CAPSTONE PROJECT

POWER SYSTEM FAULT DETECTION AND CLASSIFICATION USING MACHINE LEARNING

Presented By:

1. Student Name-K.Srinivas
2. College Name-Sri Venkateswara college of engineering
3. DEPT- CSE[AI&ML]

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

Detecting and classifying power system faults quickly is important to maintain grid stability. The goal is to build a machine learning model that uses electrical data (like voltage and current) to identify different types of faults such as line-to-ground, line-to-line, and three-phase faults. This helps improve fault detection accuracy and ensures reliable power supply.

PROPOSED SOLUTION

- Develop a machine learning model that classifies power system faults using the dataset provided.The model will process electrical measurements to identify the type of fault rapidly and accurately.This classification will help automate fault detection and assist in quicker recovery action,ensuring system reliability.
- Data Collection:
  - Use the Kaggle dataset on power system faults containing labeled measurements (voltage, current, fault type).
- Data Preprocessing:
  - Clean the dataset to remove noise or null values. Normalize the features (e.g., voltage and current magnitudes) to prepare for model training.

  Features Engineering:
  - Extract time-domain and frequency-domain features from the phasor data to improve fault classification accuracy.
  - Deployment:
  - Deploy the model using IBM Cloud Lite Services, such as: IBM Watson Studio for model development,IBM Cloud Function or App Connect for deployment
  - Evaluation:
  - Validate model performance using:Accuracy,Precision &Recall F1-Score Confusion Matrix
  - Model Training: Train a classification model such as:
  - Decision Tree,Random Forest,Support Vector Manchine
  - LSTM
  - Result:

SYSTEM APPROACH

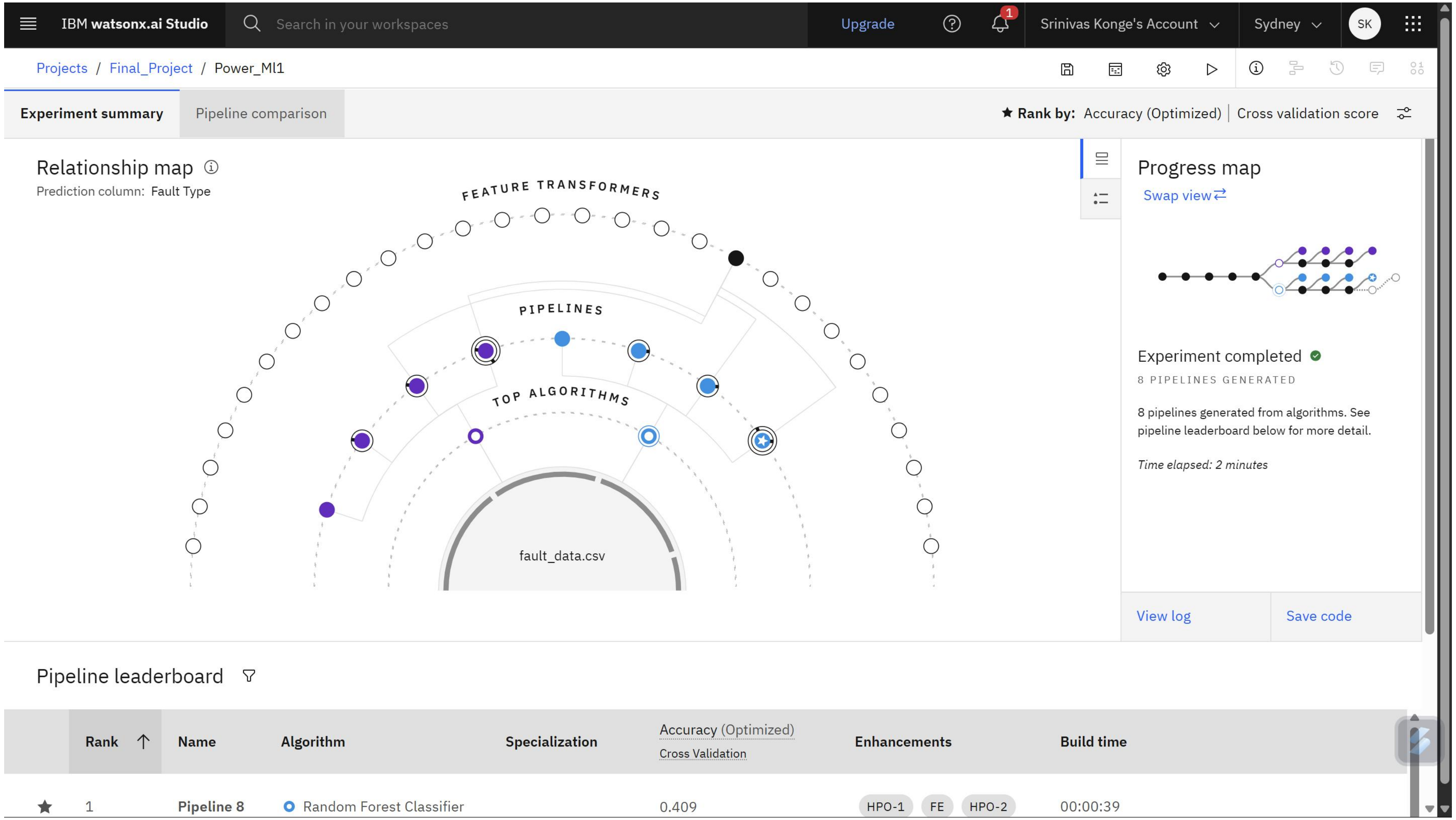
The "System Approach" section outlines the overall strategy and methodology for developing and implementing the rental bike prediction system. Here's a suggested structure for this section:

- System requirements:
- IBM Cloud(mandatory)
- IBM Watson studio for model development and deployment
- IBM Cloud Object storage for dataset handling

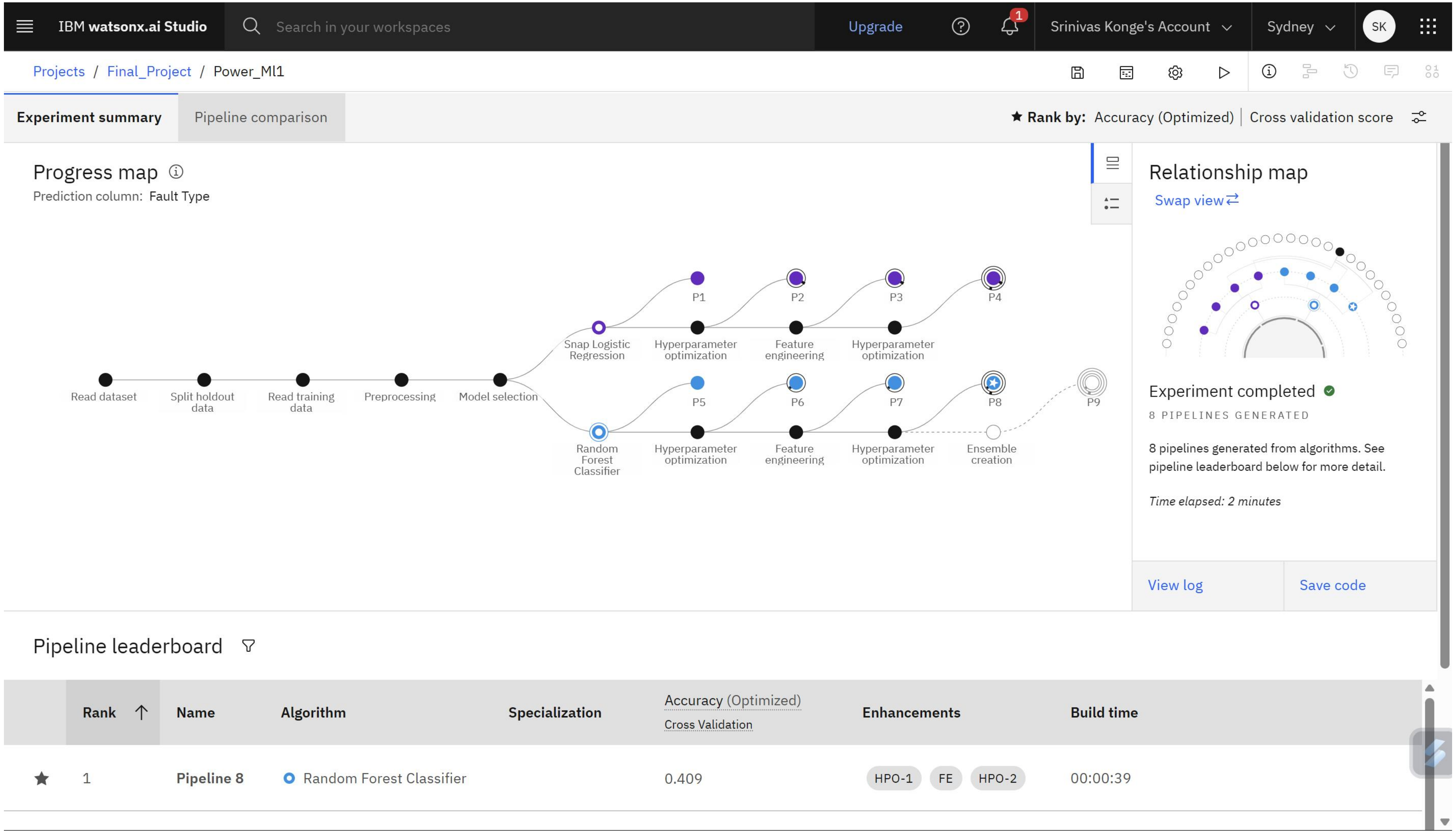
ALGORITHM & DEPLOYMENT

- In the Algorithm section, describe the machine learning algorithm chosen for predicting bike counts. Here's an example structure for this section:
- **Algorithm Selection:**
  - Random Forest Classifier (or SVM based on performance).
- **Data Input:**
  - Voltage,Current,and phasor measurements from the dataset.
- **Training Process:**
  - Supervised learninig using labelled fault types.
- **Prediction Process:**
  - Model Deployed on IBM Watson Studio with API endpoint for real-time predictions.

RESULT


IBM watsonx.ai Studio
Search in your workspaces
Upgrade
Srinivas Konge's Account
Sydney
SK
Projects / Final_Project / Power_Ml1
Experiment summary
Pipeline comparison
Rank by: Accuracy (Optimized) | Cross validation score
Relationship map
Prediction column: Fault Type
FEATURE TRANSFORMERS
PIPELINES
TOP ALGORITHMS
fault_data.csv
Progress map
Swap view
Experiment completed
8 PIPELINES GENERATED
8 pipelines generated from algorithms. See pipeline leaderboard below for more detail.
Time elapsed: 2 minutes
View log
Save code
Pipeline leaderboard
Rank
Name
Algorithm
Specialization
Accuracy (Optimized)
Cross Validation
Enhancements
Build time
1
Pipeline 8
Random Forest Classifier
0.409
HPO-1
FE
HPO-2
00:00:39

RESULT


IBM watsonx.ai Studio
Search in your workspaces
Upgrade
Srinivas Konge's Account
Sydney
SK
Projects / Final_Project / Power_Ml1
Experiment summary
Pipeline comparison
Rank by: Accuracy (Optimized) | Cross validation score
Progress map
Prediction column: Fault Type
Read dataset
Split holdout data
Read training data
Preprocessing
Model selection
Snap Logistic Regression
Random Forest Classifier
Hyperparameter optimization
Feature engineering
Hyperparameter optimization
Ensemble creation
P1
P2
P3
P4
P5
P6
P7
P8
P9
Relationship map
Swap view
Experiment completed
8 PIPELINES GENERATED
8 pipelines generated from algorithms. See pipeline leaderboard below for more detail.
Time elapsed: 2 minutes
View log
Save code
Pipeline leaderboard
Rank
Name
Algorithm
Specialization
Accuracy (Optimized)
Cross Validation
Enhancements
Build time
1
Pipeline 8
Random Forest Classifier
0.409
HPO-1
FE
HPO-2
00:00:39

RESULT

IBM watsonx.ai Studio | Search in your workspaces | Upgrade | Srinivas Konge's Account | Sydney | SK

Deployment spaces / DEP1 / P8 - Random Forest Classifier: Power_Ml1 /

Power_DEP2

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.

Clear all ×

	Fault ID (other)	Fault Location (Latitude, Longitude) (other)	Voltage (V) (double)	Current (A) (double)	Power Load (MW) (double)	Temperature (°C) (double)	Wind S
1	F001	(34.0522, -118.2437)	2200	250	50	25	20
2	F006	(34.05, -118.24)	2150	220	52	32	22
3	F011	(34.3732, -118.1586)	2118	221	45	20	20
4	F014	(34.3229, -118.46)	2289	192	52	35	28
5	F017	(34.9346, -118.9658)	2263	229	55	21	16
6	F020	(34.668, -118.6576)	2065	213	46	31	23
7							

6 rows, 12 columns

Predict

RESULT

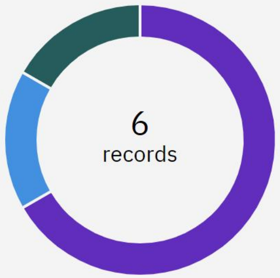
Prediction results

Close ×

Prediction type

Multiclass classification

Prediction percentage



Line Breakage Transformer Failure Overheating

Confidence level distribution

Display format for prediction results

(•) Table view ( ) JSON view Show input data ⓘ

	Prediction	Confidence
1	Line Breakage	39%
2	Line Breakage	35%
3	Transformer Failure	37%
4	Line Breakage	50%
5	Overheating	41%
6	Line Breakage	42%
7		
8		
9		
10		
11		
12		
13		

Download JSON file

edunet
foundation

CONCLUSION

- The project successfully demonstrates the use of machine learning for accurate and timely detection of power system faults. It highlights the importance of intelligent fault classification in enhancing grid reliability and provides a foundation for future improvements using real-time data and advanced models..

FUTURE SCOPE

- The system can be enhanced by using real-time data, improving model accuracy with deep learning, and expanding deployment to larger grids. Future work may also explore integration with edge computing or IoT devices for faster, decentralized fault detection.

REFERENCES

- Kaggle Dataset:
- https://www.kaggle.com/datasets/ziya07/power-system-faults-dataset
- IBM Cloud Documentation:
- https://www.ibm.com/cloud/watson-studio
- Academic Sources:
- Research papers on machine learning algorithm(eg,SVM,Random Forest)
- Journals on fault detection in power systems
- IEEE papers on grid reliability and automation
- General Resources:
- Scikit-learn Documentation
- Towards Data Science articles on classification and evaluation metrics
- Best practices in data preprocessing and model deployment

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Saisrinivas Konge

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 19, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/f4d7620d-54e1-4b1b-8875-347df04691da

IBM

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Saisrinivas Konge

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: Jul 19, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/ad557577-11b1-4b68-9803-566aa85180ff

IBM

IBM CERTIFICATIONS

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Saisrinivas Konge

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

THANK YOU