PIP4004 - INTERNSHIP

Review-3 Presentation

AI-ML Internship at GTRE (DRDO)

Student Details	
Name	V SAIPRIYA DIPIKA
Roll No	20211CSE0178
Section	8COM-CSE04/ SECTION-15
Batch No.	G-37

Under the Supervision of,

Dr. Sandeep Albert Mathias

Assistant Professor
School of Computer Science and Engineering
Presidency University

Name of the Program: **CSE**
Name of the HoD: **Dr. Asif Mohammed H.B**
Name of the Program Project Coordinator: **Mr. Amarnath J.L & Dr. Jayanthi. K.**
Name of the School Internship/Project Coordinators: **Mr. Md Ziaur Rahman /**
Dr. Sampath A K / Dr. Abdul Khadar A

Content

- Working domain or the technology
- Objectives of the work
- Literature Review
- Problem Statement
- System Requirements
- Advantages of Proposed System/Work
- Algorithm details
- Internship Roadmap
- Github Link

Working domain or the technology

Domain: Artificial Intelligence & Machine Learning (AI/ML) for Performance Analytics

Application: Aero-Gas Turbine Engine Analytics

Technologies:

- **Machine Learning Models –** Detect anomalies & predict failures
- **Data Visualization –** Graph-based analysis of sensor data
- **Predictive Maintenance –** Compare obtained data with ideal benchmarks

Impact: Early fault detection to prevent failures

Objectives of the work

Objective: Develop an AI-driven system to **monitor and analyse** aero-gas turbine engine performance.

- Data collection, preparation, cleaning and preprocessing
- UI and API design (if necessary)
- AI-Model Design
- Scaling and optimization of performance monitoring

Literature Review

1. Predicting Aircraft Engine Failures using Artificial Intelligence

(Bentaleb, Toumlal and Abouchabaka 2024)

- Uses AI to analyze sensor data (temperature, vibration, pressure) for early failure detection.
- Machine learning models identify patterns and anomalies to optimize maintenance.
- Enhances safety, reliability, and operational efficiency but lacks interactive visualization.

2. A Machine-Learning Approach to Assess Aircraft Engine System Performance

(Tong 2020)

- Focuses on ML-based engine health assessment for predictive maintenance.
- Uses data-driven modeling to detect trends and optimize maintenance schedules.
- Lacks user-driven graph generation and NLP-based interaction for deeper insights.

Literature Review

3. Enhancing Predictive Maintenance in the Industrial Sector: A Comparative Analysis of ML Models

(Levin 2024)

- Compares ML models to identify the most effective predictive maintenance approach.
- Reduces downtime and improves fault detection accuracy using AI.
- Does not explore interactive visualization or aerospace-specific challenges.

4. Predicting Machine Failures from Multivariate Time Series: An Industrial Case Study

(Vago, et al. 2024)

- Uses time-series sensor data to predict failures and improve maintenance strategies.
- AI models detect long-term performance deviations and automate alerts.
- Lacks user-driven analysis and generative AI for automated insight generation.

Literature Review

5. Advanced ML for Predictive Maintenance: A Case Study on Remaining Useful Life Prediction

(Meddaoui, Hachmoud and Hain 2024)

- Focuses on predicting the remaining useful life (RUL) of machinery using ML models.
- Helps schedule preventive maintenance and improve operational efficiency.
- Does not include interactive analysis or adaptive AI-driven insights.

6. Machine Learning-Based Fault-Oriented Predictive Maintenance in Industry 4.0

(Justus and Kanagachidambaresan 2024)

- Introduces a fault-classification framework for Industry 4.0 using ML.
- Automates failure detection and optimizes system reliability.
- Lacks flexibility for user-defined analysis and visualizations.

Problem Statement

Title: **Performance Monitoring and Analytics for Aero-Gas Turbine Engines**

- Automate graph analysis & performance evaluation
- Compare sensor data with ideal benchmarks
- Improve predictive maintenance & fault detection

System Requirements

Hardware Requirements:

- High-performance workstation with GPU support

Software Requirements:

- **DBMS** for structured data storage
- **Python with AI/ML and other Libraries** for data analysis and UI-Backend Integration
- **Visualization tools/Libraries (Matplotlib, Seaborn, etc)**

Advantages of Proposed System/Work

- **Performance Monitoring:** Faster detection of performance anomalies
- **Predictive Maintenance:** Prevents failures and reduces downtime
- **Automated Insights:** AI-driven trend analysis & recommendations
- **User-Driven Visualizations:** Dynamic selection of parameters for performance analysis.
- **Scalability & Efficiency:** Work with large datasets for accurate performance tracking

Algorithm Details

(i) Overview and Initialization

- INITIALIZE Flask app, MySQL database connection, and Mistral-7B model (offline, INT8 quantization)

- LOAD configurations: module_info.json, channel_mapping.json

- SET current_run_data = {run_id, selected_module, table_name, df, summary_cache}

- CONFIGURE logging (DEBUG, file/console output)

Algorithm Details

(ii) Data Retrieval and Summarization

- FUNCTION generate_query_data(run_id, module_name):

  - EXTRACT table_name from run_id (e.g., V12B34 from V12B34R1233)

  - QUERY database for Timestamp, module parameters, fixed sensors (up to 10,000 rows)

  - RETURN DataFrame or null if invalid

Algorithm Details

- FUNCTION summarize_data(df):
  - FOR each column (excluding Timestamp):
    - COMPUTE trend (erratic/upward/downward/stable) using differences
    - DETECT outliers (z-score > 3)
    - DETECT stagnation (diffs < 1e-5)
    - IDENTIFY missing data and anomalies (negatives/outliers)
  - RETURN summaries {trend, outliers, stagnation, missing, anomalies}

Algorithm Details

(iii) Plot Generation

- FUNCTION generate_plot_image(df, run_id, module, single_col=null):

  - SELECT columns (module-specific + fixed sensors or single_col)

  - DOWNSAMPLE to ~1,000 points for clarity

  - CREATE dual-axis plot (module params on primary, fixed sensors on secondary)

  - SAVE as PNG (150 DPI), encode to base64

  - RETURN base64 string

Algorithm Details

- ROUTE /generate_plot (POST):

  - GET run_id, selected_module

  - FETCH df = generate_query_data(run_id, selected_module)

  - UPDATE current_run_data with df, summaries

  - RETURN JSON {image: plot, comments: analysis, table_name}

Algorithm Details

(iv) AI Analysis and Chatbot

- FUNCTION analyze_with_mistral(df, module):
    - COMPUTE summaries = summarize_data(df)
    - SAMPLE 200 rows (prioritize outliers/anomalies)
    - FORMAT sampled data as CSV table
    - CONSTRUCT prompt with summaries, sampled data, fixed/module params
    - RUN Mistral-7B to generate comments (stagnation, erratic, malfunctions, anomalies)
    - RETURN comments or error

Algorithm Details

- FUNCTION chat_with_mistral(query):
    - IF query = "compare":
        - FETCH comparison_data for previous/specified run
        - SUMMARIZE both runs, sample 100 rows each
        - RUN Mistral-7B for comparison analysis
        - RETURN {response, comparison_plot}
    - IF query = "explain trend of <column>":
        - COMPUTE trend, generate single-column plot
        - RETURN {response, plot}
    - ELSE RETURN error

Algorithm Details

(v) Main Routes and Execution

- ROUTE / (GET):

    - RENDER index.html with module options

- ROUTE /chat (POST):

    - GET query, PROCESS with chat_with_mistral

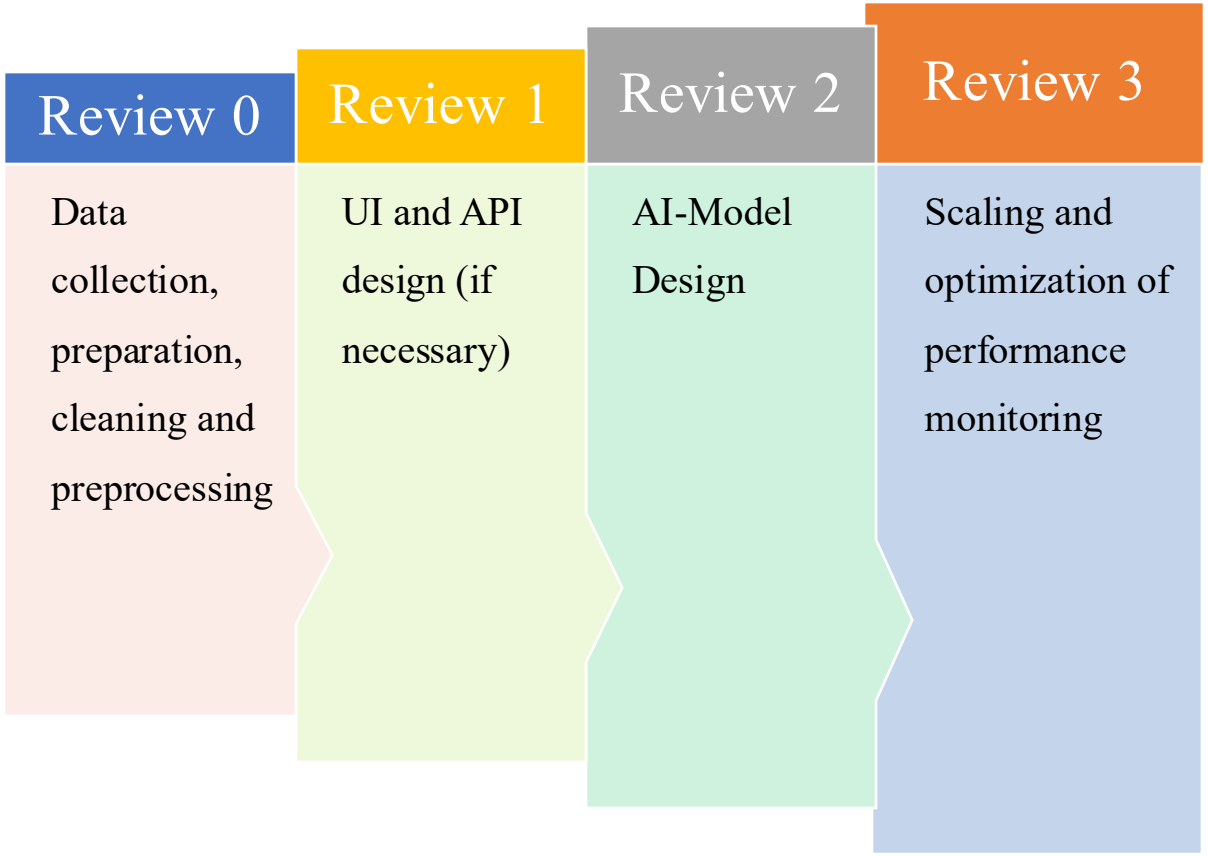
    - RETURN JSON {response, comp_image}

- MAIN:

    - START Flask server (port=5000, debug mode)

    - LOG errors if server fails

Internship Road Map



Github Link

https://github.com/saipriya-dipika/AI-ML-Internship

References

1. BENTALEB, Asmae, Kaoutar TOUMLAL, and Jaafar ABOUCHABAKA. "Predicting Aircraft Engine Failures using Artificial Intelligence." *International Journal of Advanced Computer Science & Applications* 15, no. 2 (2024)
2. Tong, Michael T. "A Machine-Learning Approach to Assess Aircraft Engine System Performance." In *Turbo Expo: Power for Land, Sea, and Air*, vol. 84058, p. V001T01A022. American Society of Mechanical Engineers, 2020.
3. Levin, Semen. "Enhancing predictive maintenance in the industrial sector: A comparative analysis of machine learning models." *In AIP Conference Proceedings, vol. 3243,* no. 1. AIP Publishing, 2024.
4. Pinciroli Vago, Nicolò Oreste, Francesca Forbicini, and Piero Fraternali. "Predicting Machine Failures from Multivariate Time Series: An Industrial Case Study." *Machines* 12, no. 6 (2024): 357.

References

5. Meddaoui, Anwar, Adil Hachmoud, and Mustapha Hain. "Advanced ML for predictive maintenance: a case study on remaining useful life prediction and reliability enhancement." *The International Journal of Advanced Manufacturing Technology* 132, no. 1 (2024): 323-335
6. Justus, Vivek, and G. R. Kanagachidambaresan. "Machine learning based fault-oriented predictive maintenance in industry 4.0." *International Journal of System Assurance Engineering and Management* 15, no. 1 (2024): 462-474.

Q&A

Thank you !!