PROGNOSAI: AI-DRIVEN PREDICTIVE MAINTENANCE SYSTEM USING TIME-SERIES SENSOR DATA

PREPARED BY: DURGA VEERA PRASAD V

DATASET: NASA TURBOFAN JET ENGINE (CMAPSS)

TABLE OF CONTENTS

- 1. Abstract
- 2. Introduction & Objectives
- 3. Dataset Description
- 4. System Architecture
- 5. Data Preparation
- 6. Model Development
- 7. Evaluation & Alert System
- 8. Visualization Dashboard
- 9. Results & Performance
- 10. Conclusion & Future Scope
- 11. Issues Faced
- 12. References & Tools Used

ABSTRACT

- PrognosAI is an AI-driven predictive maintenance framework that estimates the Remaining Useful Life (RUL) of turbofan jet engines using time-series sensor data. The system integrates deep learning, dynamic alerting, and visualization dashboards to provide interpretable insights and optimize maintenance planning.

INTRODUCTION AND OBJECTIVES

- Predictive maintenance aims to forecast equipment failures before they occur.

- Objectives:
- - Develop an LSTM model for accurate RUL prediction
- - Implement dynamic alerts for maintenance scheduling
- - Provide a real-time visualization dashboard

DATASET DESCRIPTION

- Source: NASA CMAPSS Dataset
- Features: 21 sensors + 3 operational settings
- Units: Multiple engine units (FD001–FD004)
- Target: Remaining Useful Life (RUL)

SYSTEM ARCHITECTURE

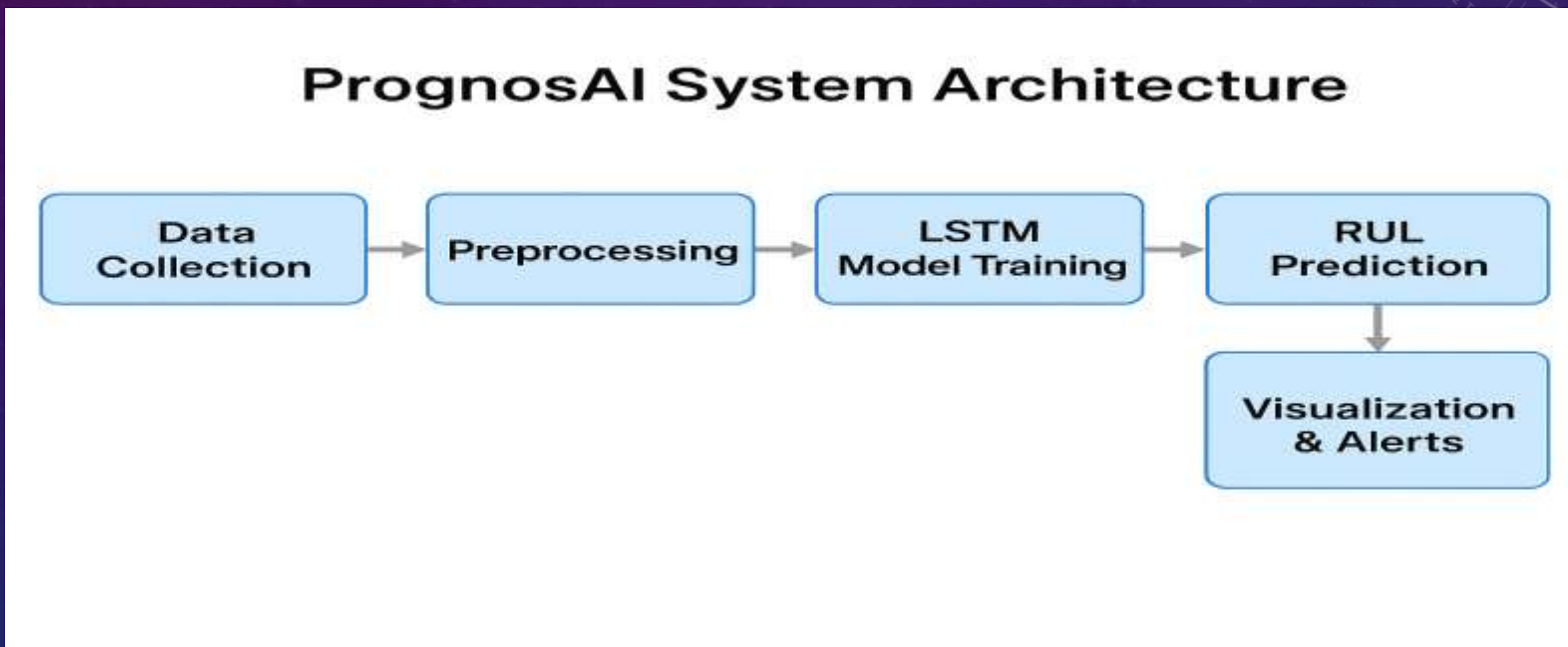

Figure1: PrognosAI System Architecture

DATA PREPARATION

- • Loaded CMAPSS datasets
- • Calculated RUL per engine cycle
- • Applied MinMaxScaler for normalization
- • Generated time-window sequences for LSTM input

- Libraries: pandas, numpy, sklearn.preprocessing

MODEL DEVELOPMENT

- • Built LSTM model using Keras Sequential API
- • Layers: LSTM(128) → Dropout → Dense(64, 32, 1)
- • Optimizer: Adam | Loss: MSE
- • Used 5-Fold Cross-Validation
- • Saved model and scalers for deployment

- Libraries: tensorflow, keras, sklearn

EVALUATION AND ALERT SYSTEM

- • Achieved $R^2 > 0.95$ with low RMSE and MAE
- • Dynamic alert thresholds:
-   - Critical: RUL ≤ 20%
-   - Warning: 20% < RUL ≤ 50%
-   - Normal: RUL > 50%
- • Generated alert summaries and visual reports

- Libraries: numpy, pandas, sklearn.metrics, matplotlib, plotly

VISUALIZATION DASHBOARD

- • Developed interactive Streamlit dashboard
- • Allowed upload of test CSV/TXT files
- • Displayed engine-wise RUL predictions and alerts
- • Supported result download and threshold control

- Libraries: streamlit, pandas, numpy, plotly, joblib, tensorflow.keras

- Figures 2,3 and 4: Dashboard Screens, RUL Trends, Alert Distribution

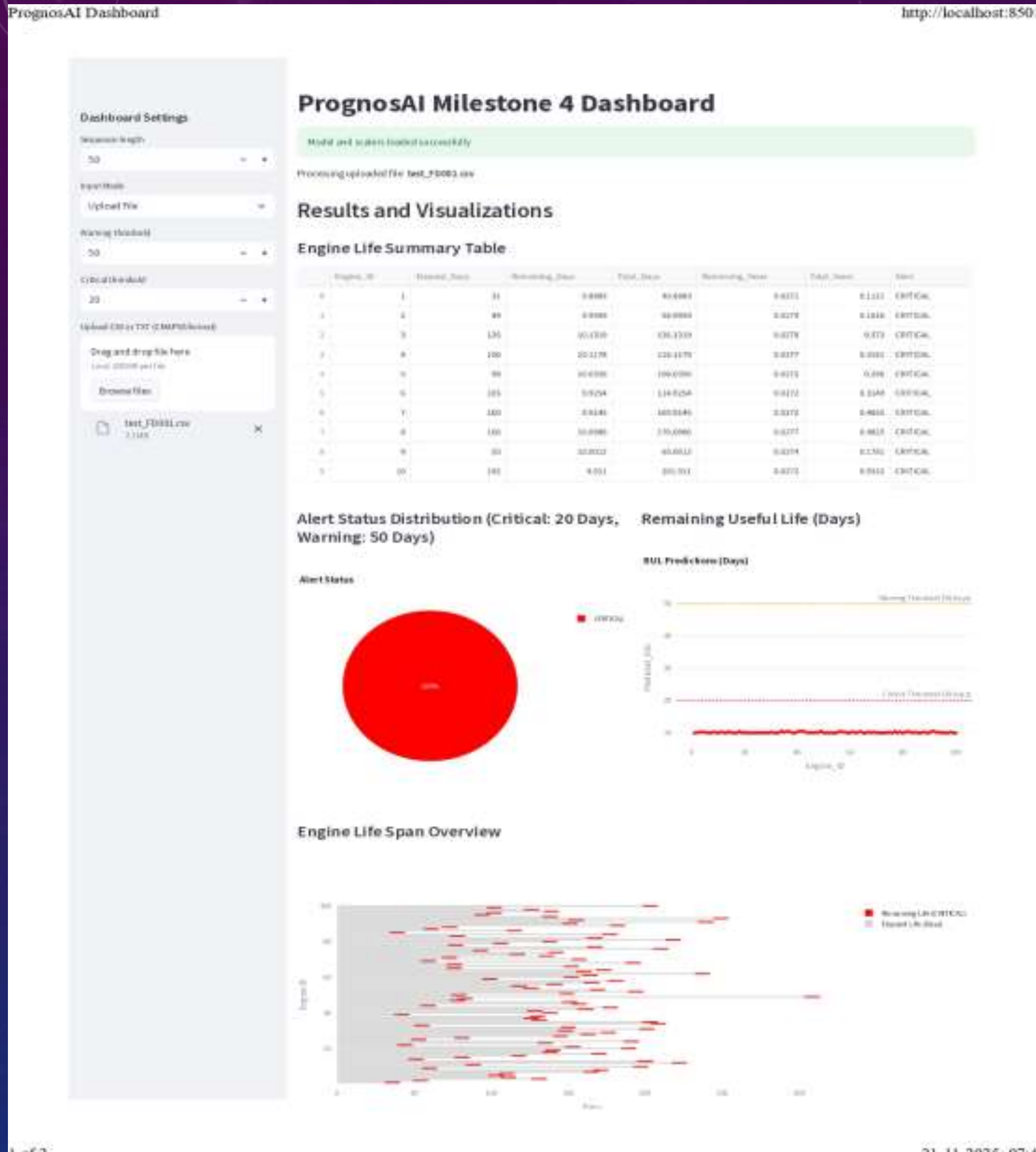

Figure2: PrognosAI Alert Distribution, Remaining Useful Life and Engine Life Span

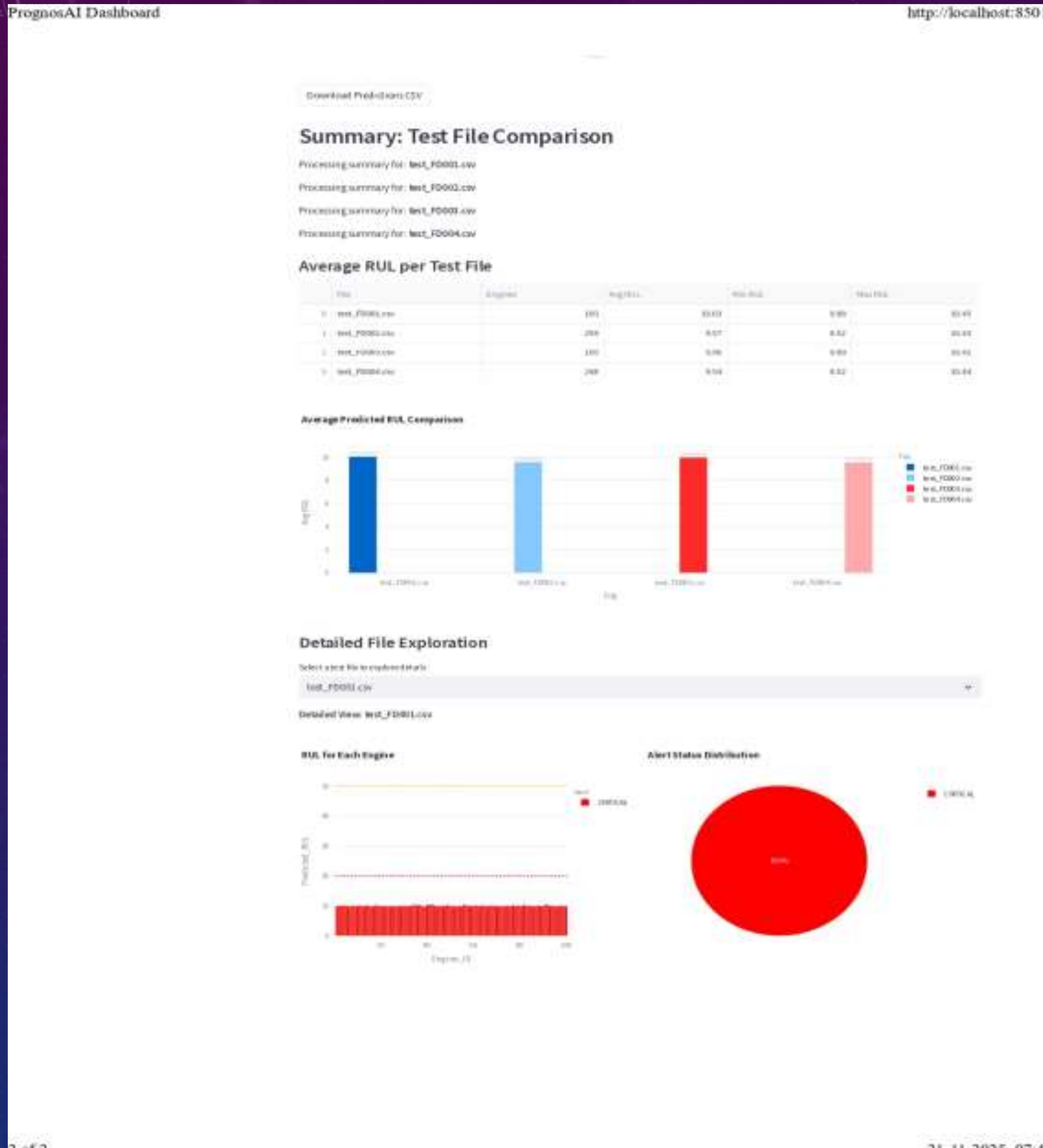

Figure3: PrognosAI Average RUL per Test File and Detailed File Exploration

RESULTS AND PERFORMANCE

- Metrics:
- - R^2: >0.95
- - RMSE: Low
- - MAE: Low

- Highlights:
- - Stable LSTM performance
- - Accurate RUL predictions with low error
- - Effective alert-based decision support

CONCLUSION AND FUTURE SCOPE

- Conclusion:
- PrognosAI effectively predicts Remaining Useful Life (RUL) using AI-based modeling and visualization.
- Future Enhancements:
- - Real-time IoT data integration
- - Adaptive online learning
- - Cloud deployment and API services

ISSUES FACED

- Feature-name mismatches between training and test sets caused pipeline failures. Thanks to peers and collaborators for support.
- Scaling and sequence generation failed when sensor columns had NaNs or fewer than the required 50 time-steps.
- Model inference and plotting lagged significantly when combining multiple large CMAPSS datasets.

REFERENCES AND TOOLS USED

- Languages: Python 3.10
- Libraries: tensorflow, keras, pandas, numpy, sklearn, matplotlib, plotly, streamlit, joblib, reportlab
- Dataset: NASA CMAPSS
- Environment: Jupyter Notebook / VS Code / Streamlit

Thank You