Industrial Internship Report on

Predictive Maintenance of Turbofan Engines

Prepared by

Agamya David

Executive Summary
This report provides details of the Industrial Internship provided by upskill Campus and The IoT Academy in collaboration with Industrial Partner UniConverge Technologies Pvt Ltd (UCT). This internship was focused on a project provided by UCT, which had to be completed, including the report, in a 6-week timeframe. My project was **Predictive Maintenance of Turbofan Engines**, which involved developing machine learning models to predict the Remaining Useful Life (RUL) of aircraft engines based on time-series sensor data. This internship gave me a very good opportunity to get exposure to a real-world industrial problem and to design and implement a solution for it. It was an overall great experience to have this internship.

TABLE OF CONTENTS

1 Preface

This report summarizes the work completed over the 6-week industrial internship, a valuable opportunity for my career development provided by UniConverge Technologies (UCT) and upskill Campus. This internship was essential in bridging the gap between academic theory and real-world industrial application. My project focused on **predictive maintenance for turbofan engines**, a critical challenge in the aerospace industry, where the goal was to predict the Remaining Useful Life (RUL) of engines from sensor data. The program was well-structured, guiding me from exploring the problem statement and planning a solution to implementing and validating the final models.

Over the six weeks, I gained practical skills in data analysis, feature engineering, and the implementation of both classical (Random Forest) and deep learning (LSTM) models. This experience has been incredibly insightful, teaching me the importance of establishing a baseline and iteratively improving upon it. I hope this report serves as a useful reference for my peers and juniors, and I encourage them to seek out similar opportunities to apply their skills to tangible, real-world problems.

2 Introduction

2.1 About UniConverge Technologies Pvt Ltd

A company established in 2013 and working in Digital Transformation domain and providing Industrial solutions with prime focus on sustainability and RoI.

For developing its products and solutions it is leveraging various **Cutting Edge Technologies e.g. Internet of Things (IoT), Cyber Security, Cloud computing (AWS, Azure), Machine Learning, Communication Technologies (4G/5G/LoRaWAN), Java Full Stack, Python, Front end** etc.

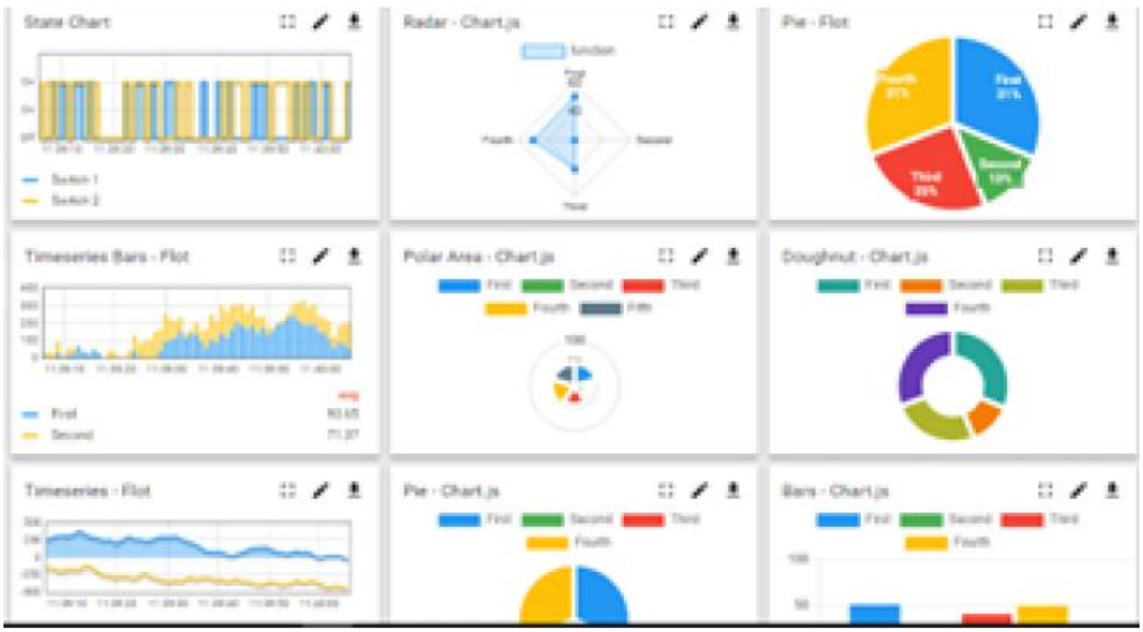
i. UCT IoT Platform

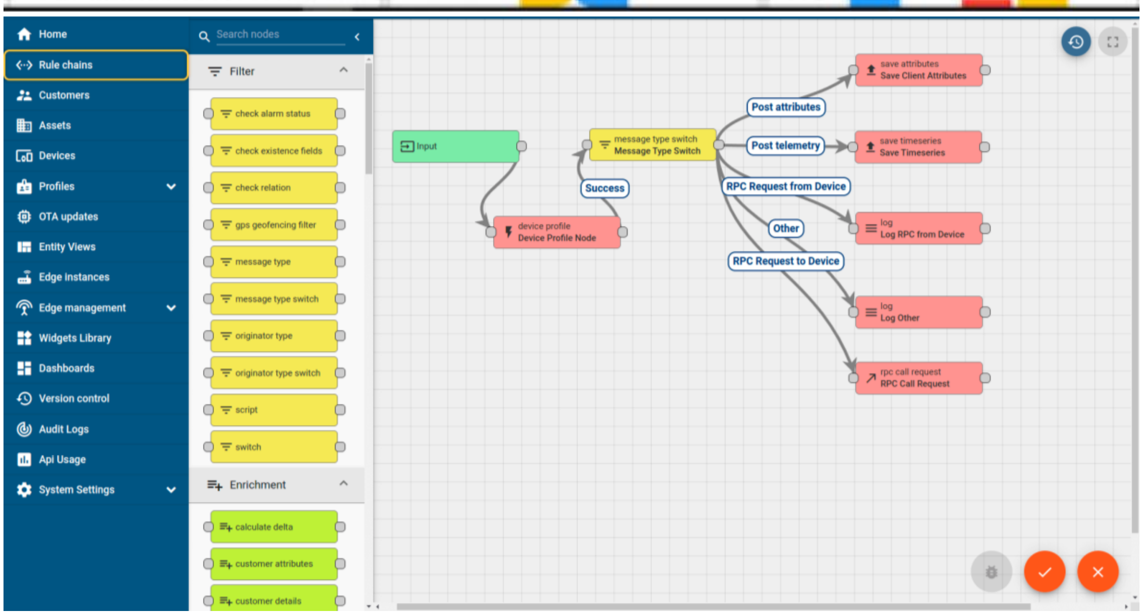
UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

- It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA
- It supports both cloud and on-premises deployments.

It has features to

- Build Your own dashboard
- Analytics and Reporting
- Alert and Notification
- Integration with third party application(Power BI, SAP, ERP)
- Rule Engine





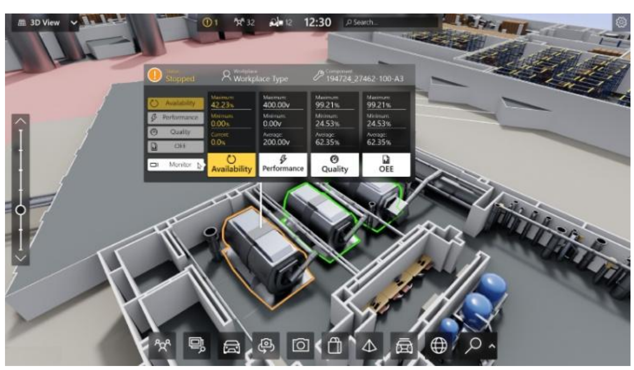
ii. Smart Factory Platform

Factory watch is a platform for smart factory needs.

It provides Users/ Factory

- with a scalable solution for their Production and asset monitoring
- OEE and predictive maintenance solution scaling up to digital twin for your assets.
- to unleased the true potential of the data that their machines are generating and helps to identify the KPIs and also improve them.
- A modular architecture that allows users to choose the service that they what to start and then can scale to more complex solutions as per their demands.

Its unique SaaS model helps users to save time, cost and money.

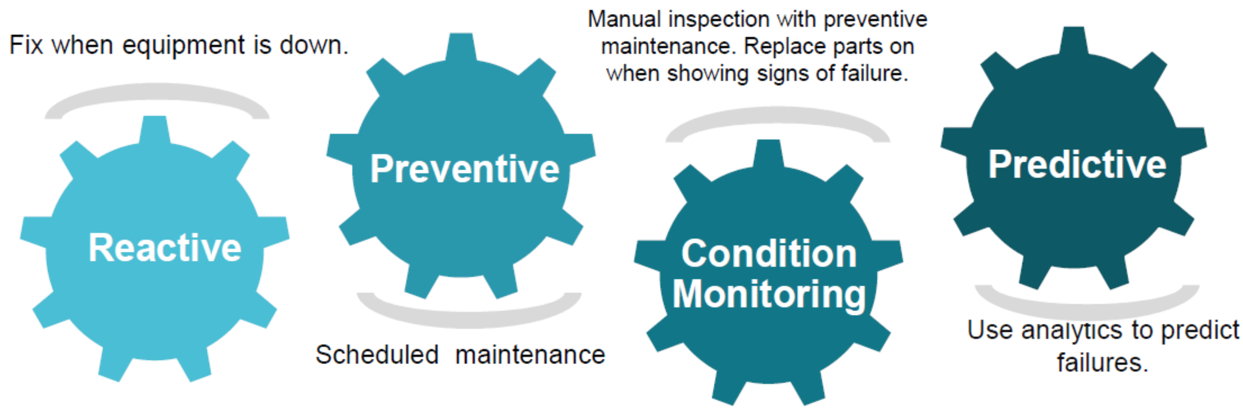


iii. LoRaWAN based Solution

UCT is one of the early adopters of LoRAWAN teschnology and providing solution in Agritech, Smart cities, Industrial Monitoring, Smart Street Light, Smart Water/ Gas/ Electricity metering solutions etc.

iv. Predictive Maintenance

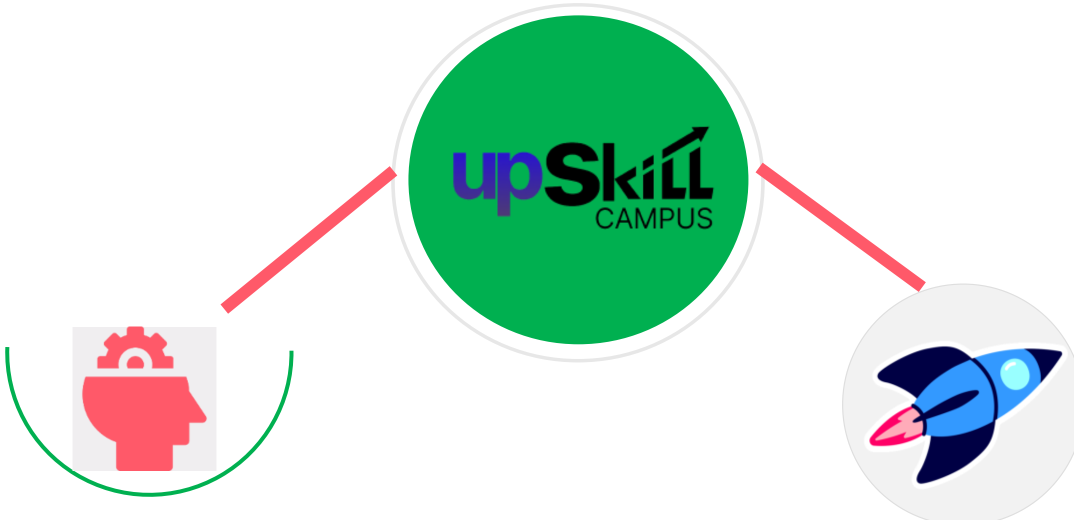
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.



2.2 About upskill Campus (USC)

upskill Campus along with The IoT Academy and in association with Uniconverge technologies has facilitated the smooth execution of the complete internship process.

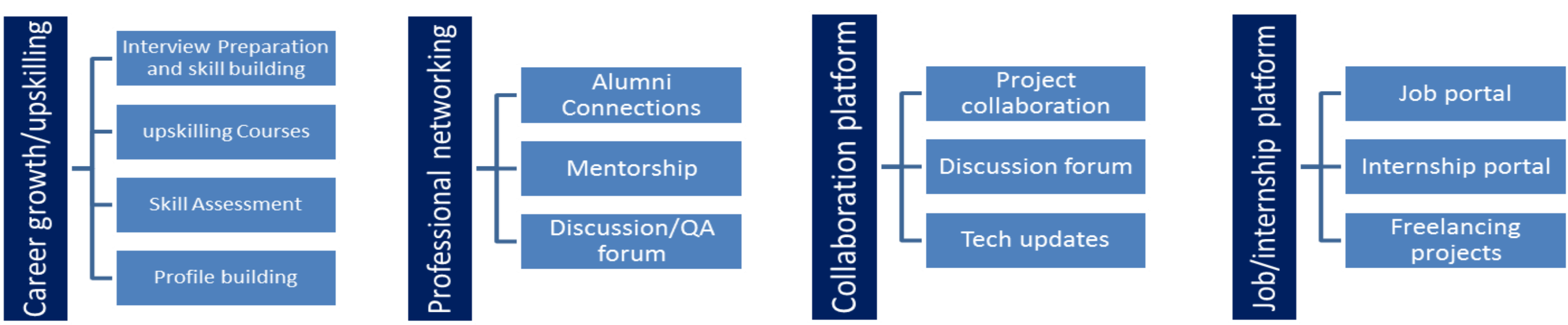
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



Seeing need of upskilling in self paced manner along-with additional support services e.g. Internship, projects, interaction with Industry experts, Career growth Services

upSkill Campus aiming to upskill 1 million learners in next 5 year

https://www.upskillcampus.com/



2.3 The IoT Academy

The IoT academy is EdTech Division of UCT that is running long executive certification programs in collaboration with EICT Academy, IITK, IITR and IITG in multiple domains.

2.4 Objectives of this Internship program

The objectives of this internship program were to:

- Get practical experience working in the industry.
- Solve a real-world problem using machine learning.
- Improve my job prospects and understanding of the field.
- Develop personal skills in problem-solving and data analysis.

2.5 Reference

[1] A. Saxena and K. Goebel (2008). "Turbofan Engine Degradation Simulation Data Set", NASA Ames Prognostics Data Repository, NASA Ames Research Center, Moffett Field, CA

[2] Li, J., Li, X., & He, D. (2019). "A Remaining Useful Life Prognosis of Turbofan Engine Using Temporal and Spatial Feature Fusion", IEEE Access.

[3] Che, C., Wang, H., Fu, Q., & Ni, X. (2023). "Remaining Useful Life Estimation of Aircraft Turbofan Engine Based on Random Forest Feature Selection and Multi-Layer Perceptron", Applied Sciences.

2.6 Glossary

Terms	Acronym
RUL	Remaining Useful Life
EDA	Exploratory Data Analysis
RMSE	Root Mean Squared Error
LSTM	Long Short-Term Memory
RNN	Recurrent Neural Network

3 Problem Statement

The assigned problem was to predict the Remaining Useful Life (RUL) of turbofan engines using time-series sensor data from a fleet of similar engines. The dataset, provided by NASA, consists of multiple multivariate time series for four different subsets, each representing different operating conditions and fault modes. The goal was to build a model that could take the sensor data from an engine in operation and predict how many cycles it had left before failure.

4 Existing and Proposed solution

Existing solutions for this problem often involve either traditional statistical models or more advanced machine learning techniques. While simple models can provide a baseline, they often struggle to capture the complex, time-dependent patterns in the data, especially when multiple operating conditions and fault modes are present.

My proposed solution was a two-pronged approach:

1. First, to establish a strong baseline using a **Random Forest Regressor**, a powerful and interpretable classical model.
2. Second, to implement a more advanced **Long Short-Term Memory (LSTM) network**, a type of recurrent neural network (RNN) specifically designed to learn from sequence data.

The value addition of this approach is the direct comparison between the two models, which demonstrates the trade-offs between simplicity and performance and highlights the LSTM's superior ability to handle the complexities of time-series data with multiple failure modes.

4.1 Code submission (Github link)

https://github.com/agamyaaa14/upskillcampus

4.2 Report submission (Github link)

https://github.com/agamyaaa14/upskillcampus/blob/main/reports/Predictive_Maintenance_Turbofan_Engine_Agamya_USC_UCT.pdf

5 Proposed Design/ Model

The design flow of the solution followed a standard machine learning pipeline:

1. **Exploratory Data Analysis (EDA):** For each of the four datasets, the sensor data was visualized to identify trends. The standard deviation of each sensor was then calculated to programmatically identify and remove "useless" sensors with little to no variance.
2. **Feature Engineering:** The target variable, Remaining Useful Life (RUL), was calculated for the training data by finding the maximum cycle for each engine and subtracting the current cycle.
3. **Preprocessing:** All input features were normalized using **Min-Max Scaling** to a range of [0, 1]. For the LSTM model, the target variable (RUL) was also scaled, and the data was reshaped into sequences.
4. **Model Training:** Both the Random Forest and LSTM models were trained on the preprocessed data for each of the four datasets.

6 Performance Test

This section addresses the performance of the proposed models against key industrial constraints. For a predictive maintenance solution to be viable, it must not only be accurate but also computationally efficient.

Identified Constraints:

- **Accuracy:** The primary constraint. The model's predictions must be close enough to the true RUL to be useful for maintenance scheduling. The key metric is Root Mean Squared Error (RMSE).
- **Computational Speed (MIPS):** This relates to the time required to train the model and, more importantly, to make a prediction (inference). A model that is too slow to train is difficult to iterate on, and slow inference is not practical for real-time monitoring.

6.1 Test Plan/ Test Cases

The test plan involved evaluating both the Random Forest and LSTM models on the unseen test data for all four datasets (FD001, FD002, FD003, and FD004). The performance was measured by comparing the predicted RUL against the ground truth RUL values.

6.2 Test Procedure

For each dataset, the models were trained on the full training set. Predictions were then made on the final available data point for each engine in the corresponding test set. The list of predictions was then compared against the true RUL values to calculate the final RMSE.

6.3 Performance Outcome

The results demonstrate a clear trade-off between model complexity, accuracy, and training time.

Dataset	Conditions	Fault Modes	Random Forest (RMSE)	LSTM (RMSE)
FD001	1	1	34.60	80.64
FD002	6	1	34.58	37.55
FD003	1	2	50.42	14.39
FD004	6	2	42.92	40.65

- **Accuracy Constraint:** The LSTM model demonstrated superior accuracy on the FD003 dataset, which featured multiple fault modes. Its RMSE of **14.39** is a very strong result and shows its ability to handle complex failure patterns, a key requirement for a real-world system. The Random Forest was more accurate on the simpler FD002 dataset.
- **Speed Constraint:** The Random Forest model was significantly faster to train, with training times of just a few seconds per dataset. The LSTM model, being a deep learning architecture, required several minutes to train for 20-30 epochs. However, for inference (making a single prediction), both models were extremely fast and suitable for real-time application.

The results indicate that while the Random Forest is a strong and fast baseline, the LSTM model is the superior choice for handling the more complex and realistic scenarios involving different types of engine failure, justifying its longer training time.

7 My learnings

Throughout this internship, I gained several valuable skills and insights:

- I learned the importance of a structured, end-to-end machine learning workflow, from data exploration to model comparison.
- I gained practical experience in feature engineering and preprocessing for time-series data.
- I learned that the "best" model is highly dependent on the data's complexity and that a more complex model like an LSTM doesn't always perform better without careful tuning.
- The project highlighted the importance of establishing a strong baseline model to make data-driven decisions about more advanced approaches.

8 Future work scope

While the project was successful, there are several avenues for future work:

- **Hyperparameter Tuning:** A more exhaustive search for the optimal hyperparameters for both the Random Forest and LSTM models could lead to further performance improvements.
- **More Advanced Architectures:** Exploring more complex deep learning architectures, such as GRUs or attention-based models, could potentially yield even better results.
- **Feature Engineering:** More advanced feature engineering, such as creating rolling averages or other time-based features, could provide the models with more predictive signals.