CAPSTONE PROJECT

Network Intrusion Detection Using Machine Learning

Presented By:
1. Nihal DR-CMR institute of technology-AI&DS

edunet foundation

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

- **The proposed system aims to address the challenge of predicting the class of network intrusion. This involves leveraging data analytics and machine learning techniques to forecast class patterns accurately. The solution will consist of the following components:**
- **Data Collection:**
  - The primary dataset for this project is the well-known **NSL-KDD dataset**, obtained from Kaggle. This is a refined and widely used benchmark dataset for intrusion detection research. It contains a large number of network connection records, each described by 41 features and labeled as either `normal` or a specific type of `anomaly` (attack).
- **Data Preprocessing:**
  - This is a critical step to prepare the data for the machine learning model. The process involves:
    1. **Handling Categorical Data:** Converting non-numeric features like `protocol_type`, `service`, and `flag` into a numerical format using **one-hot encoding**.
    2. **Feature Scaling:** Normalizing the numerical features using a technique like **StandardScaler**. This ensures that all features contribute equally to the model's learning process, preventing features with larger ranges from dominating the outcome.
- **Machine Learning Algorithm:**
  - We chose a **Binary Classification** algorithm because it offers high accuracy, is robust against overfitting, and can effectively handle the complexity of network data. It also helps us understand which network features are most important for detecting threats.
- **Deployment:**

  IBM cloud object storage for dataset handling

- **Evaluation:**
  The model's performance will be rigorously evaluated using a withheld test set. The key metrics for this classification problem include:
    - **Accuracy:** The overall percentage of correct predictions.
    - **Precision:** The accuracy of positive predictions

- Result:The expected result is a **highly accurate and reliable Network Intrusion Detection System (NIDS)**

SYSTEM APPROACH

The "System Approach" section outlines the overall strategy and methodology for developing and
implementing the power system fault detection and classification. Here's a suggested structure for
this section:

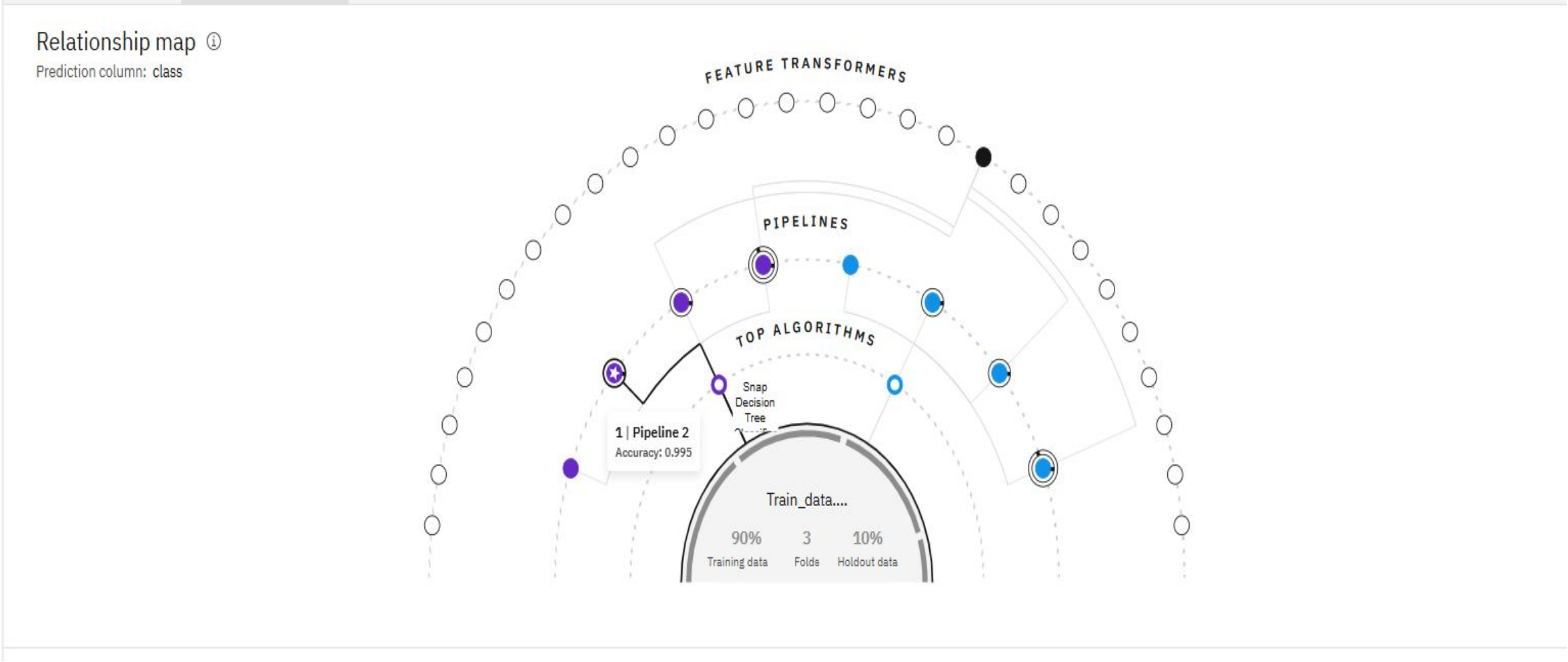
System requirements :

- IBM Cloud(mandatory)
- IBM Watson studio for model development and deployment
- IBM cloud object storage for dataset handling

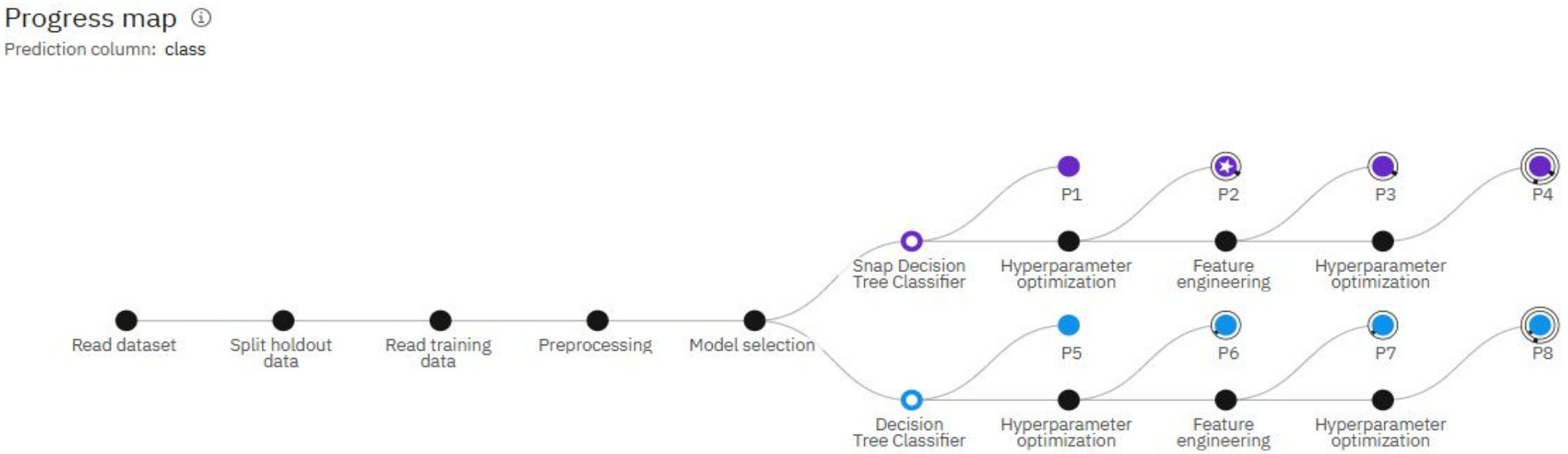
ALGORITHM & DEPLOYMENT

- In the Algorithm section, describe the machine learning algorithm chosen for predicting bike counts. Here's an example structure for this section:
- **Algorithm Selection:**
  - We chose a **Binary Classification** algorithm because it offers high accuracy, is robust against overfitting, and can effectively handle the complexity of network data. It also helps us understand which network features are most important for detecting threats.
- **Data Input:**
  - The model is trained on **41 features** from a network dataset, including connection duration, protocol type, and traffic statistics.
- **Training Process:**
  - The algorithm builds hundreds of decision trees using a labeled historical dataset (`normal` vs. `anomaly`). We use cross-validation and hyperparameter tuning to ensure the model is highly accurate and generalizes well to new data.
- **Prediction Process:**
  - The trained model receives real-time network data, processes it through its forest of trees, and instantly classifies the connection by "voting." This provides an immediate flag for malicious activity.

RESULT



RESULT



RESULT

Model evaluation measure

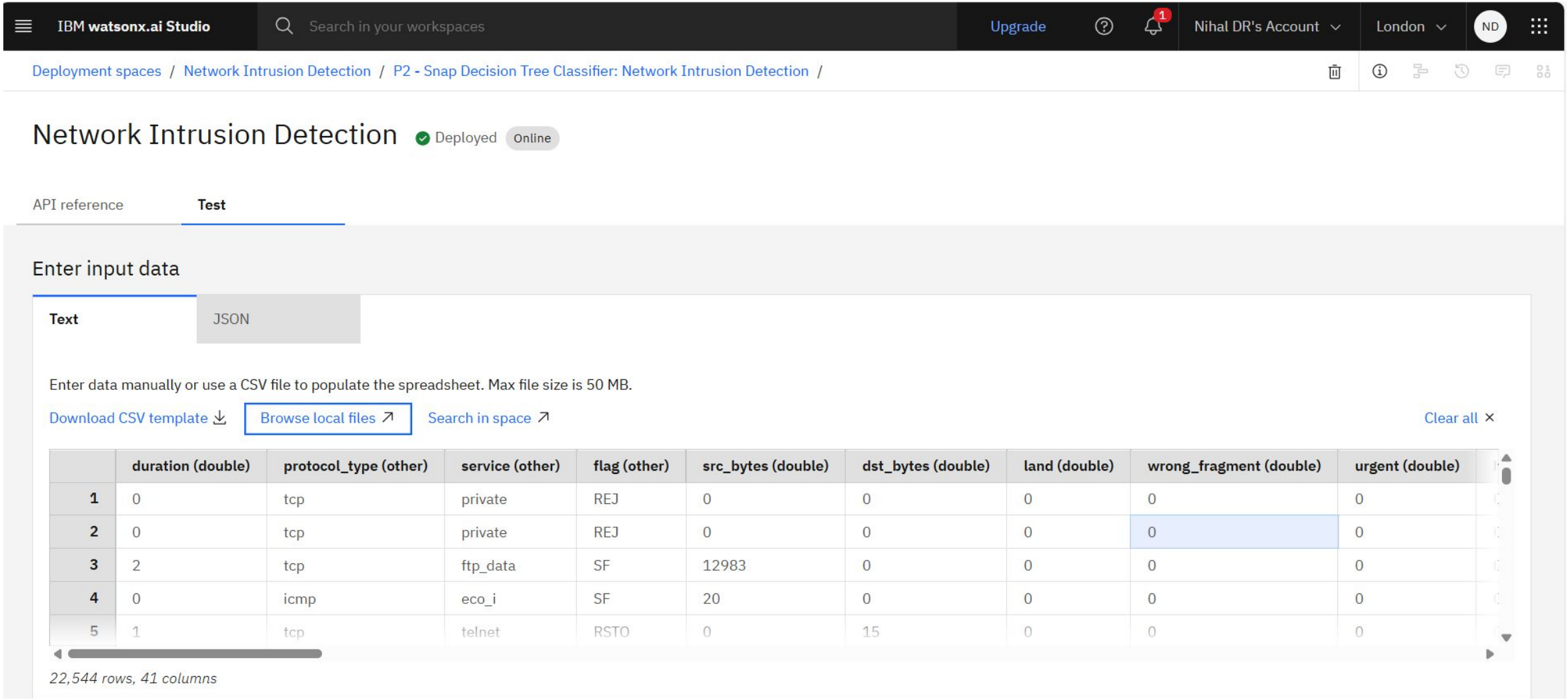
Measures	Holdout score	Cross validation score
Accuracy	0.998	0.995
Area under ROC	0.998	0.995
Precision	0.997	0.995
Recall	0.999	0.995
F1	0.998	0.995
Average precision	0.996	0.993
Log loss	0.086	0.196

Confusion matrix

Observed	Predicted		
	normal	anomaly	Percent correct
normal	1343	2	99.9%
anomaly	4	1171	99.7%
Percent correct	99.7%	99.8%	99.8%

Less correct — More correct

RESULT



	duration (double)	protocol_type (other)	service (other)	flag (other)	src_bytes (double)	dst_bytes (double)	land (double)	wrong_fragment (double)	urgent (double)
1	0	tcp	private	REJ	0	0	0	0	0
2	0	tcp	private	REJ	0	0	0	0	0
3	2	tcp	ftp_data	SF	12983	0	0	0	0
4	0	icmp	eco_i	SF	20	0	0	0	0
5	1	tcp	telnet	RSTO	0	15	0	0	0

RESULT

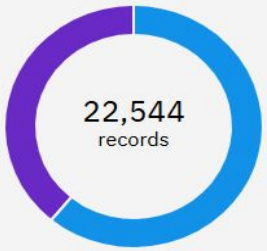
Prediction results

Close

Prediction type

Binary classification

Prediction percentage



anomaly normal

Confidence level distribution

Display format for prediction results

(•) Table view ( ) JSON view

Show input data

	Prediction	Confidence
1	anomaly	100%
2	anomaly	100%
3	normal	100%
4	anomaly	100%
5	normal	100%
6	normal	100%
7	normal	100%
8	normal	100%
9	normal	100%
10	anomaly	100%
11	anomaly	100%
12	normal	100%
13	anomaly	100%
14	anomaly	100%
15	normal	100%
16	normal	100%
17	normal	100%

Download JSON file

CONCLUSION

- In conclusion, this project successfully demonstrates the power of machine learning in building a robust Network Intrusion Detection System (NIDS). By leveraging a classic dataset and the capabilities of cloud-based tools like IBM Watson Studio, we were able to preprocess network traffic data, train a sophisticated classification model, and create a system capable of distinguishing between normal activity and various cyber-attacks such as DoS, Probe, R2L, and U2R. The resulting model not only provides an essential early warning system for malicious activities but also establishes a strong foundation for future enhancements. This work underscores the critical role of data-driven security solutions in protecting modern communication networks and opens avenues for more advanced implementations, including real-time automated threat response and the integration of deep learning for even greater accuracy and foresight in identifying novel cyber threats.

FUTURE SCOPE

- This project has significant future potential, evolving from a warning system into a proactive security solution. The most immediate advancement is creating a fully automated **Intrusion Prevention System (IPS)** that doesn't just detect threats but actively responds by blocking malicious IP addresses or quarantining suspicious connections in real-time. To counter new and unseen "zero-day" attacks, the system can be upgraded with more sophisticated **deep learning models**, like LSTMs, and unsupervised learning algorithms capable of identifying novel anomalies without prior training data. Furthermore, incorporating **Explainable AI (XAI)** would make the system's decisions transparent, showing security analysts precisely why an activity was flagged as malicious, thereby building trust and accelerating incident response. To handle the massive data volumes of modern networks, the system can be integrated with big data platforms like **Apache Spark** for scalable processing and deployed on **edge devices** for instant, on-site threat detection.

REFERENCES

- The dataset to be audited was provided which consists of a wide variety of intrusions simulated in a military network environment. It created an environment to acquire raw TCP/IP dump data for a network by simulating a typical US Air Force LAN. The LAN was focused like a real environment and blasted with multiple attacks. Also, each connection is labelled as either normal or as an attack with exactly one specific attack type. Each connection record consists of about 100 bytes.For each TCP/IP connection, 41 quantitative and qualitative features are obtained from normal and attack data (3 qualitative and 38 quantitative features) .The class variable has two categories:
  • Normal
  • Anomalous
  Dataset Link:
  Network Intrusion Detection
  Public deployment Link:
  https://eu-gb.ml.cloud.ibm.com/ml/v4/deployments/nihal_77/predictions?version=2021-05-01
  Github deployment Link:
  https://github.com/NihalDR/IBM_Finalproject

IBM CERTIFICATIONS

IBM CERTIFICATIONS

edunet foundation

IBM CERTIFICATIONS

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Nihal DR

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 18 Jul 2025 (GMT)

Learning hours: 20 mins

edunet foundation

THANK YOU

edunet
foundation