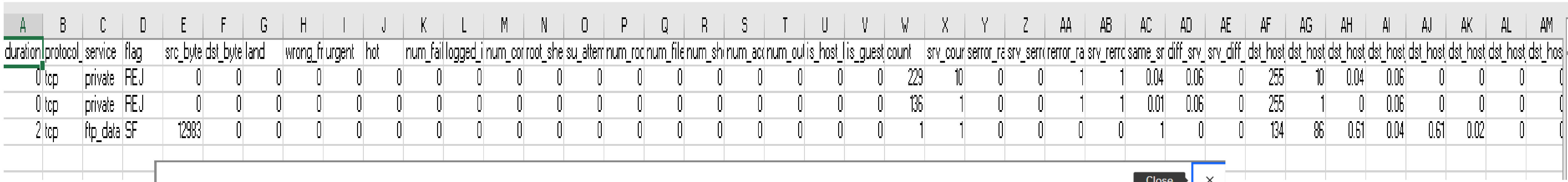
RESULT



Pipeline leaderboard

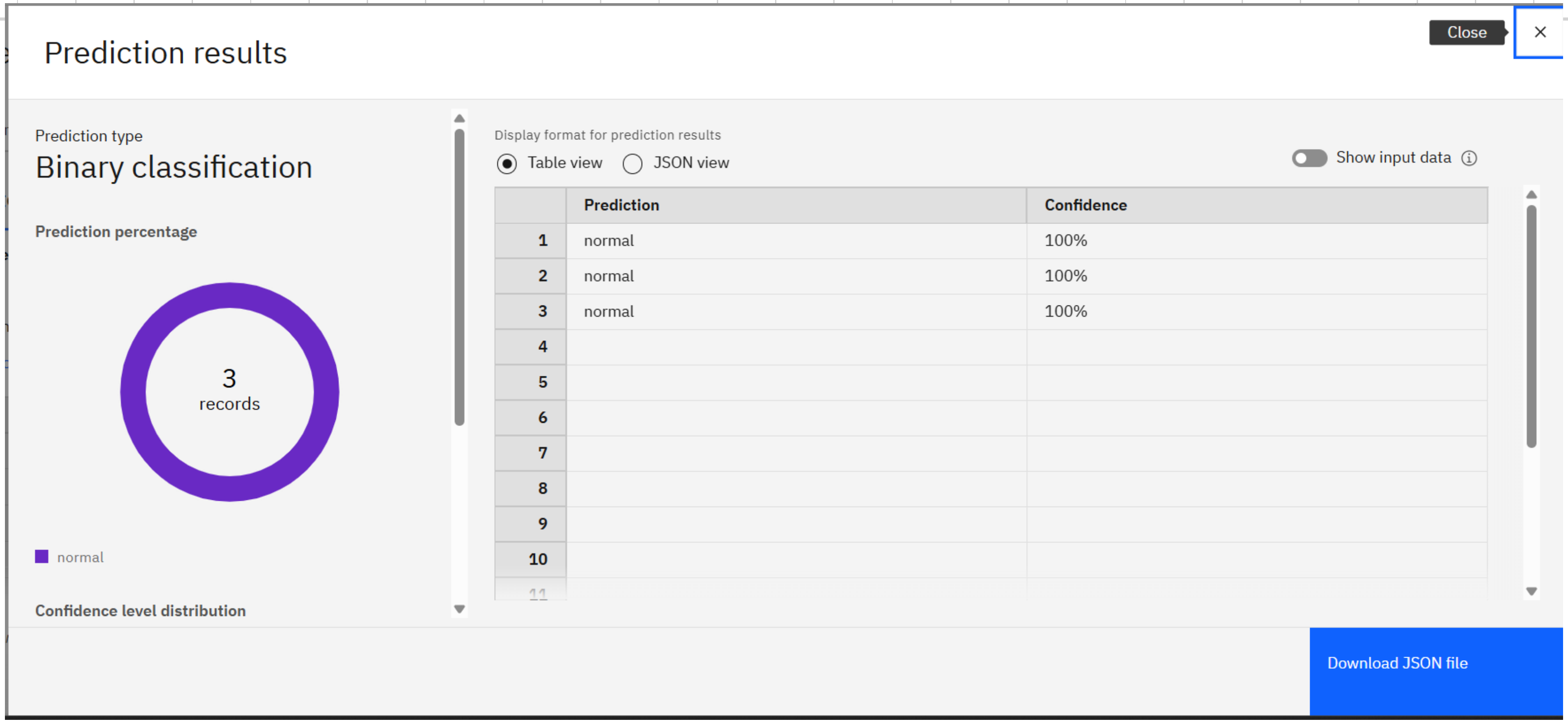
	Rank ↑	Name	Algorithm	Specialization	Accuracy (Optimized) Cross Validation	Enhancements	Build time
★	1	Pipeline 5	Batched Tree Ensemble Classifier (Snap Random Forest Classifier)	INCR	1	HPO-1 FE HPO-2 BATCH	00:01:31
	2	Pipeline 4	Snap Random Forest Classifier		1	HPO-1 FE HPO-2	00:01:28
	3	Pipeline 3	Snap Random Forest Classifier		1	HPO-1 FE	00:01:21
	4	Pipeline 2	Snap Random Forest Classifier		1	HPO-1	00:00:17

RESULT



A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
duration	protocol_	service	flag	src_byte	dst_byte	land	wrong_fr	urgent	hot	num_fail	logged_i	num_cor	root_she	su_atterr	num_roc	num_file	num_sh	num_acc
0	tcp	private	REJ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	tcp	private	REJ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	tcp	ftp_data	SF	12983	0	0	0	0	0	0	0	0	0	0	0	0	0	0

T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM
num_oul	is_host_l	is_guest	count	srv_cour	serror_ra	srv_serr	rerror_ra	srv_rerrc	same_sr	diff_srv_	srv_diff_	dst_host	dst_host	dst_host	dst_host	dst_host	dst_host	dst_host	dst_hos
0	0	0	229	10	0	0	1	1	0.04	0.06	0	255	10	0.04	0.06	0	0	0	
0	0	0	136	1	0	0	1	1	0.01	0.06	0	255	1	0	0.06	0	0	0	
0	0	0	1	1	0	0	0	0	1	0	0	134	86	0.61	0.04	0.61	0.02	0	



Prediction results

Prediction type

Binary classification

Prediction percentage

3 records

■ normal

Confidence level distribution

Display format for prediction results

[✓ Table view] JSON view

Show input data

	Prediction	Confidence
1	normal	100%
2	normal	100%
3	normal	100%
4		
5		
6		
7		
8		
9		
10		
11		

edunet foundation

CONCLUSION

- Through this project, I was able to successfully build, train, and deploy a machine learning model using IBM Watsonx.ai on IBM Cloud. By leveraging the platform's AutoAI and deployment features, I could automate much of the model-building process and generate a working API endpoint for real-time predictions.
- This hands-on experience gave me deeper insights into how enterprise-grade AI workflows operate — from data preprocessing to model evaluation and deployment. With the model now live, I can integrate it into real-world applications and continue refining it with more data or feedback.
- This project not only improved my technical skills in cloud-based AI but also gave me a clear picture of how AI solutions can be practically implemented to solve real-world problems.

edunet foundation

FUTURE SCOPE

This project marks just the beginning of what's possible with AI on the cloud. Going forward, the model can be improved and expanded in the following ways:

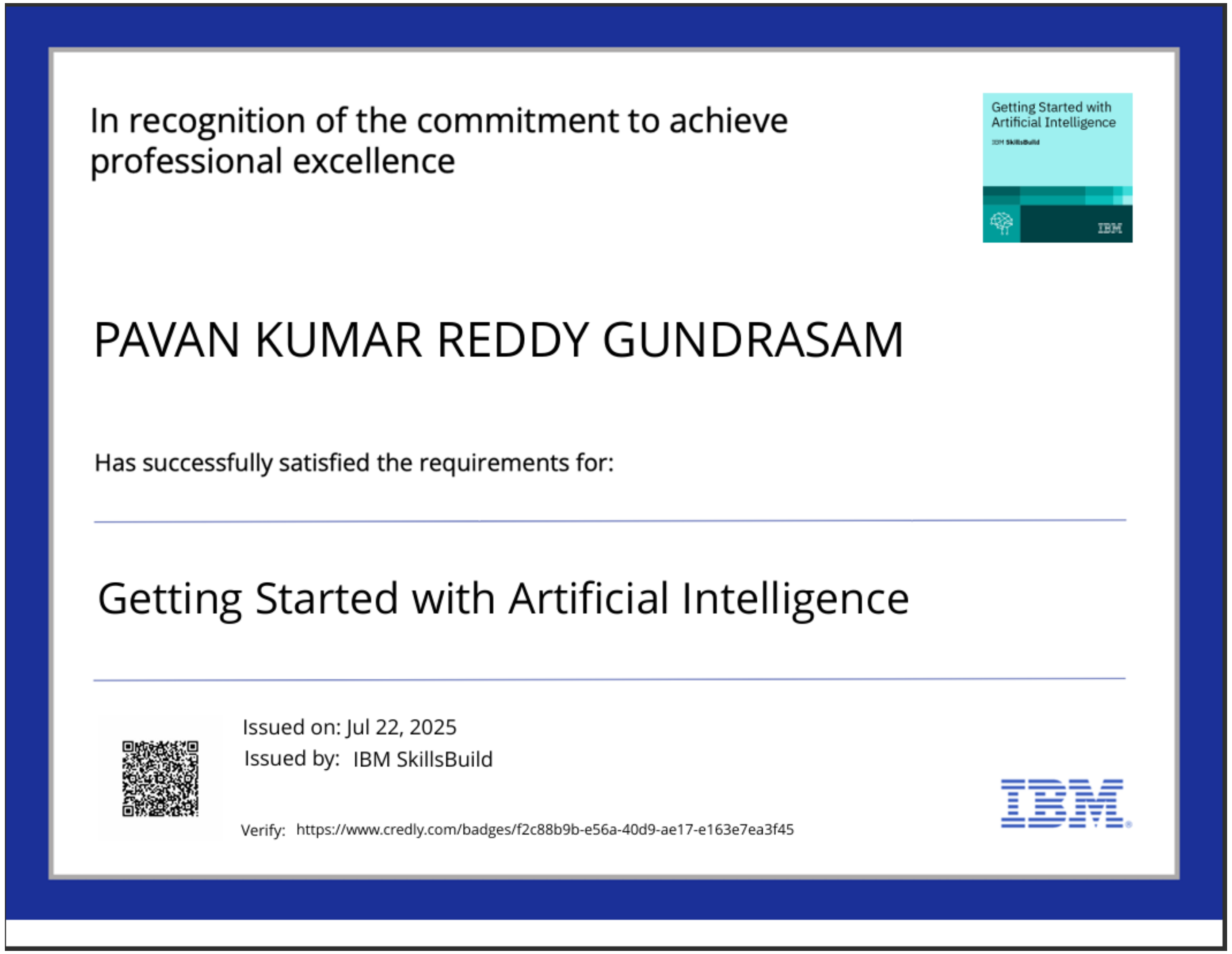
- **Data Enrichment**: Incorporate more diverse and real-time data sources to improve prediction accuracy and model generalization.
- **Model Optimization**: Explore advanced algorithms or fine-tune hyperparameters for better performance.
- **Interactive Interface**: Build a simple web or mobile app that uses the deployed API to deliver predictions in real time to end users.
- **Continuous Learning**: Implement feedback loops where the model learns from new data over time, making it smarter and more adaptive.
- **Scalability**: Deploy the model to serve more users simultaneously using load balancing and other scalable cloud infrastructure options.
- By extending and refining this project, it can evolve into a robust, production-ready AI solution tailored to specific business or social needs.

edunet foundation

REFERENCES

- IBM Watsonx.ai https://www.ibm.com/cloud/watsonx-ai
- IBM Cloud Machine Learning REST API Reference. https://cloud.ibm.com/apidocs/machine-learning
- IBM Developer Guide: Authenticating to IBM Cloud. https://cloud.ibm.com/docs/cloud-foundry?topic=cloud-foundry-using-credentials
- Scikit-learn: Machine Learning in Python. https://scikit-learn.org/
- Kaggle Dataset Repository. https://www.kaggle.com/

IBM CERTIFICATIONS



In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence

IBM SkillsBuild

IBM

PAVAN KUMAR REDDY GUNDRASAM

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 22, 2025

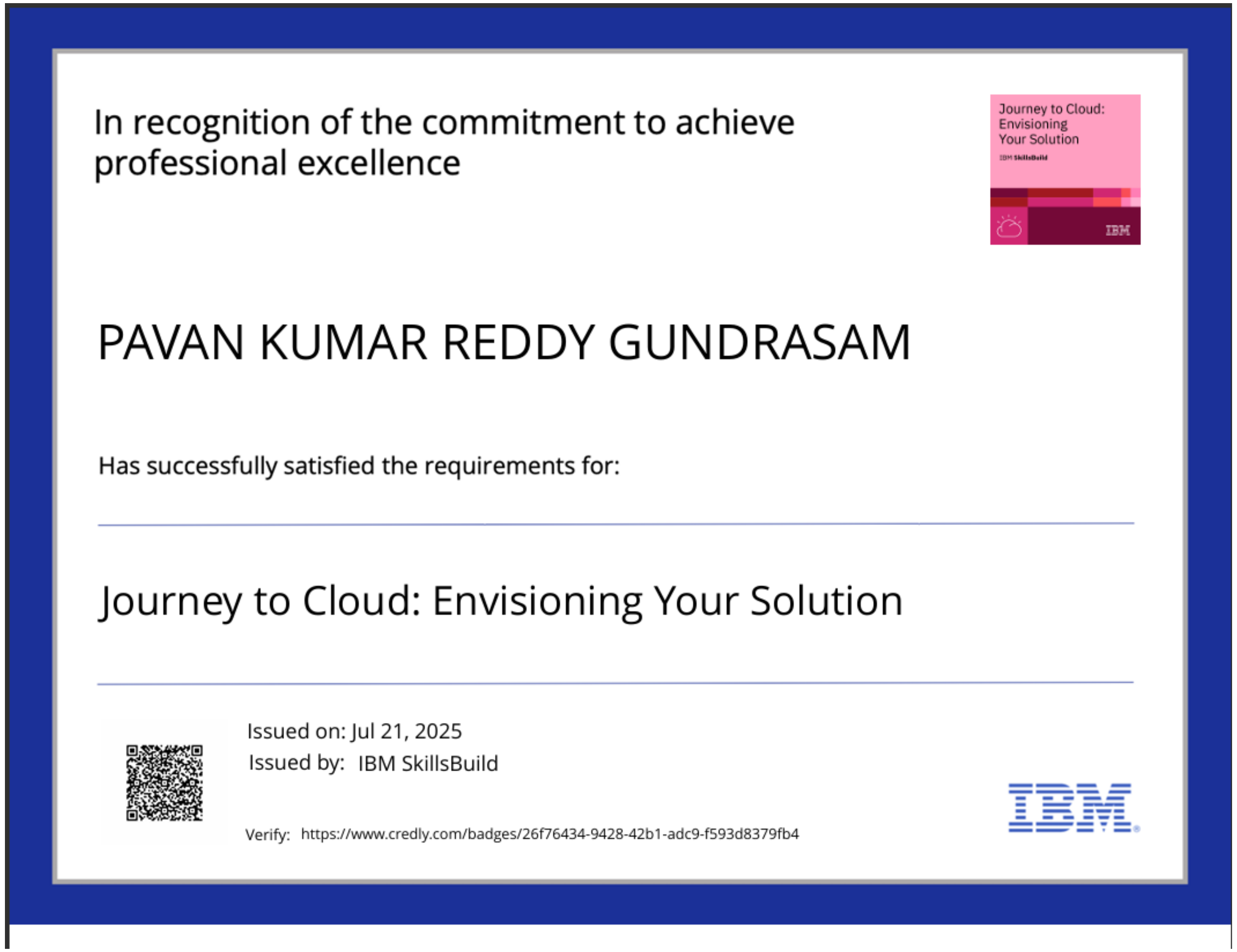
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/f2c88b9b-e56a-40d9-ae17-e163e7ea3f45

IBM

edunet foundation

IBM CERTIFICATIONS



In recognition of the commitment to achieve professional excellence

Journey to Cloud: Envisioning Your Solution

PAVAN KUMAR REDDY GUNDRASAM

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

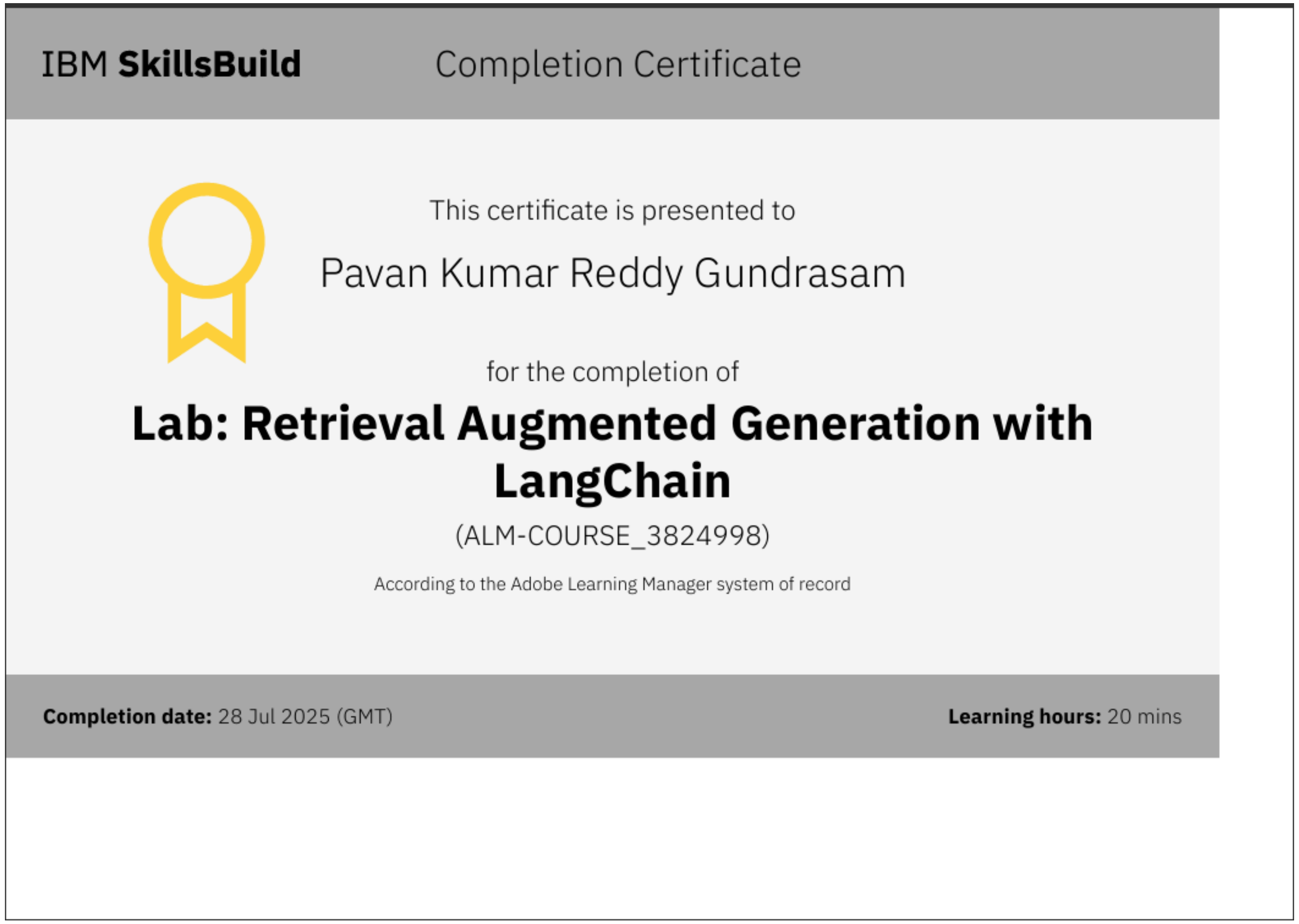
Issued on: Jul 21, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/26f76434-9428-42b1-adc9-f593d8379fb4

IBM

edunet foundation

IBM CERTIFICATIONS



IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Pavan Kumar Reddy Gundrasam

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 28 Jul 2025 (GMT) **Learning hours:** 20 mins

edunet foundation

THANK YOU

edunet
foundation