PROJECT ABSTRACT

Turbofan Engine RUL Prediction using LSTM (NASA C-MAPSS)

A long short-term memory network that predicts remaining useful life from turbofan sensor sequences on the NASA C-MAPSS dataset, evaluated with RMSE and the asymmetric scoring function, with an interactive degradation demo, notebook and viva kit.

01. Title

Turbofan Engine RUL Prediction using LSTM (NASA C-MAPSS)

02. Introduction / Background

Predictive maintenance replaces fixed overhaul schedules with data-driven ones: if you can estimate how many operating cycles an engine has left, you service it just in time — avoiding both catastrophic failure and wasted early teardowns. The NASA C-MAPSS dataset (Saxena & Goebel) is the standard benchmark: run-to-failure simulations of turbofan engines with 21 sensor channels and 3 operating settings per cycle across four sub-datasets (FD001–FD004) of increasing difficulty. Remaining useful life (RUL) is a sequence-regression problem, and LSTMs are the natural architecture for learning degradation dynamics from sensor histories.

03. Problem Statement

Maintenance engineers need a per-unit RUL estimate from sensor streams, with errors on the over-prediction side penalized more (predicting 50 cycles left when 10 remain is dangerous). A defensible project must use the real C-MAPSS benchmark, a documented LSTM training protocol, and the field's own evaluation — RMSE plus the asymmetric scoring function — with honest comparison against a classical baseline.

04. Objectives

- Train an LSTM sequence regressor on C-MAPSS FD001 with RMSE ≤ 16 cycles on the test set (design target).
- Apply the standard preprocessing: sensor normalization, piecewise-linear RUL labels, sliding windows of 30 cycles.
- Evaluate with RMSE and the asymmetric scoring function; compare against an SVR/linear baseline.
- Build an interactive demo showing sensor degradation traces, the RUL gauge and the reasoning behind each prediction.
- Deliver the complete student kit: notebook, trained model, demo, report, PPT and viva Q&A.;

05. Proposed Methodology

FD001 run-to-failure trajectories are normalized per sensor; RUL labels use the piecewise-linear convention (capped at 125 cycles early in life). Sliding windows of 30 cycles × 21 sensors feed a two-layer LSTM (100 + 50 units) with dropout and a dense regression head, trained with Adam and MSE loss. Evaluation on the held-out test engines reports RMSE and the asymmetric score $S = \Sigma \exp(-d/13)-1$ for under-prediction, $\Sigma \exp(d/10)-1$ for over-prediction. A classical SVR baseline on the same windows quantifies what the LSTM adds.

06. System Architecture / Working

Input (30 × 21 sensor window) → LSTM(100) → LSTM(50) → Dropout → Dense(1) predicting remaining cycles. In the web demo, selecting an engine unit and scrubbing the operating-cycle slider renders the sensor degradation traces with the "now" line and failure marker; the gauge shows predicted RUL while the side panel explains the health index, trending sensors and operating regime. The footer states the demo uses a compact estimator on representative traces.

07. Hardware / Software Requirements

- Python 3.10+, TensorFlow/Keras, NumPy, Pandas, Matplotlib, scikit-learn, Jupyter
- NASA C-MAPSS dataset (public, FD001–FD004)
- Any modern laptop; a GPU shortens training (~30–60 min expected) but is not required
- Web demo: any current browser, runs offline after download

08. Expected Outcomes

- Test RMSE ≤ 16 cycles on FD001 (design target), asymmetric score, SVR baseline comparison
- Degradation-trace and RUL-vs-true plots documented in the report
- Interactive demo with scrubbable unit life-cycle and prediction reasoning
- Measured values are reported after the built-to-order training run — the target is not presented as a measured result

09. Applications

Demonstrates sequence-based predictive maintenance for rotating machinery; the windowed-LSTM pattern transfers to batteries, bearings and tool wear. Educational prototype on simulated data — no airworthiness or operational claim is made.

10. Future Scope

Multi-condition training on FD002–FD004 with operating-regime conditioning; uncertainty-aware RUL with prediction intervals; remaining-life cost optimization for maintenance scheduling.