Project Report

Project Name: PrognosAI: AI-Powered Predictive Maintenance System Using Sensor

Data – 09-10-2025

Prepared by: Shivani Sharma

1. Objective

The objective of Milestone 2 is to develop and train a **time-series deep learning model (LSTM/GRU)** capable of learning patterns from sensor data to **predict the Remaining Useful Life (RUL)** of machine components.

2. Dataset

- **Source:** Milestone 1 preprocessed sensor data (sensor_data.csv).
- **Features:** Sensor readings (e.g., SensorA, SensorB, SensorC).
- **Target:** RUL (Remaining Useful Life), computed as a decreasing sequence from maximum cycles to 0.
- **Preprocessing steps:**
    - Missing values filled using forward fill.
    - Sensor values normalized using **StandardScaler**.
    - RUL calculated for each data point.

3. Methodology

3.1 Sequence Preparation

- Rolling window sequences created for LSTM input.
- **Window size:** 30 time steps.
- Train-validation split: 80:20, maintaining time order (no shuffling).

3.2 Model Architecture

- **Layers:**
    1. LSTM (64 units, return_sequences=True) → Dropout 0.2
    2. LSTM (32 units, return_sequences=False) → Dropout 0.2
    3. Dense (16 units, activation='relu')
    4. Dense (1 unit) → Output RUL

- **Loss function:** Mean Squared Error (MSE)
- **Optimizer:** Adam
- **Epochs:** 30
- **Batch size:** 32

Optional: Future improvements can include Early Stopping and Model Checkpoint for better performance.

4. Training Results

4.1 Training & Validation Loss Curve

- The training and validation loss decreased over epochs, indicating **proper convergence** of the model.
- The model did not overfit significantly, showing good learning capability.

4.2 Validation Predictions vs Actual RUL

- The predicted RUL closely follows the actual RUL values.
- Visual inspection confirms that the model captures the trend and provides reasonable RUL predictions.

5. Model Saving

- The trained model was saved as **LSTM_RUL_model.h5**.
- This model can be loaded later for evaluation, testing, or deployment.

6. Conclusion

- Successfully developed an **LSTM-based time-series model** for RUL prediction using Milestone 1 sensor data.
- The model shows **good convergence** and reasonable predictive performance.
- **Future improvements:**
    1. Use **GRU layers** for faster training.
    2. Implement **EarlyStopping** and **Model Checkpoint** to avoid overfitting.
    3. Hyperparameter tuning (LSTM units, window size, batch size, learning rate).

7. References

1. Chollet, François. *Deep Learning with Python.* Manning, 2017.

2. Brownlee, Jason. *Deep Learning for Time Series Forecasting.* Machine Learning Mastery, 2018.
3. TensorFlow Documentation: https://www.tensorflow.org