

Machine learning project

Network Intrusion Detection system

Problem statement No.40 – Network Intrusion Detection

The Challenge:

Create a robust network intrusion detection system (NIDS) using machine learning. The system should be capable of analysing 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.

About Dataset:

Background

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. A connection is a sequence of TCP packets starting and ending at some time duration between which data flows to and from a source IP address to a target IP address under some well-defined protocol. 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

[Kaggle dataset link](#) - Used in this representation – this will generate a model that only give prediction of class(normal or anomaly)

Technology Used - IBM cloud lite services.

[GitHub repo link](#)

Upgraded NIDS Model – Attack Type Classification Rules

For a upgraded model ,traditional binary classification (Normal vs Anomaly) of kaggle dataset is updated to `DoS` ,`Probe` ,`R2L` ,`U2R` , or `Normal` for more specific analysis on type of attack.

These categories are based on theoretical behavior patterns and implemented using rule-based logic in code using python:

```
import pandas as pd
import numpy as np
import random
from collections import defaultdict

# Define theoretical features for each attack type
```

```

attack_profiles = {
    "DoS": {
        "src_bytes": (0, 500), # Low volume sent
        "dst_bytes": (1000, 10000), # High volume received
        "count": (50, 100), # High connection count
        "serror_rate": (0.8, 1.0), # High error rate
        "same_srv_rate": (0.9, 1.0),
        "diff_srv_rate": (0.0, 0.2)
    },
    "Probe": {
        "src_bytes": (0, 200),
        "dst_bytes": (0, 500),
        "count": (20, 60),
        "serror_rate": (0.0, 0.2),
        "same_srv_rate": (0.2, 0.6),
        "diff_srv_rate": (0.4, 0.9)
    },
    "R2L": {
        "src_bytes": (0, 100),
        "dst_bytes": (0, 100),
        "count": (1, 5),
        "serror_rate": (0.0, 0.1),
        "num_failed_logins": (2, 5),
        "logged_in": (0, 0),
        "is_guest_login": (1, 1)
    },
    "U2R": {
        "src_bytes": (0, 50),
        "dst_bytes": (0, 50),
        "count": (1, 3),
        "serror_rate": (0.0, 0.1),
        "num_file_creations": (1, 5),
        "root_shell": (1, 1),
        "su_attempted": (1, 1)
    },
    "Normal": {
        "src_bytes": (100, 10000),
        "dst_bytes": (100, 10000),
        "count": (1, 10),
        "serror_rate": (0.0, 0.1),
        "same_srv_rate": (0.7, 1.0),
        "diff_srv_rate": (0.0, 0.3)
    }
}

# Common fields across all rows
common_fields = {
    "duration": (0, 60),

```

```

    "protocol_type": ["tcp", "udp", "icmp"],
    "service": ["http", "ftp", "smtp", "dns", "ssh"],
    "flag": ["SF", "REJ", "S0", "RSTR"],
}

# Map attack type to class (binary)
attack_class = {
    "DoS": "anomaly",
    "Probe": "anomaly",
    "R2L": "anomaly",
    "U2R": "anomaly",
    "Normal": "normal"
}

# Function to score how well a row matches a profile
def match_score(row, profile):
    score = 0
    for feature, (low, high) in profile.items():
        if feature in row:
            try:
                val = float(row[feature])
                if low <= val <= high:
                    score += 1
            except:
                continue
    return score

# Assign attack_type and class to each row
def assign_attack_type(row):
    best_attack = None
    best_score = -1
    for attack, profile in attack_profiles.items():
        score = match_score(row, profile)
        if score > best_score:
            best_score = score
            best_attack = attack
    return best_attack

# Generate a dataset
def generate_theoretical_attack_dataset(num_samples=1000):
    data = []
    attack_types = list(attack_profiles.keys())
    samples_per_attack = num_samples // len(attack_types)

    for attack in attack_types:
        for _ in range(samples_per_attack):
            row = {}

```

```

# Add common fields
row["duration"] = random.randint(*common_fields["duration"])
row["protocol_type"] =
random.choice(common_fields["protocol_type"])
row["service"] = random.choice(common_fields["service"])
row["flag"] = random.choice(common_fields["flag"])

# Add features for the current attack type
for feature, (low, high) in attack_profiles[attack].items():
    row[feature] = round(random.uniform(low, high), 2)

# Fill missing features with 0 if not defined for this attack type
all_possible_features = set(f for d in attack_profiles.values()
for f in d)
for feature in all_possible_features:
    if feature not in row:
        row[feature] = 0

# Add labels
row["attack_type"] = attack
row["class"] = attack_class[attack]

data.append(row)

df = pd.DataFrame(data)
return df

# Generate and show the synthetic dataset
synthetic_df = generate_theoretical_attack_dataset(1000)

# Load your uploaded dataset
input_path = "Train_data.csv"
df = pd.read_csv(input_path)

df['attack_type'] = df.apply(assign_attack_type, axis=1)
df['class'] = df['attack_type'].map(attack_class)

# Save the dataset for export if needed
output_path = "Train_data_with_attack_types_enhanced.csv"
df.to_csv(output_path, index=False)

print(f"Saved labeled dataset to {output_path}")

```

Attack Type Breakdown

Attack Type	Full Form	Description
DoS	Denial of Service	Overwhelms network to disrupt service availability
Probe	Probing/Reconnaissance	Gathers network/system info through scanning
R2L	Remote to Local	External attacker tries to gain unauthorized access
U2R	User to Root	Authorized user tries to escalate to root privileges
Normal	—	Legitimate connection with no signs of intrusion

Rule-Based Classification Logic:

Below are the key behavioral indicators (features) and red flags used to classify each attack type:

1. DoS (Denial of Service)

- Large volume of traffic or repeated bursts
- Targets same service/host continuously
- Symptoms of SYN flood or similar volumetric attacks

Key Indicators:

- `count` ≥ 50
- `serror_rate` ≥ 0.8
- `diff_srv_rate` ≤ 0.2

2. Probe

- Scans across many services/host
- Low payload but high variety
- Attempting to map open ports/services

Key Indicators:

- count in $[20, 60]$
- $diff_srv_rate \geq 0.4$
- $serror_rate \leq 0.2$

3. R2L (Remote to Local)

- Suspicious or repeated login failures
- Attempts by external hosts to gain internal access

Key Indicators:

- num_failed_logins > 1
- logged_in = 0
- is_guest_login = 1

4. U2R (User to Root)

- Subtle privilege escalation attempts
- Presence of root access shells, su attempts, or file manipulations

Key Indicators:

- root_shell = 1
- su_attempted = 1
- num_file_creations > 0

Normal Traffic

If none of the red flag rules above apply, the traffic is assumed to be Normal.

Implementation Notes

- All records are scored using a `match_score()` function.
- The highest-scoring attack type is selected as the predicted label.
- The label is stored in a new column: `attack_type`
- The original binary class is mapped as:

anomaly → DoS, Probe, R2L, U2R

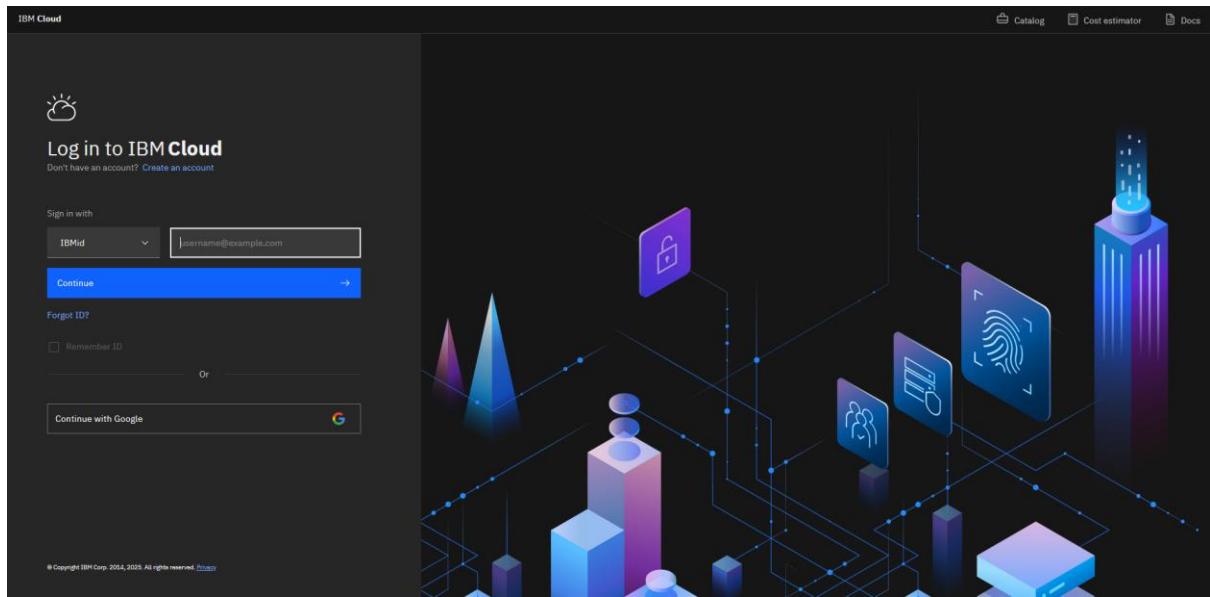
normal → Normal

Usage

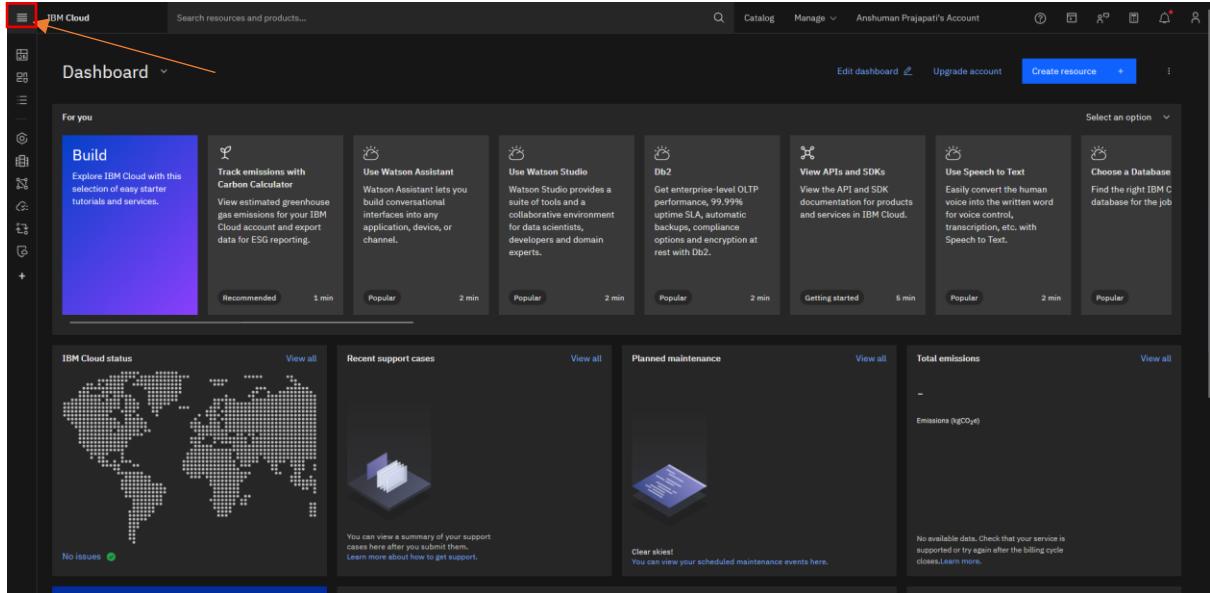
- Can be used as training data for supervised multi-class classification models (e.g., Random Forest, XGBoost).
- Compatible with IBM Watsonx.ai AutoAI for model generation and deployment.
- Provides interpretable security labels for SIEM or firewall logs.

DATA made using python and Kaggle dataset sets which can be used for upgraded model:

-[New data set for training of model](#)



1. Login to [IBM cloud](#) with your credentials.



2. This is homepage of IBM cloud now press on top left corner of this page.

The screenshot shows the IBM Cloud dashboard with a sidebar on the left containing various service categories like Dashboard, Projects, Resource List, Containers, Databases, Infrastructure, Observability, Platform Automation, Security, API Management, Cloud Pak for Data, Partner Center, SAP, Satellite, vm VMware, and Watsonx. A red arrow points to the 'Resource List' item in the sidebar. The main content area displays several cards: 'Track emissions with Carbon Calculator' (Recommended, 1 min), 'Use Watson Assistant' (Popular, 2 min), 'Use Watson Studio' (Popular, 2 min), 'Db2' (Popular, 2 min), 'API Connect' (Recommended, 2 min), 'Create a project' (Popular, 5 min), 'IBM Watson Machine Learning' (Popular, 5 min). Below these cards are sections for 'Recent support cases', 'Planned maintenance', and 'Total emissions'. A tooltip for the 'Screenshot copied to clipboard' button is visible.

3. Press on Resource list

The screenshot shows the 'Resource list' page. On the left, there is a sidebar with a tree view of resource categories: Compute (0), Containers (0), Networking (0), Storage (1+), Converged Infrastructure (0), Enterprise applications (0), AI / Machine Learning (2+), Analytics (0), Blockchain (0), Databases (0), Developer tools (0), Observability (0), Migration (0), Integration (0+), Internet of Things (0), Security (0), Mobile (0), and Other (0+). The main area is a table with columns: Name, Group, Location, Product, Status, and Tags. There are filter bars at the top of each column. A blue 'Create resource' button is located in the top right corner.

4. In this page if any resource is in use delete it and make it free.

The screenshot shows the IBM Cloud Resource list interface. On the left, there's a sidebar with various service categories like Compute, Containers, Networking, and Storage. Under the Storage category, there's a single item: 'Cloud Object Storage-cl'. This item is highlighted with a blue border. To its right, the details panel shows the product as 'Cloud Object Storage', status as 'active', and a 'Delete' button at the bottom.

This screenshot shows the same IBM Cloud Resource list interface, but with a modal dialog box overlaid. The dialog is titled 'Delete resource' and contains a warning message: 'Deleting the resource will remove it from all connected apps and permanently delete any associated data and service credentials. Are you sure that you want to delete the "Cloud Object Storage-cl" service?'. Below the message, there's a text input field with 'Cloud Object Storage-cl' typed into it, followed by a note: 'Type "Cloud Object Storage-cl" to confirm'. At the bottom of the dialog are two buttons: 'Cancel' and 'Delete', with 'Delete' being the primary action button.

5. Repeat this for all the occupied resources and make sure no resources are in use.

The screenshot shows the IBM Cloud Resource list interface. At the top, there is a search bar labeled "Search resources and products...". Below the search bar is a navigation bar with links for Catalog, Manage, and Anshuman Prajapati's Account. On the far right of the navigation bar are several small icons. The main area is titled "Resource list" and contains a table with columns: Name, Group, Location, Product, Status, and Tags. The "Name" column has a search input field. The "Group" column has a dropdown menu. The "Location" column has a dropdown menu. The "Product" column has a search input field. The "Status" column has a search input field. The "Tags" column has a dropdown menu. To the left of the table is a sidebar with a tree view of resource categories. Categories include Compute, Containers, Networking, Storage, Converged infrastructure, Enterprise applications, AI / Machine Learning, Analytics, Blockchain, Databases, Developer tools, Observability, Migration, Integration, Internet of Things, Security, Mobile, and Other. Each category has a count of resources in parentheses.

The screenshot shows the IBM Cloud Resource list interface after performing a search for "watsonx". The search results are displayed in a modal window titled "Catalog Results". The results list includes: "watsonx" Service, "Watsonx.ai SaaS with Assistant and Governance" Software, "Cloud automation for watsonx.ai" Software, "watsonx.ai Studio" Service, and "watsonx.data intelligence" Service. Below the results, there are two links: "Search 'watsonx' in Support Cases" and "Search 'watsonx' in Docs". The rest of the interface is identical to the first screenshot, showing the sidebar with resource categories and the main table for managing resources.

6. Now click search for watsonx.ai Studio in search bar and press enter.

Watsonx.ai Studio
 (Formerly known as Watson Studio) Develop powerful AI solutions with an integrated collaborative studio and industry-standard APIs and SDKs.

Create **About**

Type: Service
 Provider: IBM
 Last updated: 05/06/2025
 Category: AI / Machine Learning
 Compliance: HIPAA Enabled, IAM-enabled
 Location: Sydney (au-syd), Frankfurt (eu-de), London (eu-gb), Tokyo (ap-tok), Dallas (us-south), Toronto (ca-tor)
 Related links: Docs, Terms

Select a location: Sydney (au-syd)

Select a pricing plan
 Prices shown are for country or location: United States

Plan	Features and capabilities	Pricing
Lite	5 authorized users 50 capacity unit-hours monthly limit Environment = # of capacity units required per hour • 1 vCPU + 4 GB RAM = 0.5 • 2 vCPU + 8 GB RAM = 1 • 4 vCPU + 16 GB RAM = 2 • Decision Optimization + Watson NLP + Environment + 6 • Synthetic Data Generator, 2 vCPU + 8 GB RAM = 7 (requires Watsonx Runtime)	Free
Professional	Unlimited collaborator Unlimited elastic compute environments Environment = # of capacity units required per hour • 1 vCPU + 4 GB RAM = 0.5 • 2 vCPU + 8 GB RAM = 1 • 4 vCPU + 16 GB RAM = 2	\$1.02 USD/Capacity Unit-Hour

The Lite plan offers most Watsonx.ai Studio data science and AI features with usage restrictions.
 Lite plan services are deleted after 30 days of inactivity.

I have read and agree to the following license agreements:
 Terms

Create **Add to estimate**

7. Now create with lite plan and and make sure to check on license agreements.

watsonx.ai Studio-07 ● Add tags ●

Manage

Plan

watsonx.ai Studio in Cloud Pak for Data and watsonx

Build and deploy machine learning models on either platform. Work with foundation models on Watsonx as a Service.

Launch in

Helpful links

- Documentation**: Learn about tools, features, and how to perform a wide variety of Data and AI tasks. [Cloud Pak for Data →](#) [watsonx →](#)
- Learning path**: Start a step-by-step tutorial to get up and running quickly. [Cloud Pak for Data →](#) [watsonx →](#)
- Videos**: Watch videos to learn about Watsonx Studio. [Cloud Pak for Data →](#) [watsonx →](#)

How to use Watsonx Studio

IBM Watsonx Studio in Cloud Pak for Data and Watsonx
IBM Cloud Pak for Data, Watsonx Unifying platforms
IBM Cloud Base cloud infrastructure

IBM Watsonx Studio is part of IBM Cloud Pak for Data and Watsonx, and serves as the AI capability of the data fabric architecture.

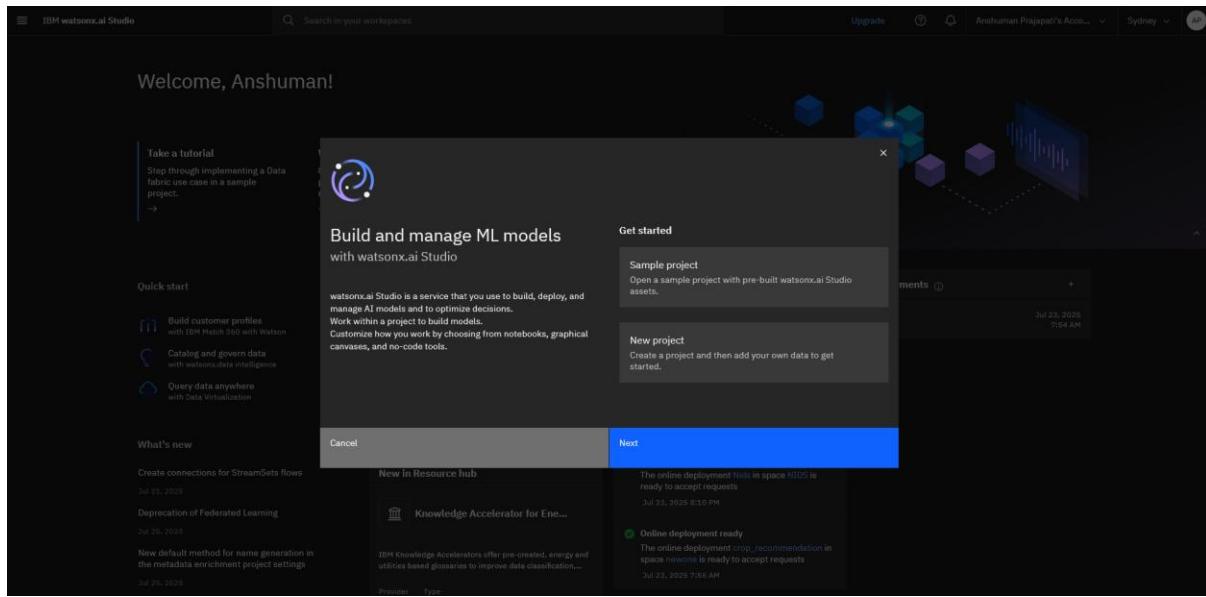
8. Now click on launch in.

The screenshot shows the IBM Watsonx.ai Studio interface. At the top, it says "Welcome, Anshuman!". On the left, there's a sidebar with "Take a tutorial", "Quick start" (listing "Build customer profiles with IBM Match 360 with Watson", "Catalog and govern data with Watsonx.ai Intelligence", "Build and manage ML models with Watsonx.ai Studio", and "Query data anywhere with Data Virtualization"), and "What's new" (listing "Create connections for StreamSets flows" (Jul 21, 2025), "Deprecation of Federated Learning" (Jul 25, 2025), and "New default method for name generation in the manifest/enrichment project settings"). The main area has a heading "Build and manage ML models with Watsonx.ai Studio". It includes a "Get started" section with a callout "Provision Watsonx.ai Runtime" (Create an instance of Watsonx.ai Runtime from the service catalog). Below this is a "Next" button. To the right, there's a "Summary" section for a "Watsonx.ai Runtime" instance, showing details like Region: Sydney, Plan: Lite, Service name: watsonx.ai Runtime-vm, and Resource group: Default. There are also "Create", "View terms", and "Cancel" buttons.

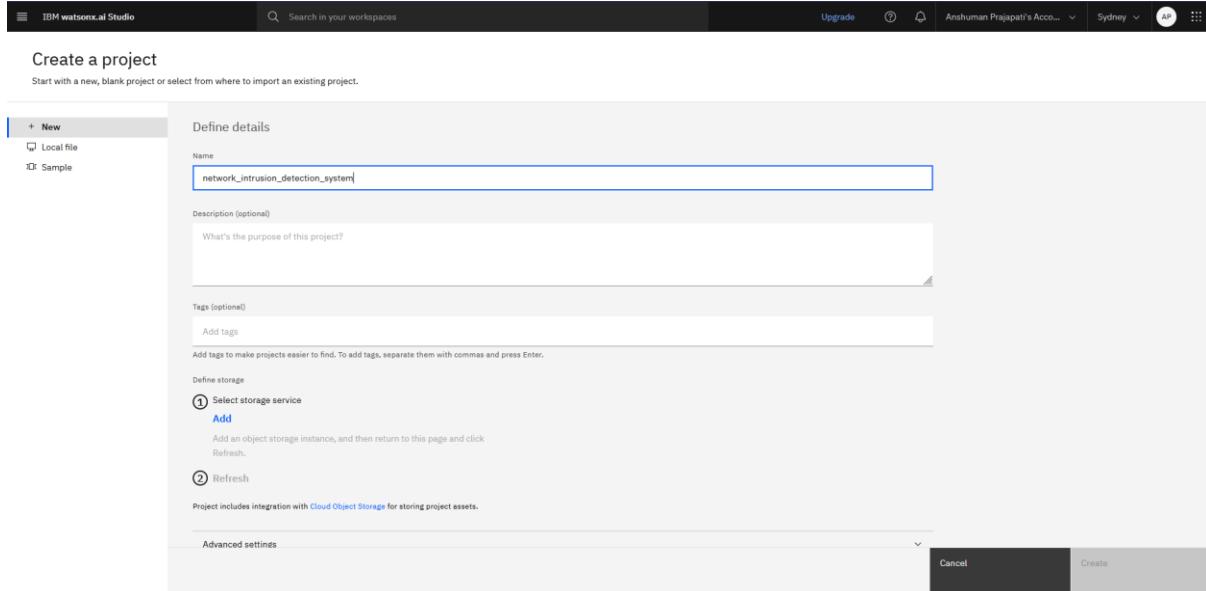
9. Now select provision watsonx.ai Runtime and press next.

The screenshot shows the "watsonx.ai Runtime" provisioning interface. It starts with a "Select a region" step where "Sydney" is chosen. Then it moves to a "Pricing plan" step, noting that prices do not include tax and are for the United States. It lists a "Lite" plan with features like "Service instance", "Instance includes: 20 capacity unit-hours (CUH) per month, 50,000 tokens/data points per month, 100 pages per month", and "Foundation models: Inferring for text generation consumes tokens (as Resource Units)", "Text extraction: Each document page or image file or .tiff frame is considered 1 Page", and "Machine learning training tools: Compute usage counted as CUH, CUH rate based on training tool, hardware specification, and runtime environment". Finally, it reaches a "Summary" step for the "watsonx.ai Runtime" instance, which is already provisioned with the details: Region: Sydney, Plan: Lite, Service name: watsonx.ai Runtime-vm, and Resource group: Default. The "Create" button is prominently displayed at the bottom of the summary panel.

10. Press create.



11. Select new project and press next.



12. Give the project a name.

Create a project

Start with a new, blank project or select from where to import an existing project.

+ New
Local file
Sample

Define details

Name: network_intrusion_detection_system

Description (optional): What's the purpose of this project?

Tags (optional): Add tags

① Select storage service
Add
Add an object storage instance, and then return to this page and click Refresh.
② Refresh

Project includes integration with Cloud Object Storage for storing project assets.

Advanced settings

Cancel Create

13. Now add storage to it.

IBM WatsonX AI Studio Search in your workspaces

Services catalog /

Cloud Object Storage

Author: IBM • Date of last update: Apr 15, 2025 • Docs • API Docs

Create About

Pricing plan
Displayed prices do not include tax. Monthly prices shown are for country or region: United States

Plan	Features	Pricing
One-Rate	One-Rate Plan is a Pay-as-You-Go option with a single, flat monthly rate (\$/GB) that includes storage, API operations, retrieval, and outbound bandwidth—making it ideal for high-activity workloads with frequent access and data transfer, such as analytics, media, and web apps. The plan includes built-in allowances that scale with stored capacity and offers automatic volume discounts as usage grows.	
Lite(deprecated)	Lite plan instance is free to use for Storage capacity up to 25 GB per month. Lite plan instance is used for trial, and can be easily upgraded to Standard plan for unlimited scalability and full functionality. None Lite plan services are deleted after 30 days of inactivity.	Free
Standard	Standard Plan is a flexible Pay-as-You-Go option with no minimum fee—ideal for workloads with large storage needs but low or infrequent access and outbound traffic. It includes a Free Tier with 5GB of Smart Tier storage for 12 months. Charges are based on actual usage, with separate billing for storage, outbound bandwidth, API operations, and data retrieval. Multiple storage classes help you optimize costs based on how often data is accessed. Free Tier allowance: Storage up to 5GB/month Up to 2000 Class A requests/month Up to 20,000 Class B requests/month Up to 10GB/month of data retrieval Up to 5GB/month of egress Applies to aggregate total across all smart tier buckets in your account	

Summary

Cloud Object Storage
Region: Global
Plan: Lite(deprecated)
Service name: Cloud Object Storage-zg
Resource group: Default

Create View terms Cancel

14. Create a object storge in lite plan.

IBM Watsonx.ai Studio Search in your workspaces Upgrade ? Anshuman Prajapati's Acco... Sydney AP ::

Create a project

Start with a new, blank project or select from where to import an existing project.

+ New Local file Sample

Define details

Name network_intrusion_detection_system

Description (optional)

What's the purpose of this project?

Tags (optional)

Add tags Add tags to make projects easier to find. To add tags, separate them with commas and press Enter.

Define storage

① Select storage service Add Add an object storage instance, and then return to this page and click Refresh.

② Refresh Project includes integration with Cloud Object Storage for storing project assets.

Advanced settings

Cancel Create

15. Press refresh to show up the storage.

IBM Watsonx.ai Studio Search in your workspaces Upgrade ? Anshuman Prajapati's Acco... Sydney AP ::

Create a project

Start with a new, blank project or select from where to import an existing project.

+ New Local file Sample

Define details

Name network_intrusion_detection_system

Description (optional)

What's the purpose of this project?

Tags (optional)

Add tags Add tags to make projects easier to find. To add tags, separate them with commas and press Enter.

Storage

Cloud Object Storage-zg Project includes integration with Cloud Object Storage for storing project assets.

Advanced settings

Cancel Create

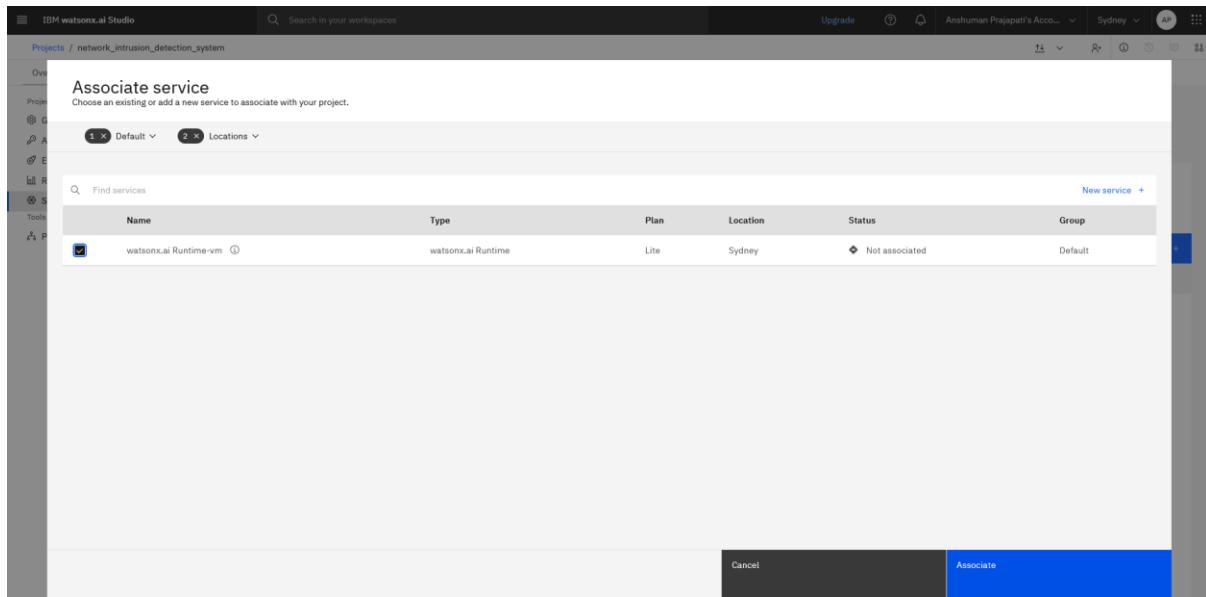
16. Press create.

The screenshot shows the 'Manage' tab of the IBM WatsonX AI Studio interface. On the left, a sidebar lists project categories: General, Access control, Environments, Resource usage, Services & integrations (which is selected and highlighted in blue), and Pipeline. The main content area is divided into several sections: 'General' (with fields for Name, Description, and Tags), 'Storage' (showing 0 Bytes used in a bucket named 'networkintrusionsdetectionsystem-donotdelete-pr-cdqatfu0phjei3'), and 'Controls' (with options for Opt-in to folders, Cloud Pak for Data platform, and Restricted collaborator eligibility). A 'Project ID' field contains the value '2e2d7faa-926b-4d86-9224-5c6ffbbf963f'. At the bottom right of the main area is a 'Manage in IBM Cloud' button.

17. Now on manage tab go to services and integration.

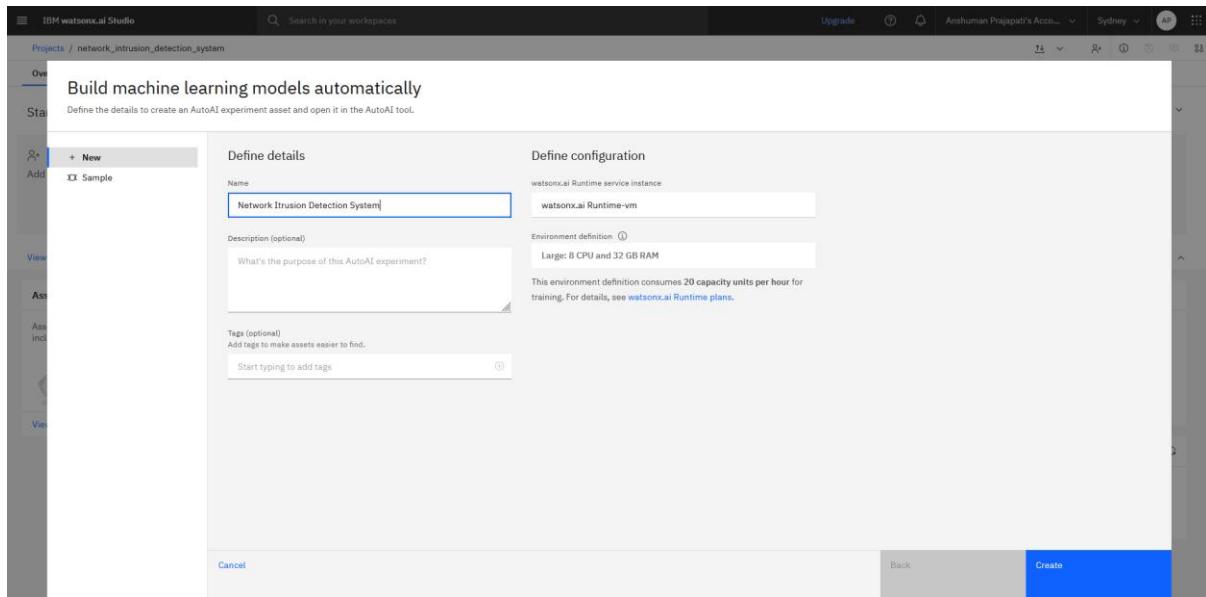
The screenshot shows the 'Services & integrations' tab of the IBM WatsonX AI Studio interface. The sidebar still shows the 'Services & integrations' category as selected. The main area displays the 'IBM services' section, which includes a note about associating IBM Cloud services with the project, a search bar for 'Find services', and a 'Service type' filter. A prominent blue button at the top right says 'Associate service'. Below the search bar, it says 'No services' and 'Click Associate service or ask a project Admin to associate one'. There is also a small icon of a white cube.

18. Now click on associate service.

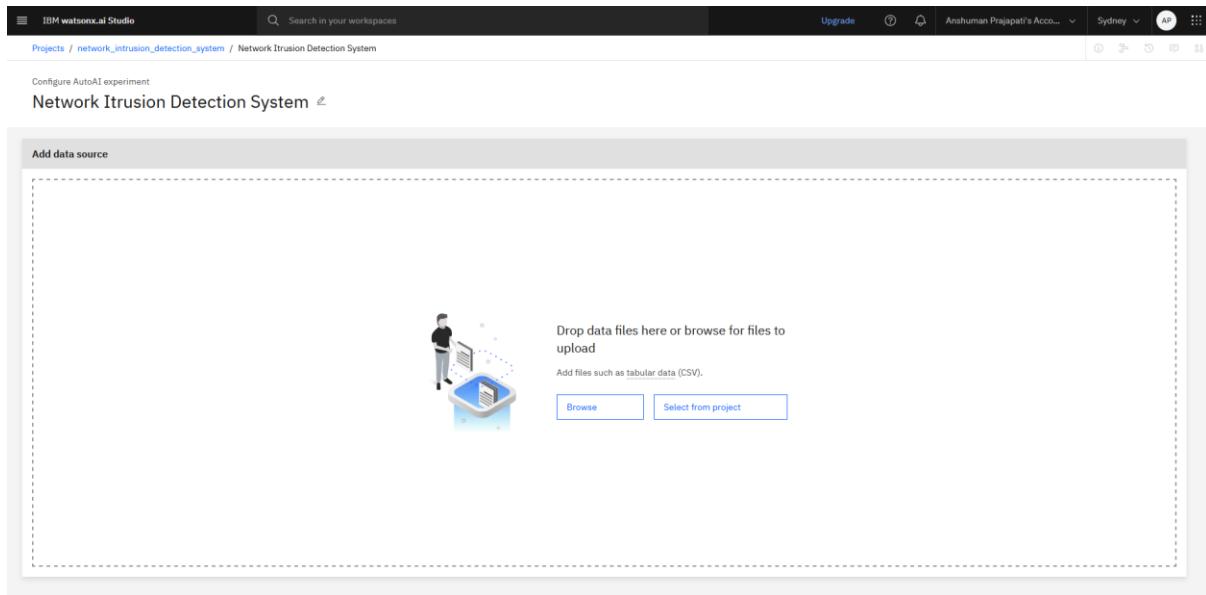


19. Check on watsonx.ai Runtime service and associate it.

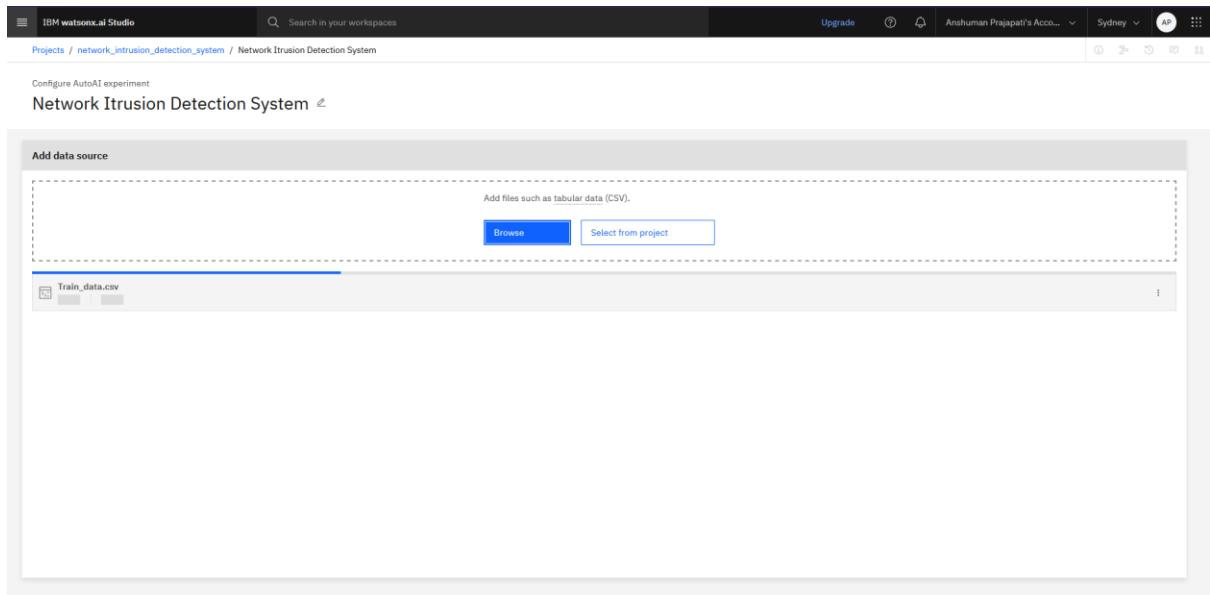
20. Now comeback to overview tab and select ***Build machine learning models automatically***



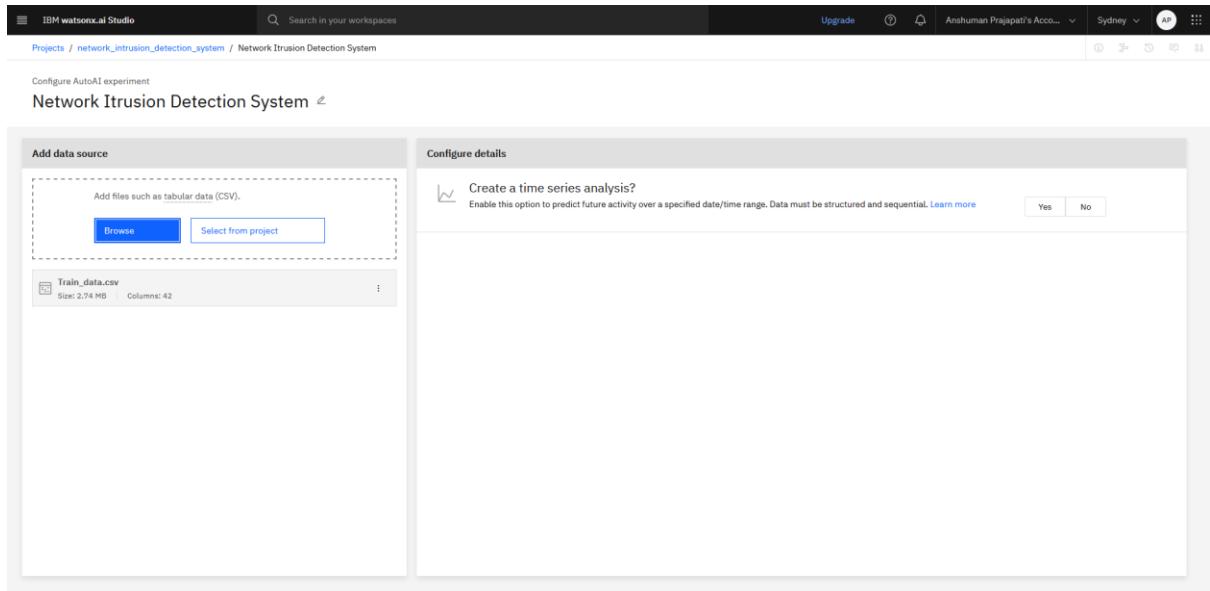
21. Give the experiment a name and click on create.



22. Now browse the file to upload (i.e. the .csv file to train the model).



23. Wait for the file to upload.

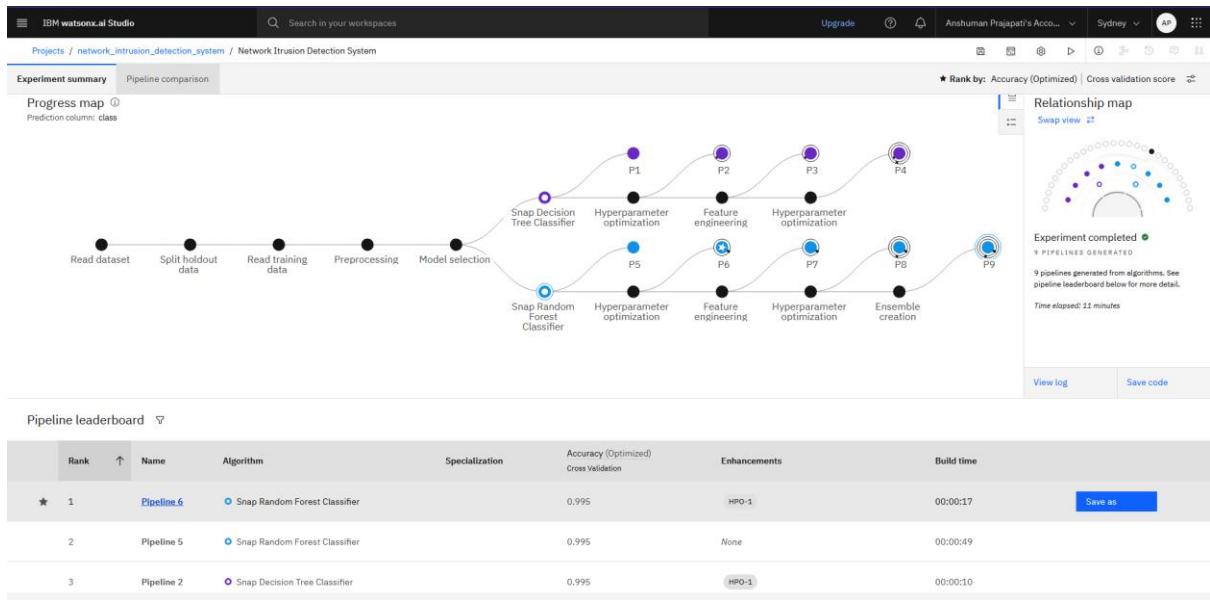


24. Now after the file gets completely uploaded it will ask for time series analysis select **no**.

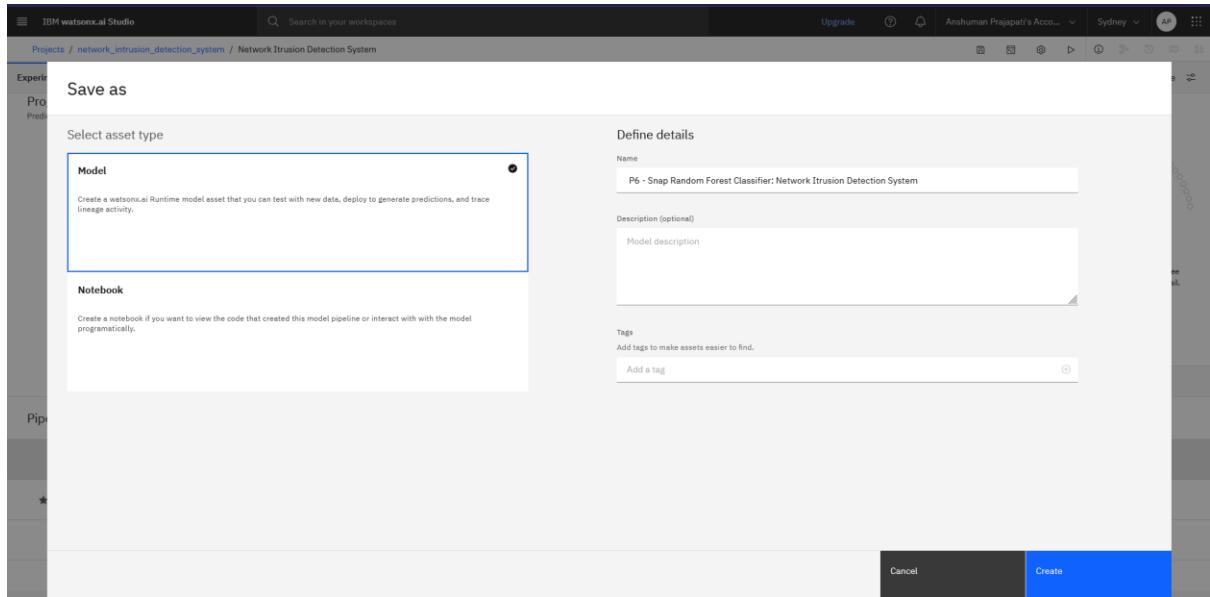
25. Select the prediction type (in this case its class) and make sure the prediction type is multiclass classification(as prediction column contains multiple distinct categories) and run experiment.

Rank	Name	Algorithm	Accuracy (Optimized) Cross Validation	Enhancements	Build time

26. It will take some time to perform the experiment.



27. Now after the experiment is completed we can see the best performing algorithm is Pipeline 6 with enhancement of hyperparameter optimization now click on save as.



28. Press on create.

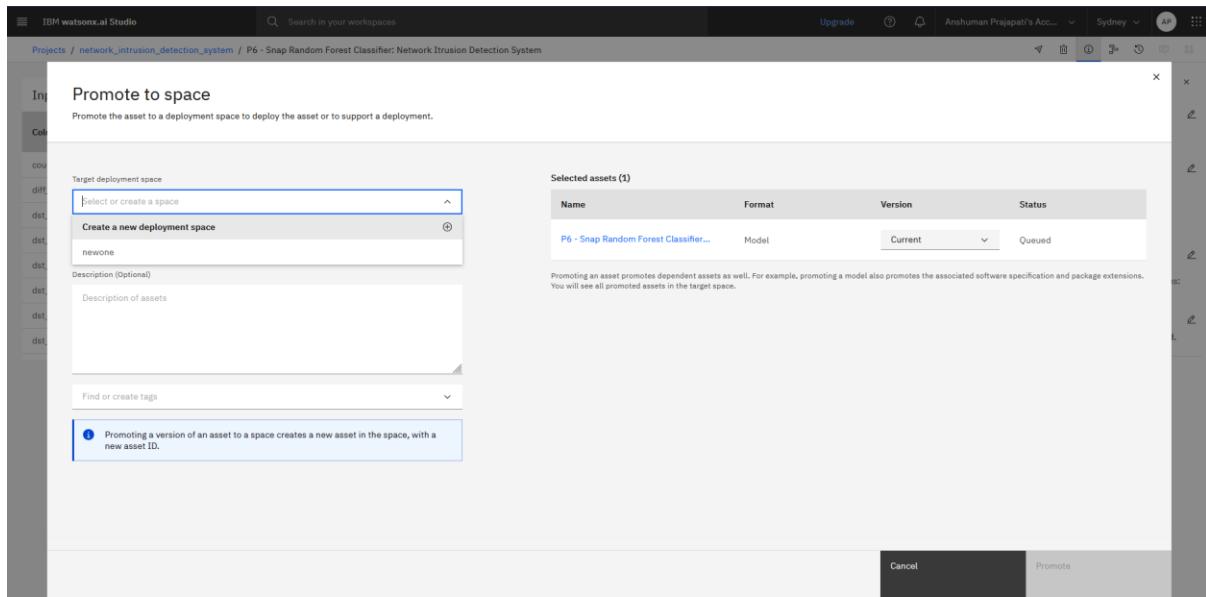
The screenshot shows the IBM Watsonx.ai Studio interface. At the top, there's a navigation bar with 'IBM Watsonx.ai Studio', a search bar, and user information. Below it is a 'Pipeline comparison' section with tabs for 'Experiment summary' and 'Pipeline comparison'. A 'Progress map' is displayed, showing a sequence of nodes: 'Read dataset', 'Split holdout data', 'Read training data', 'Preprocessing', 'Model selection', 'Snap Decision Tree Classifier', 'Hyperparameter optimization' (labeled P1), 'Feature engineering', 'Hyperparameter optimization' (labeled P2), 'Hyperparameter optimization' (labeled P3), 'Ensemble creation', and 'Snap Random Forest Classifier' (labeled P5). A red arrow points from the 'Experiment completed' message to a 'view in project' button.

Rank	Name	Algorithm	Specialization	Accuracy (Optimized) Cross Validation	Enhancements	Build time
★ 1	Pipeline 6	Snap Random Forest Classifier		0.995	HPO-1	00:00:17
2	Pipeline 5	Snap Random Forest Classifier		0.995	None	00:00:49
3	Pipeline 2	Snap Decision Tree Classifier		0.995	HPO-1	00:00:10

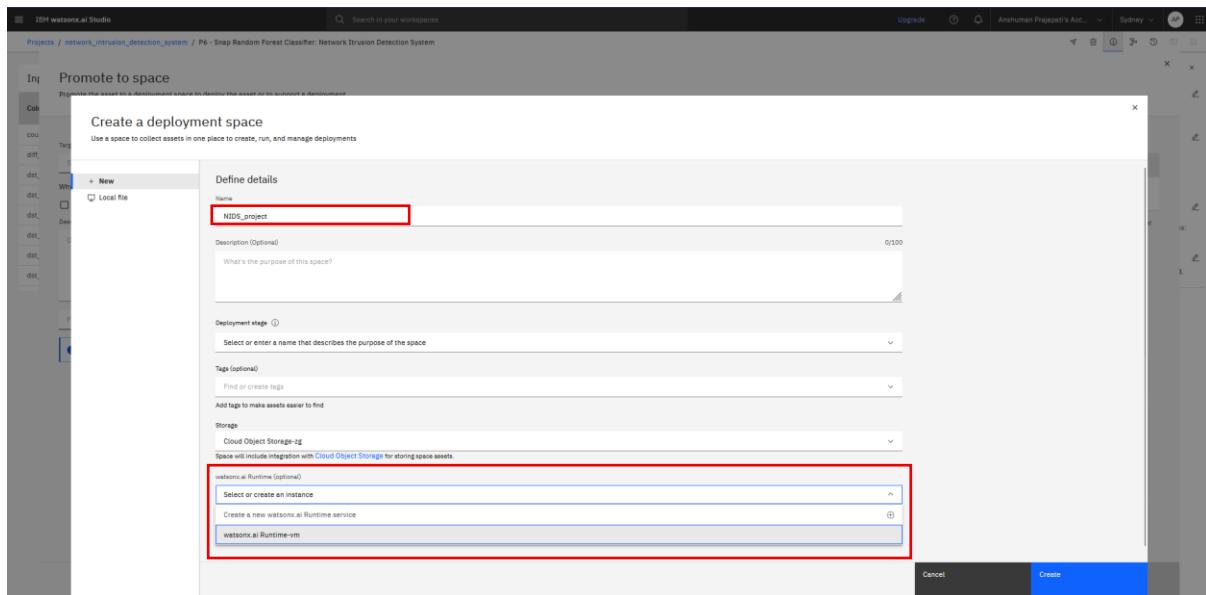
29. After the model is saved click on view in project.

This screenshot shows the 'Input (1)' table and a 'Promote to space' dialog. The table lists columns: 'count', 'diff_srv_rate', 'dst_bytes', 'dst_host_count', 'dst_host_diff_srv_rate', 'dst_host_error_rate', 'dst_host_same_src_port_rate', and 'dst_host_same_src_rate', all of type 'double'. To the right, the 'Promote to space' dialog is open, showing asset details: Name (P6 - Snap Random Forest Classifier: Network Intrusion Detection System), Description (No description provided), Asset Details (Type: wml-hybrid_0.1, Model ID: c764af36-892d-4c...), Software specification: hybrid_0.1, Hybrid pipeline software specifications: autoai-kb_r24.1-py3.11, Tags (Add tags to make assets easier to find), Last modified (1 minute ago by Service), and Created on (Aug 3, 2025 by Anshuman Prajapati).

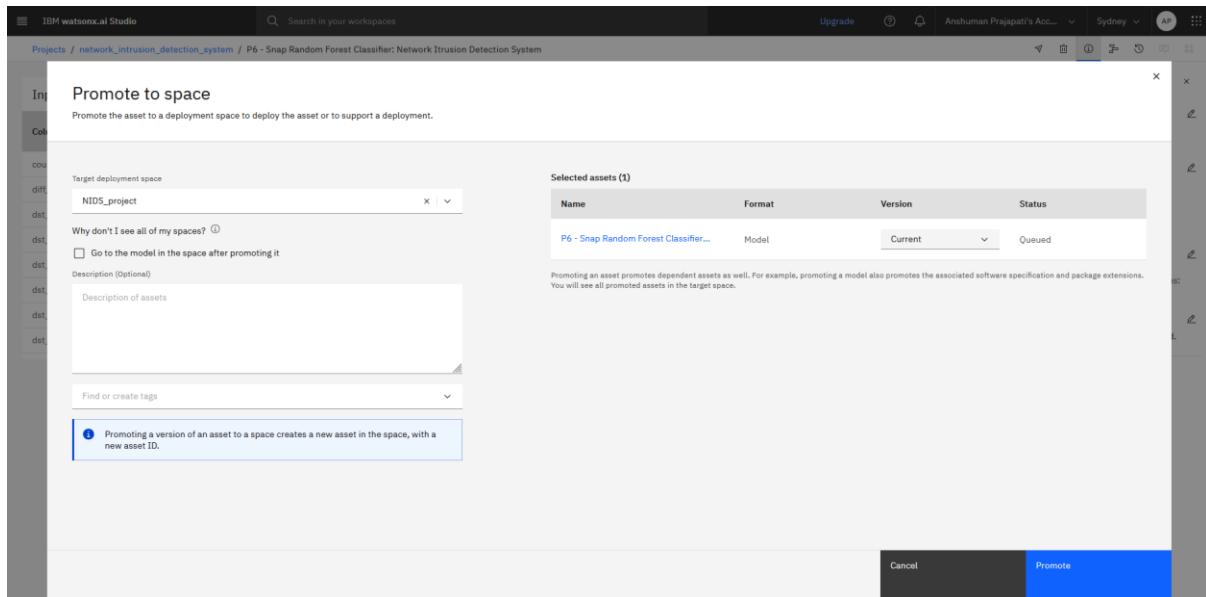
30. Now on the top towards right press on promote to space.



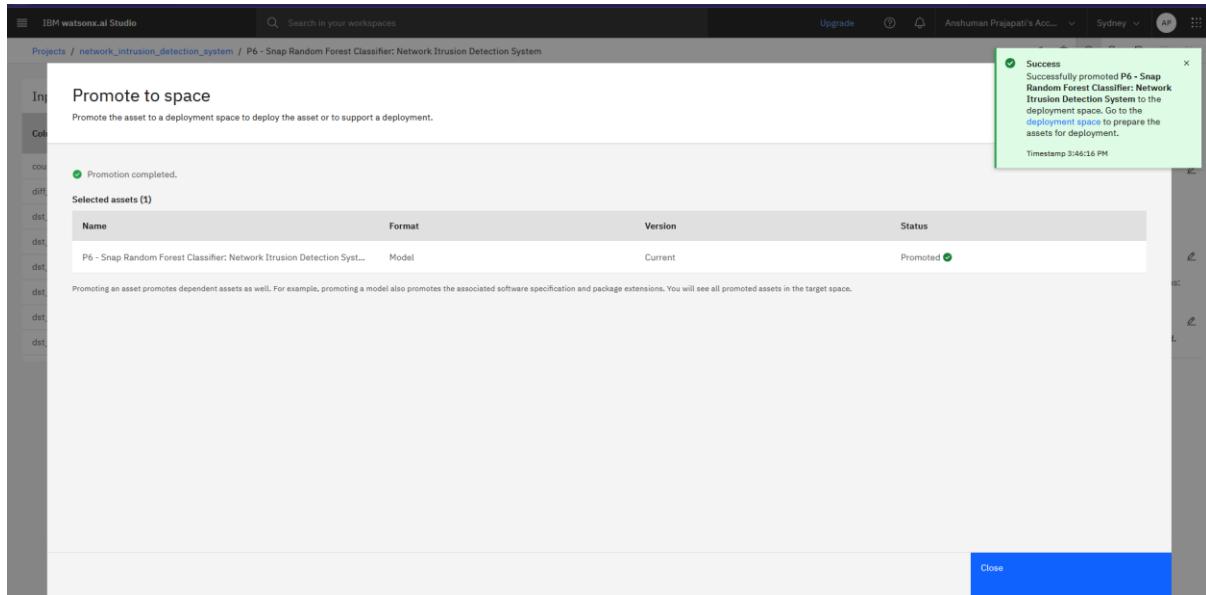
31. Select create new deployment .



32. Name the deployment space and make sure to select instance of watsonx.ai Runtime and press create.



33. Now press on promote.



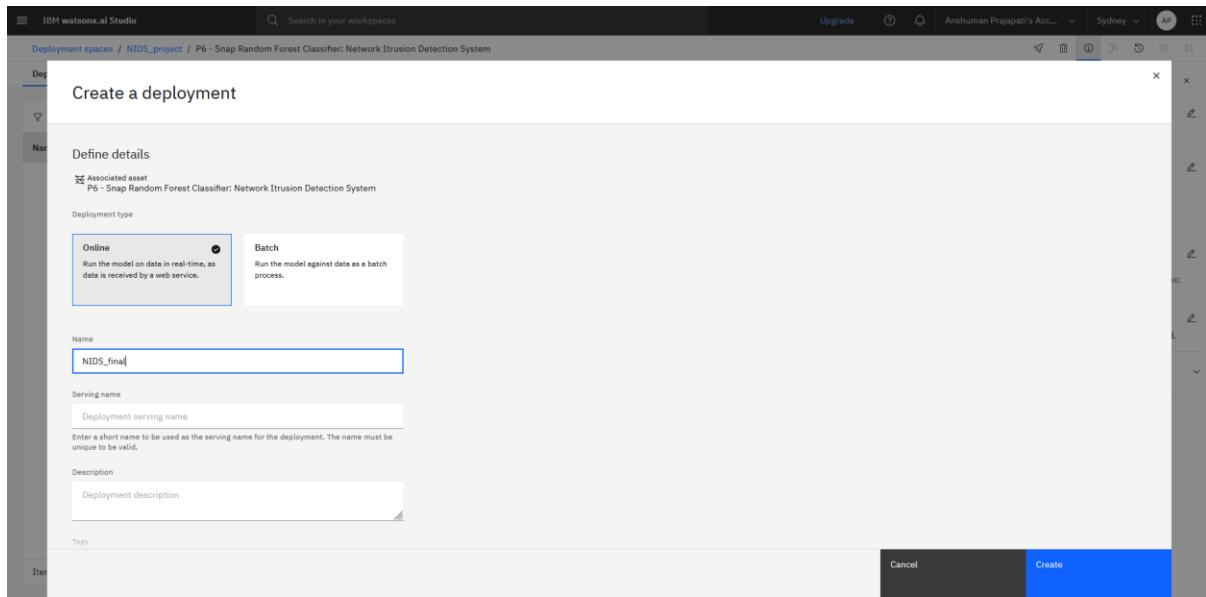
34. Now after it got promoted to space press on deployment space ,the pop up on top right corner.

The screenshot shows the IBM Watson AI Studio interface. At the top, there's a navigation bar with 'IBM Watson AI Studio', a search bar, and account information for 'Anshuman Prajapati's Acc...'. Below the navigation bar, the title 'NIDS_project' is displayed. The main area has tabs for 'Overview', 'Assets' (which is selected), 'Deployments', 'Jobs', and 'Manage'. On the left, there's a sidebar with '1 asset' and 'Asset types' (Models). The main content area is titled 'All assets' and lists one item: 'P6 - Snap Random Forest Classifier; Network Intrusion Detection System'. This item is described as a 'Machine learning model from AutoAI'. The table includes columns for 'Name', 'Last modified', and 'Service'. At the bottom right of the table, there are buttons for 'Import assets' and 'New asset'. The footer shows 'Items per page: 20' and '1 of 1 pages'.

35. Now click on the asset.

This screenshot shows the same IBM Watson AI Studio interface as the previous one, but with the 'Deployments' tab selected. The title 'Deployment spaces / NIDS_project / P6 - Snap Random Forest Classifier; Network Intrusion Detection System' is visible. The main content area shows a table with columns 'Name', 'Type', 'Status', 'Tags', and 'Last modified'. A message indicates that the asset doesn't have any deployments yet and suggests using the 'New Deployment' button. To the right, there's a detailed panel for the asset, including sections for 'About this asset' (Name: P6 - Snap Random Forest Classifier; Network Intrusion Detection System, Description: No description provided), 'Asset Details' (Type: wml-hybrid_0.1, Model ID: ab45f621-1fa9-4412-9307-a7964657ff15, Software specification: hybrid_0.1, Hybrid pipeline software specifications: autoai-kb_r724.1-py3.11), 'Tags' (Add tags to make assets easier to find), and 'Source asset details' (Last modified: 3 minutes ago by Service, Created on: Aug 3, 2025 by Anshuman Prajapati).

36. Now click on new **New Deployment**



37. Now select online and give this final deployment a name and press create.

The screenshot shows the 'Deployments' tab in IBM Watson Studio. A deployment named 'NIDS_final' is listed, showing it is 'Online' and 'Deployed'. The 'About this asset' panel is open, displaying the asset's name, type (wml-hybrid_0.1), model ID, software specification, and source asset details. The deployment row is highlighted with a red border.

Name	Type	Status	Tags	Last modified
NIDS_final	Online	Deployed		29 seconds ago Anshuman Prajapati (You)

38. After some time the deployment will be completed, then press on the deployment name.

The screenshot shows the IBM Watson AI Studio interface. In the top navigation bar, there are tabs for 'Deployment spaces / NIDS_project / P6 - Snap Random Forest Classifier: Network Intrusion Detection System /'. Below this, there's a section for 'NIDS_final' with a status of 'Deployed Online'. A red box highlights the 'Test' button under the 'API reference' tab. To the right, there's a detailed view of the deployment named 'NIDS_final' with sections for 'Name', 'Description', 'Deployment Details', 'Tags', and 'Associated asset'. The 'Associated asset' section lists 'P6 - Snap Random Forest Classifier: Net...'. At the bottom, it shows 'Last modified 1 minute ago' and 'Created on Aug 3, 2025'.

39. This will give the API reference page where API's are present to implement this model with other projects. Now press on test to test the model.

The screenshot shows the 'Enter input data' section of the API reference page. It has tabs for 'Text' and 'JSON', with 'Text' selected. There's a note: 'Enter data manually or use a CSV file to populate the spreadsheet. Max file size is 50 MB.' Below this are buttons for 'Download CSV template', 'Browse local files', and 'Search in space'. A table grid is shown for input data, with rows numbered 1 to 10. The first row contains the column headers: duration (double), protocol_type (other), service (other), flag (other), src_bytes (double), dst_bytes (double), land (double), wrong_fragment (double), urgent (double), hot (double), num_failed_logins (double), and logged_in (double). The table also includes a footer note: '0 rows, 41 columns' and a 'Predict' button.

40. We have many options to give input for test ,I choose local file as I have one.

IBM Watson AI Studio Search in your workspaces Upgrade Anshuman Prajapati's Acc... Sydney AP More

Deployment spaces / NIDS_project / P6 - Snap Random Forest Classifier: Network Intrusion Detection System /

NIDS_final Deployed Online

API reference Test

Enter input data

Text JSON

Enter data manually or use a CSV file to populate the spreadsheet. Max file size is 50 MB.

Download CSV template Browse local files Search in space

Clear all ×

	duration (double)	protocol_type (other)	service (other)	flag (other)	src_bytes (double)	dst_bytes (double)	land (double)	wrong_fragment (double)	urgent (double)	hot (double)	num_failed_logins (double)	logged_in (double)
1	0	tcp	private	REJ	0	0	0	0	0	0	0	0
2	0	tcp	private	REJ	0	0	0	0	0	0	0	0
3	2	tcp	ftp_data	SF	12983	0	0	0	0	0	0	0
4	0	icmp	eco_i	SF	20	0	0	0	0	0	0	0
5	1	tcp	telnet	RSTO	0	15	0	0	0	0	0	0
6	0	tcp	http	SF	267	14515	0	0	0	0	0	1
7	0	tcp	smtp	SF	1022	387	0	0	0	0	0	1
8	0	tcp	telnet	SF	129	174	0	0	0	0	1	0
9	0	tcp	http	SF	327	467	0	0	0	0	0	1
10	0	tcp	ftp	SF	26	157	0	0	0	0	1	0
11	0	tcp	telnet	SF	0	0	0	0	0	0	0	0

22,544 rows, 42 columns

Predict

41. After the file is loaded press predict. It will take some time to generate the result.

Prediction results

Prediction type: Binary classification

Prediction percentage: 22,544 records

Confidence level distribution: Number of records vs Confidence level (50-60%, 60-70%, 70-80%, 80-90%, 90-100%)

Display format for prediction results: Table view (selected) 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

42. This will generate a final result with up to 100% accuracy.

For upgrade of different types of attack detection just the data(.csv file) on which model is trained is to be changed for new data(Train_data_with_attack_types_enhanced.csv file) provided above and also you can refer this [GitHub](#) repo.