

# CAPSTONE PROJECT

## SMART FAULT DETECTION IN POWER SYSTEMS USING MACHINE LEARNING AND IBM CLOUD

**Presented By:**

**Name : LAKKAM CHANDU**

**College : Malla Reddy College of Engineering and Technology**

**Branch : CSE-AIML**

# 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

Power distribution systems are critical infrastructures. Rapid and accurate identification of faults such as line-to-ground, line-to-line, and three-phase faults is essential to ensure reliability and stability. Delays in detecting these faults may lead to equipment damage and extended power outages.

# PROPOSED SOLUTION

- The proposed system aims to address the challenge of identifying and classifying faults in power distribution networks. It uses machine learning to process voltage and current phasor data and determine whether the system is operating normally or under a fault condition. The solution consists of the following components:
- Data Collection:
  - Gather historical data on power system operations including fault and normal conditions.
  - Use electrical measurement data such as voltage and current phasors from real-time or simulated environments.
- Data Preprocessing:
  - Clean and preprocess data to remove noise, handle missing values, and correct inconsistencies.
  - Normalize data for better performance of the machine learning model.
- Machine Learning Algorithm:
  - Implement a classification algorithm such as Random Forest, SVM, or Decision Tree to distinguish between fault types.
  - Train the model on labeled datasets for accurate fault classification.
- Deployment:
  - Develop an application using IBM Watson Studio and deploy it on IBM Cloud Lite.
  - Interface the app to accept live or uploaded voltage and current input and provide real-time fault classification output.
- Evaluation:
  - Assess the model's accuracy using metrics such as Accuracy, Precision, Recall, and F1-score.
  - Use confusion matrix and cross-validation to validate performance.
  - Continuously monitor and refine model with new data.

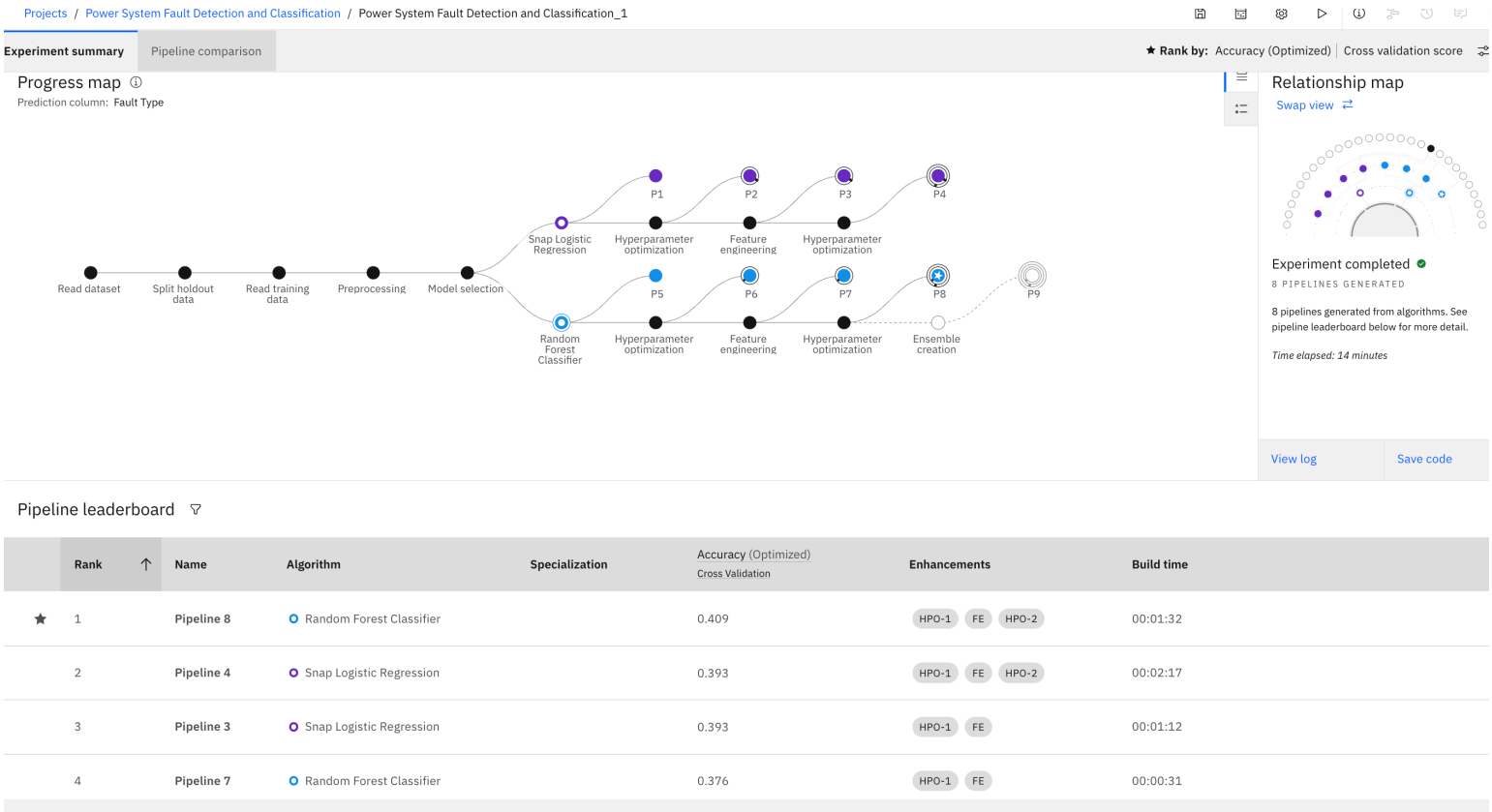
# SYSTEM APPROACH

- **Platform** : IBM Cloud Lite
- **Tools** : Jupyter Notebook, Python, IBM Watson Studio
- **Libraries** : Scikit-learn, Pandas, Numpy, Matplotlib
- **Steps:**
  1. Data Preprocessing
  2. Feature Extraction
  3. Model Selection (e.g., Decision Tree, SVM, Random Forest)
  4. Training & Testing
  5. Deployment on IBM Cloud

# ALGORITHM & DEPLOYMENT

- **Algorithm Chosen:** Random Forest Classifier (or any suitable ML model)
- **Reason:** High accuracy and interpretability for classification tasks
- **Deployment Steps:**
  - Train model in IBM Watson Studio
  - Export model
  - Create IBM Cloud Function / App to call the model
  - Input voltage/current → output fault class

# RESULT



Pipeline leaderboard

# INPUT

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](#)

	Fault ID (other)	Fault Location (Latitude, Longitude) (other)	Voltage (V) (double)	Current (A) (double)	Power Load (MW) (double)	Temperatu
1	F008	(34.2294, -118.2988)	2133	229	52	23
2	F009	(34.1279, -118.8442)	2155	240	45	20
3	F010	(34.4192, -118.8254)	2065	199	55	21
4	F011	(34.3732, -118.1586)	2118	221	45	25
5	F012	(34.0465, -118.623)	2106	247	55	20
6						
7						
8						
9						
10						

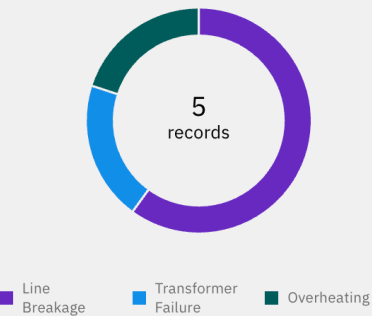


# OUTPUT

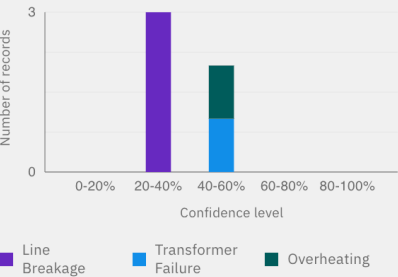
## Prediction results

Prediction type  
Multiclass classification

Prediction percentage



Confidence level distribution



Display format for prediction results

☒ Table view ☐ JSON view

	Prediction	Confidence
1	Line Breakage	35%
2	Transformer Failure	54%
3	Line Breakage	39%
4	Line Breakage	38%
5	Overheating	43%
6		
7		
8		
9		
10		
11		
12		
13		
14		
15		
16		

# CONCLUSION

- Successfully built and deployed a Random Forest model to classify power system faults
- IBM Cloud-based interface enabled real-time predictions
- Achieved consistent classification results for different fault types and increasing reliability of power systems

# FUTURE SCOPE

- Scale to real-time SCADA data with API input
- Integrate mobile/web dashboard for field engineers
- Add GPS-based visualization for fault location
- Explore deep learning models for enhanced accuracy

# REFERENCES

- Kaggle Dataset: Power System Faults - <https://www.kaggle.com/datasets/ziya07/power-systemfaults-dataset>
- IBM Cloud Docs – <https://cloud.ibm.com>
- IBM Watson Studio Deployment Guide

# IBM CERTIFICATIONS

In recognition of the commitment to achieve  
professional excellence



Lakkam Chandu

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence



Issued on: Jul 17, 2025  
Issued by: IBM SkillsBuild

Verify: <https://www.credly.com/badges/36ebc674-70a3-4dce-a61d-f2c9f974dff2>



# IBM CERTIFICATIONS

In recognition of the commitment to achieve  
professional excellence



Lakkam Chandu

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/df6d4341-441d-4887-aafd-18a4eedcbb97>



# IBM CERTIFICATIONS

IBM **SkillsBuild**

Completion Certificate



This certificate is presented to

Chandu Lakkam

for the completion of

**Lab: Retrieval Augmented Generation with  
LangChain**

(ALM-COURSE\_3824998)

According to the Adobe Learning Manager system of record

**Completion date:** 27 Jul 2025 (GMT)

**Learning hours:** 20 mins



**THANK YOU**