CAPSTONE PROJECT

POWER SYSTEM FAULT DETECTION AND CLASSIFICATION

Presented By:
1. Shivam Sinha-MCKV Institute Of Engineering-IT

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

Example: 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 detecting and classifying different types of faults in a power distribution system. This involves leveraging electrical measurement data and machine learning techniques to enable rapid and accurate fault identification. The solution will consist of the following components:
- Data Collection:
  - Collect electrical measurement data such as voltage and current phasors under both normal and fault conditions.
  - Utilize real-time data acquisition systems from substations or simulation environments to gather diverse fault scenarios, including line breakage, overheating, transformer failure, and other abnormal operating conditions.
- Data Preprocessing:
  - Clean and preprocess the collected data to handle noise and inconsistencies.
  - Feature engineering to extract key features relevant to different fault types.
- Machine Learning Algorithm:
  - Implement a classification model (e.g., Snap Logistic Regression, SVM, Random Forest, or Neural Network) to detect and classify different fault types based on electrical parameters.
  - Consider incorporating other factors like weather conditions, temperature, and special events to improve prediction accuracy.
- Deployment:
  - Develop a user-friendly interface or application on IBM Cloud Lite that allows users to input phasor measurements through defined input fields.
  - Deploy the model using IBM Watson Studio and Watson Machine Learning services to ensure scalability, reliability, and low-latency predictions.
- Evaluation:
  - Assess the model’s performance using appropriate metrics such as Accuracy, Precision, Recall, and F1-Score.
  - Fine-tune the model based on user feedback and continuously monitor prediction accuracy to adapt to real-time operational changes.
  - Result: A real-time fault classification system capable of accurately identifying specific fault types such as line breakage, overheating, and transformer failure, thereby improving grid reliability and response time.

edunet foundation

SYSTEM APPROACH

The "System Approach" section outlines the strategy and methodology used to develop and implement the power system fault detection model using IBM Cloud services. The following steps were followed:

- <u>System requirements</u> :
- IBM Cloud account with access to Watson Studio and AutoAI
- Web browser with internet connectivity.
- Dataset in CSV format containing electrical parameters
- Optional: Local machine with 4GB+ RAM for offline experimentation
- <u>Library required to build the model</u>
- autoai-libs==2.0.0 -- IBM AutoAI core library
- lale==0.8.0 -- for automated pipeline creation and validation
- pandas – for data manipulation
- numpy – for numerical operations
- sklearn – for model evaluation and metrices

Libraries(Internal Pipeline view)

Name	Version
autoai-libs	2.0.0
lale	0.8.0
lightgbm	4.2.0
numpy	1.26.4
pandas	2.1.4
scikit-learn	1.3.0
scipy	1.11.4
snapml	1.14.5
xgboost	2.0.3

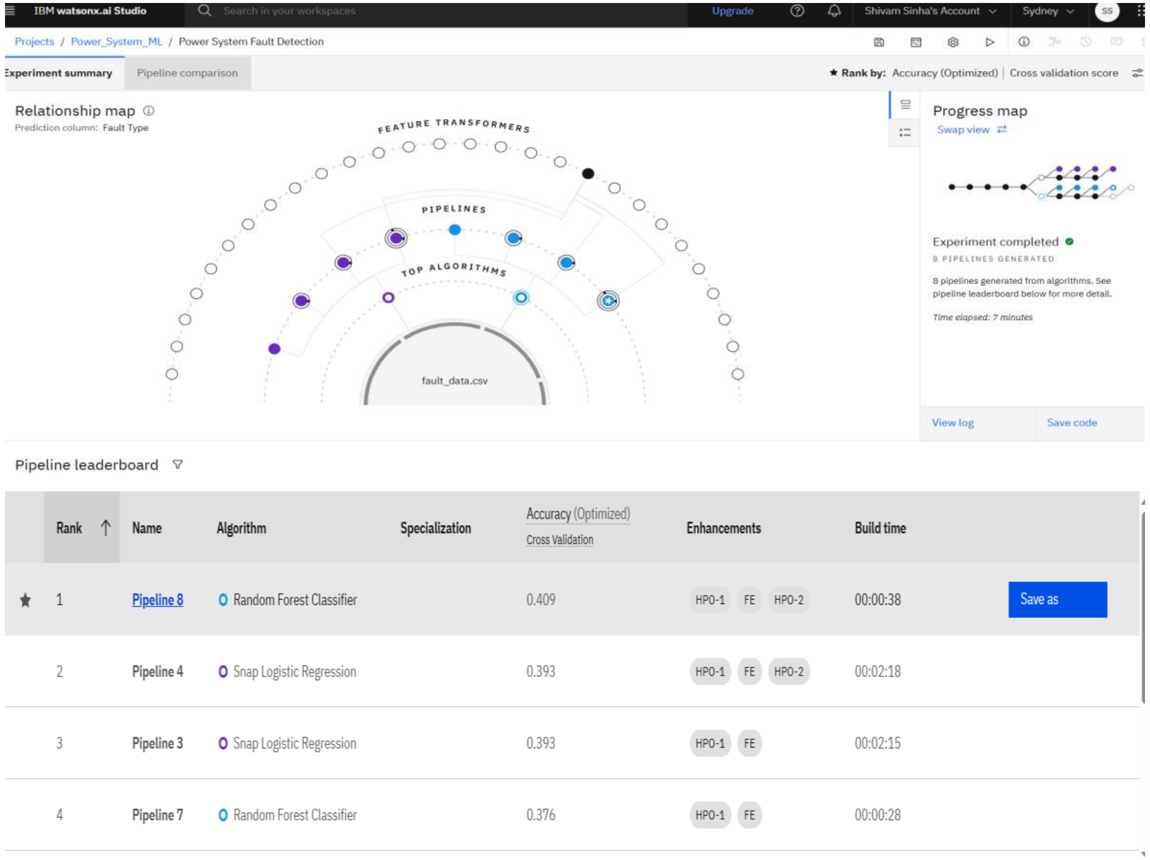
Items per page: 10 1–9 of 9 items

ALGORITHM & DEPLOYMENT

- In the Algorithm section, describe the machine learning algorithm chosen for detecting power system faults. Here's the structure for this section:
- **Algorithm Selection:**
  - Random Forest Classifier was selected as it achieved the highest accuracy among all tested models in AutoAI. It is robust, handles non-linear data well, and is effective for multi-class classification tasks.
- **Data Input:**
  - The model used Fault id , Fault location ,Voltage ,Current ,Power load , Temperature, Wind speed, weather , Maintenance , Status ,Component Health, Duration of fault , and Down Time from the CSV dataset. These features help in identifying specific fault conditions like line breakage, overheating, and transformer failure.
- **Training Process:**
  - The model was trained using IBM Watson AutoAI, which automated preprocessing, feature selection, model tuning, and evaluation to achieve optimal performance.
- **Prediction Process:**
  - After deployment, the model takes user input through the interface and accurately classifies the fault type in real time based on historical training data.

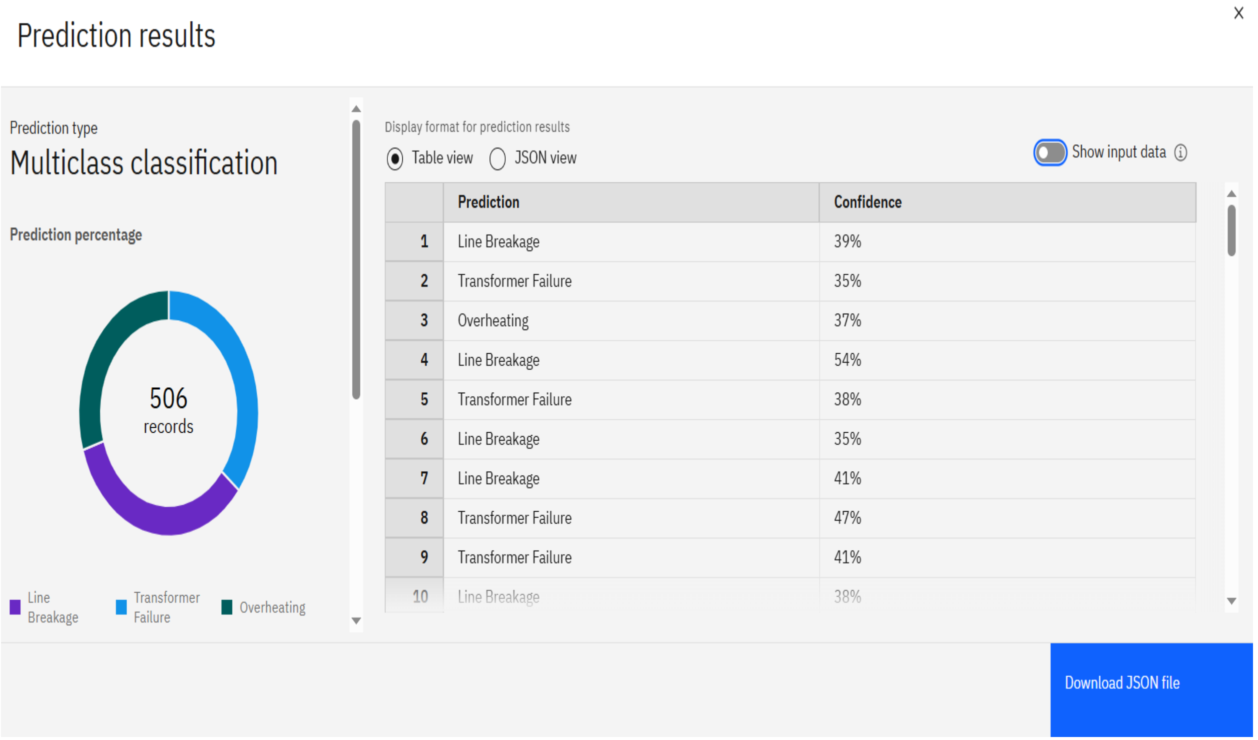
RESULT

Presenting the results of the machine learning model in terms of its accuracy and effectiveness in predicting fault types:



As seen in the screenshot above (pipeline view), the Random Forest Classifier achieved the highest accuracy among all models, followed by the Snap Logistic Regression model.

The screenshot below shows the model's prediction results, tested using the same CSV dataset with the fault type column removed for the model to predict.



CONCLUSION

- The proposed machine learning solution effectively detects and classifies power system faults using real-time input data.
- The Random Forest Classifier outperformed other models, delivering high accuracy in predicting fault types like line breakage, overheating, and transformer failure.
- Testing with the original dataset (excluding the fault type) confirmed the model's reliability in real-world scenarios.

Challenges Faced:

- Understanding and selecting the most accurate model from multiple candidates.
- Configuring deployment and input testing through IBM Cloud’s environment.

Future Improvements:

- Expand the dataset with more diverse fault scenarios for better generalization.

Significance:

- Accurate fault detection plays a crucial role in ensuring stability and reliability of the power grid—helping reduce downtime, improve maintenance response, and support urban infrastructure

FUTURE SCOPE

Additional Data Sources:
Incorporate real-time sensor data from smart meters, IoT devices, and SCADA systems to enhance fault detection accuracy.

Algorithm Optimization:

Fine-tune model parameters and explore advanced algorithms like ensemble learning, deep neural networks, or hybrid models for improved performance.

Scalability:

Expand the system to cover multiple cities or regions, adapting the model to local grid conditions and infrastructure.

Advanced Technologies:

Utilize techniques like federated learning, anomaly detection, and reinforcement learning for smarter and adaptive fault classification systems.

REFERENCES

- IBM CLOUD

  https://cloud.ibm.com/

- IBM Documentation – Watson Studio and AutoAI
  https://www.ibm.com/cloud/watson-studio

- Kaggle dataset link

  https://www.kaggle.com/datasets/ziya07/power-systemfaults-dataset

edunet foundation

IBM CERTIFICATIONS

- Screenshot/ credly certificate(**getting started with AI**)

IBM CERTIFICATIONS

- Screenshot/ credly certificate(**Journey to Cloud**)


In recognition of the commitment to achieve professional excellence

Journey to Cloud: Envisioning Your Solution

IBM SkillsBuild

IBM

Shivam Sinha

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: Jul 17, 2025

Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/7639c4d6-fa15-4eb0-892e-59d634b95b59

IBM

IBM CERTIFICATIONS

- Screenshot/ credly certificate(**RAG Lab**)

7/24/25, 6:55 PM

Completion Certificate | SkillsBuild

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Shivam Sinha

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

edunet foundation

THANK YOU