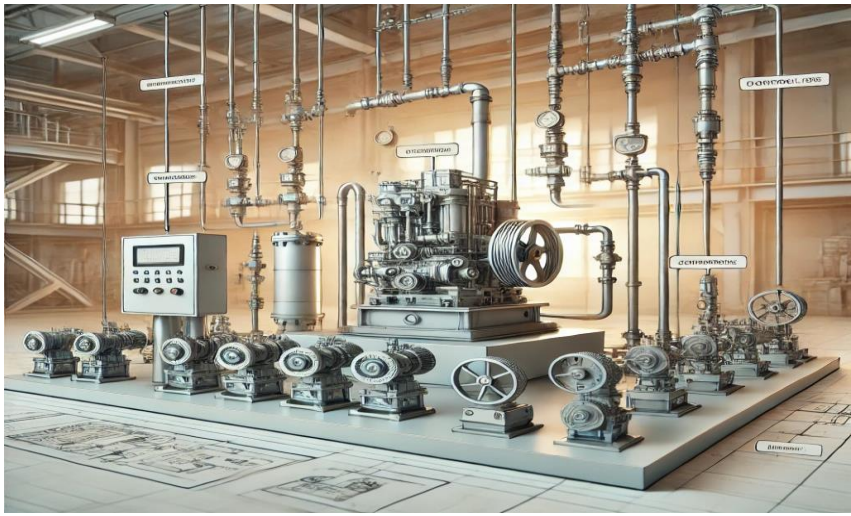
Predictive Maintenance for Hydraulic Systems Using Machine Learning

Rock Deng

2025/2/16

Introduction



- Why Predictive Maintenance?
  - Hydraulic system failures cause costly downtime.
  - Preventive maintenance reduces unexpected failures and improves efficiency.

- Objective:
  - Develop a machine learning model to predict failures using sensor data.

- Dataset:
  - Condition Monitoring of Hydraulic Systems dataset.

Problem Statement

- Current Challenges:

  - Manual inspections are time-consuming.

  - Unexpected failures cause operational disruptions.

  - Traditional maintenance is reactive, not predictive.

Problem Statement

- Proposed Solution:

  - Utilize machine learning to classify failure risks in hydraulic systems.

  - Implement efficient models for real-time monitoring.

Related Work

- Prior Research:

  - Machine learning models (Random Forests, SVMs, Neural Networks) have been tested for failure prediction.

  - Feature engineering techniques (PCA, time-series forecasting) have been applied.

  - Deep learning (LSTMs) has shown promise but requires high computational power.

- Gaps in Research:

  - Need for interpretable and computationally efficient models.

  - Limited real-time deployment applications.

How This Project Builds on Prior Work

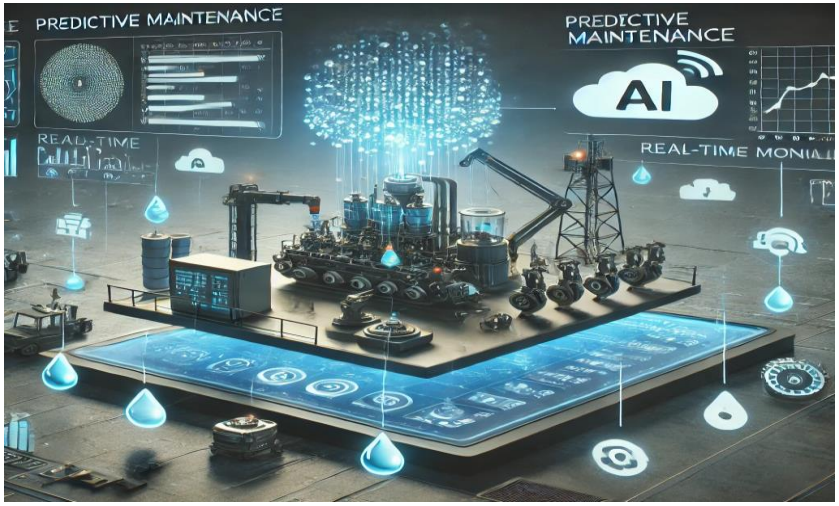
- Enhancing Model Interpretability:
  - Use explainable AI (XAI) techniques.

- Lightweight Models for Real-World Deployment:
  - Investigate XGBoost and Decision Trees.

- Exploring Incremental Learning for Adaptive Models:
  - Adjust models dynamically based on new sensor data.

Proposed Work

- Dataset & Tools:
  - Condition Monitoring of Hydraulic Systems dataset.
  - Python (Pandas, NumPy, Scikit-learn, TensorFlow, PyTorch).
  - Time-series analysis tools (statsmodels, Prophet).



- Key Steps:
  1. Data preprocessing (handling missing values, normalization).
  2. Model selection (Decision Trees, XGBoost, ARIMA, LSTM).
  3. Performance evaluation (Accuracy, Precision, Recall, RMSE).
  4. Optimizing model for real-time deployment.

Evaluation Plan

- Metrics for Performance Measurement:

  - Classification Models: Accuracy, Precision, Recall, F1-score.

  - Time-Series Models: Root Mean Squared Error (RMSE).

  - Efficiency: Training time and inference speed.

- Comparison of Models:

  - Test various models and analyze feature importance.

  - Trade-off between accuracy and computational cost.

Project Timeline

- Week 1: Dataset exploration and preprocessing
- Week 2-3: Implement baseline models (Decision Trees, XGBoost)
- Week 4: Experiment with time-series models (LSTM, ARIMA)
- Week 5: Evaluation and performance comparison
- Week 6: Final report writing and refinements

Thank You!

- Questions?