CAPSTONE PROJECT

POWER SYSTEM FAULT DETECTION AND CLASSIFICATION

Presented By:
1.Name- Soumya Banerjee
2.College-CMR College of Engineering and Technology
3.Department- Computer Science and Engineering

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

Power System Fault Detection and Classification -- Design a machine learning model to detect and classify different types of faults in a power distribution system.

- Using electrical measurement data (e.g., voltage and current phasors), the model should be able to distinguish between normal operating conditions and various fault conditions (such as line-to-ground, line-to-line, or three-phase faults).
- The objective is to enable rapid and accurate fault identification, which is crucial for maintaining power grid stability and reliability.

PROPOSED SOLUTION

- The proposed system aims to address the challenge of predicting the type of faults for each electrical component to ensure stability of power grid.
- This involves leveraging predictive analytics and machine learning techniques to classify the type of fault for unknown specification of electrical component. The solution will consist of the following components:
- Data Collection:
  - Gather historical data on electrical components, including voltage, current, power load, temperature and downtime.
  - Utilize real-time data sources, such as weather conditions, windspeed ,maintenance status, and fault data, to enhance prediction accuracy.
- Data Preprocessing:
  - Clean and preprocess the collected data to handle missing values, outliers, and inconsistencies.
  - Feature engineering to extract relevant features from the data that might impact fault detection.

- Machine Learning Algorithm:
  - Implement a machine learning algorithm, such as a classification model (e.g., random forest classification , snap logistic regression, or multiclass regression), to predict fault type based on historical patterns.
  - Consider incorporating other factors like weather conditions , maintenance status , and fault conditions to improve prediction accuracy.
- Deployment:
  - Develop a user-friendly interface or application that provides real-time predictions for fault types for different electrical components.
  - Deploy the solution on a scalable and reliable platform, considering factors like server infrastructure, response time, and user accessibility.
- Evaluation:
  - Assess the model's performance using appropriate metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), or other relevant metrics.
  - Fine-tune the model based on feedback and continuous monitoring of prediction accuracy.

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 Lite Account
- Cloud Object Storage
- Watson Studio
- Internet Browser
- Hardware

- **Library required to build the model**

  The following Python libraries were used within IBM Watson Studio notebooks:

- pandas: For data manipulation and preprocessing.
- numpy: For numerical computations.
- matplotlib & seaborn: For data visualization.
- scikit-learn: For machine learning model building (e.g., Logistic Regression, RandomForest).
- ibm_watson_machine_learning: For integrating with Watson ML services and deploying the model.

ALGORITHM & DEPLOYMENT

- **Algorithm Selection:**
  - Random Forest algorithm is a supervised ensemble machine learning algorithm used for classification and regression analysis.
  - The algorithm works by building multiple decision trees and combining their predictions to produce more accurate and stable results.
  - In context of the problem statement it is highly effective due to high accuracy in multiclass fault classification, robustness to noisy measurements, ability to handle unbalance datasets and fast inference speeds which makes it suitable for real-time detection.
- **Data Input:**
  - The input features that has been used for building the machine learning model are Fault ID, "Fault Location (Latitude, Longitude)",Voltage (V),Current (A),Power Load (MW),Temperature (°C),Wind Speed (km/h),Weather Condition, Maintenance Status, Component Health, Duration of Fault (hrs) and Down time (hrs).

- **Training Process:**

Dataset Preparation

- The dataset is split into training and testing sets (e.g., 80/20).
- Standardization or normalization may be applied to scale the features.

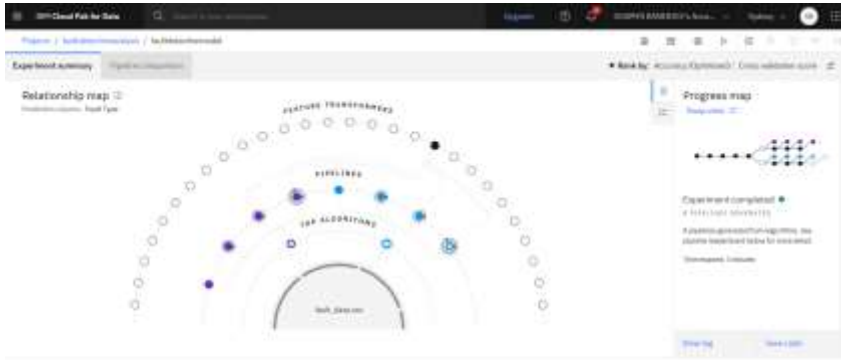
Model Training

- Multiple decision trees are constructed using random subsets of the data and features.
- Each tree learns different fault patterns.
- The final prediction is made using majority voting across all trees.

Prediction Process:

- The trained Random Forest model receives the new input as a feature vector.
- Each tree in the forest independently makes a prediction.
- The final output is the tree's prediction with majority of votes.
- The machine learning model takes **unknown power system feature** set as the test input and predicts the **fault type** for the power system.

RESULT



Summary interface for model training task

Machine learning model deployed successfully.

Machine learning model predicting the fault type for test input.

Structure of the test input data

CONCLUSION

- Power system fault detection and classification are critical for ensuring the reliability and safety of electrical grids.
- Fault detection systems are effective in combining AI with traditional methods (like impedance relays) yields better fault detection robustness.
- Cloud-based platforms like IBM Cloud, allow scalable model training and system monitoring.
- As systems become more complex ,fault detection and classification pose various challenges such as high complexity of modern grids,noise and signal distortion and data scarcity and imbalance.

FUTURE SCOPE

- This project has the potential for significant enhancements and real-world deployment. Below are some key directions in which this work can be extended:

Real-Time Monitoring Integration

- The system can be integrated with real-time sensors and IoT devices to monitor power systems live and detect faults as they occur.

Edge and Cloud Deployment

- Deployment on edge devices and cloud platforms (like IBM Cloud, AWS IoT, or Azure) can ensure high availability, remote accessibility, and scalable monitoring.

Enhanced Classification Accuracy

- Incorporating deep learning algorithms (e.g., LSTM, CNN) and larger datasets can improve the precision and robustness of fault classification.

REFERENCES

- P. Ilius, M. Almuhaini, M. Javaid, and M. Abido, “A Machine Learning–Based Approach for Fault Detection in Power Systems”, *Eng. Technol. Appl. Sci. Res.*, vol. 13, no. 4, pp. 11216–11221, Aug. 2023
- Power system faults dataset link – https://www.kaggle.com/datasets/ziya07/power-system faults-dataset
- IBM, "Getting started with watsonx.ai studio," IBM Documentation. [Online]. Available: https://dataplatform.cloud.ibm.com/docs/content/svc-welcome/wsl.html. [Accessed: Jul. 27, 2025].

IBM CERTIFICATIONS

- Screenshot/ credly certificate(getting started with AI)

IBM CERTIFICATIONS

- Screenshot/ credly certificate(Journey to Cloud)

edunet foundation

IBM CERTIFICATIONS

- Screenshot/ credly certificate(RAG Lab)

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Soumya Banerjee

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 23 Jul 2025 (GMT) **Learning hours:** 20 mins

THANK YOU