Data Driven Mission Readiness

PURDUE UNIVERSITY | The Data Mine

Raytheon Technologies

Pierce Warburton - Gahn Mungarndee - Anders Gilliland - Matthew Beringer - Rugved Dikay - Ivan Esparza - Delaney Elder - Kapil Manicka - Christopher Meza - Gabriel Muzio - Rishab Pangal - Luis Salazar - Jack Secor - Simay Savas - Shaynoah Bedford

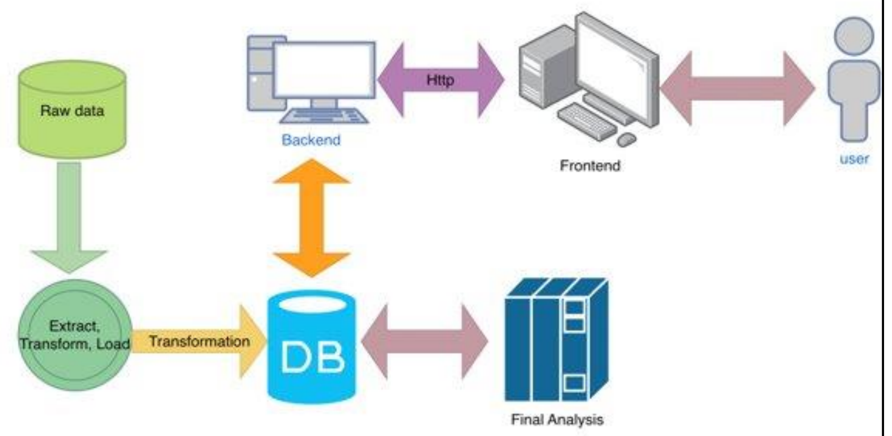
Project Introduction

RUL Prediction for Improved Operational Capability

Our goal for this year was to develop models to predict remaining useful life (RUL) and ingest run-to-failure data for turbofan engines. Predicting RUL improves operation capabilities by reducing downtime of equipment and mitigating catastrophic failures. In order to develop a robust enterprise solution to ingest sensor data, transform it, and feed it into neural networks for RUL prediction we considered the NASA CMAPSS Jet Engine Simulated Data. The high-level architecture of our solution is shown in the diagram below.

NASA Turbofan Dataset

The NASA CMAMPSS Jet Engine dataset describes the lifecycle of a turbofan engine through hundreds of computer-simulated engines run to failure. The data comes from 24 sensors that record different parameters as the engine runs. The data also included Gaussian noise added to the sensors, increasing in severity throughout three batches of simulations.

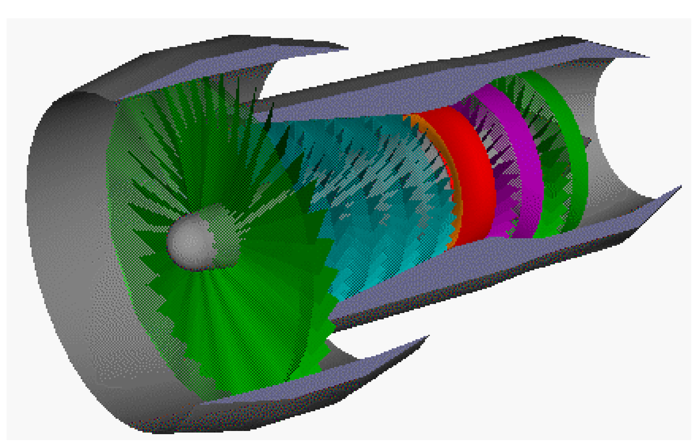


Enterprise Solution

Pipeline Overview

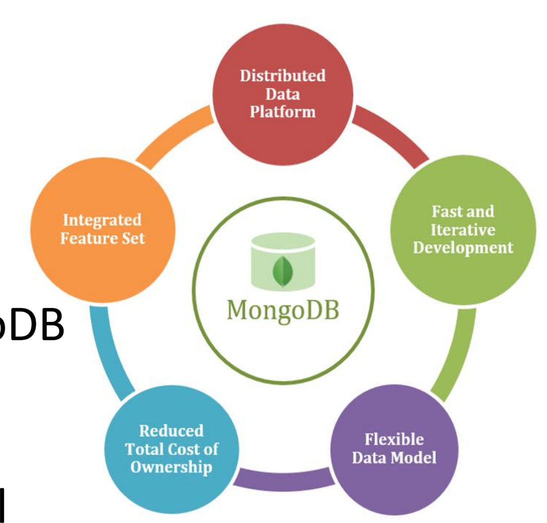
The enterprise solution is comprised of a data pipeline capable of ingesting data, processing it through ETL (Extract, Transform, Load) methods, storing it in a secure and accessible data lake, and finally cleaning or transforming the data to best support future analysis. Each step has been independently developed and optimized to ensure that the pipeline is robust. While brief testing of the entire system was undertaken the analysis detailed in the RUL section was given priority and the full assemble of this pipeline left to future work.

ETL (Extract Transform Load)



ETL consisted of refining and loading raw data into a more usable, organized format. Specifically, a focus was placed on first structuring the raw data into labeled, CSVs, and then "cleaning" the data into files free of null values or gaps. Transformation was also explored through applying physics equations (isentropic flow, combustion) to identify anomalies in the data compared to theoretical turbofan engine function. This combated both noise and incorrect values.

Database

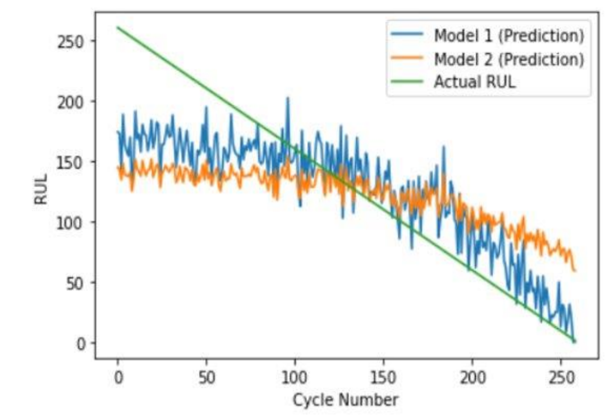


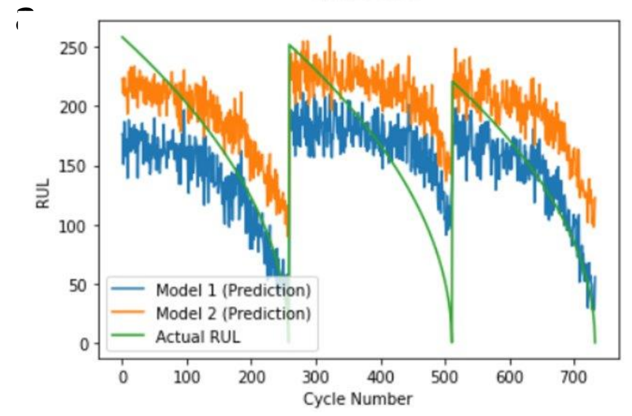
After the ETL process the data was in a format easily uploaded to our MongoDB database. This database was hosted on the Geddes cluster by deploying a workload that contained a MongoDB image which has improved security, scalability, and efficiency over other options. The selection of a nonrelational NoSQL database allowed for large amounts of unstructured data to be hosted.

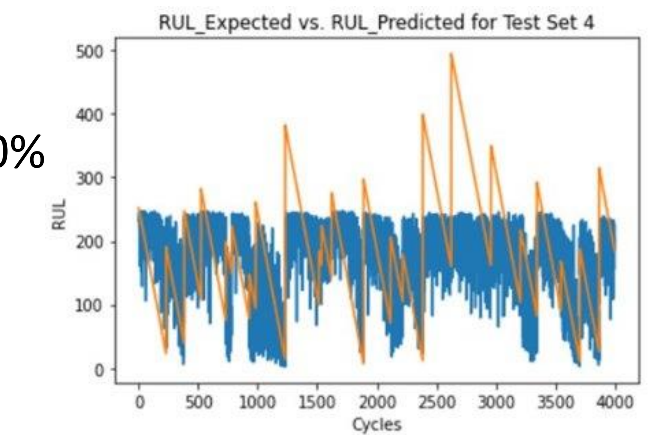
RUL Prediction

LSTM (Long Short-Term Memory)

LSTMs are a powerful regression technique that accounts for past trends in data to make future predictions. In this project, we develop an LSTM to directly compute RUL for an engine based upon specific sensors identified with PCA. Two methods of degradation, and thus two models, were considered, linear degradation and a differential degradation. The Linear Model yielded a 65% accuracy of RUL prediction with the Differential Model surpassing it with close to 80% accuracy. An alternative method simultaneously explored was an engine health classifier, intended on determining if an engine was "healthy" or "faulty" to support more complex computation of an LSTM model downstream. This method had a relatively moderate accuracy of 65-70% but high error for direct RUL computation.

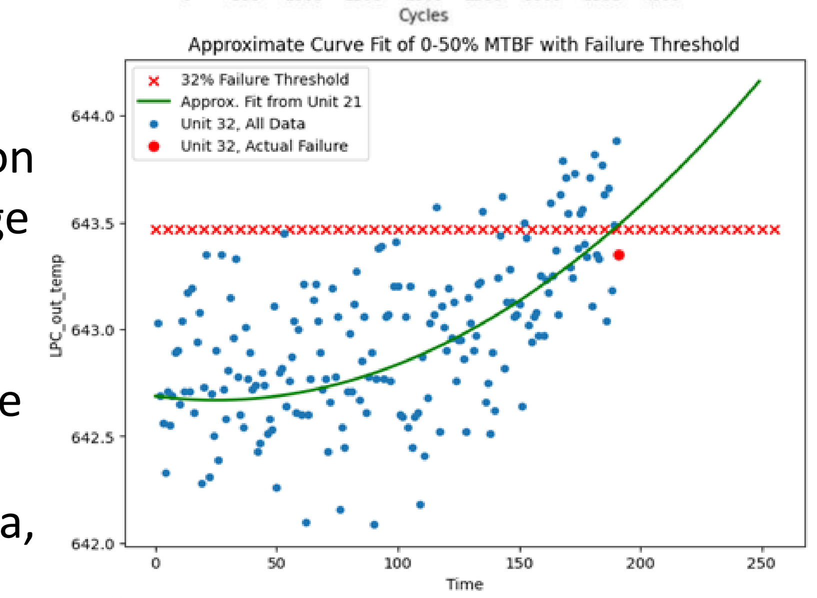


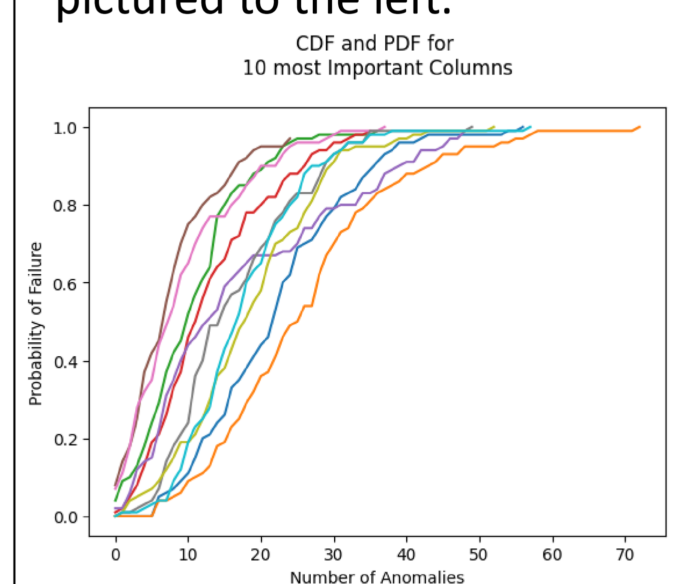




Regression Curves

Using PCA (Principal Component Analysis), we found that all failures occurred in a specific range of sensor values. From this, we made a regression analysis of sensor data. Given the range of "dangerous" sensor values, we placed an arbitrary threshold where 32% of units had failed. From there, we curve fitted the data to predict future sensor values from portions of the data, pictured to the left.





Clustering

Using DBSCAN for unsupervised anomaly detection we developed sensor specific CDFs for the probability of failure which allowed for a RUL prediction.

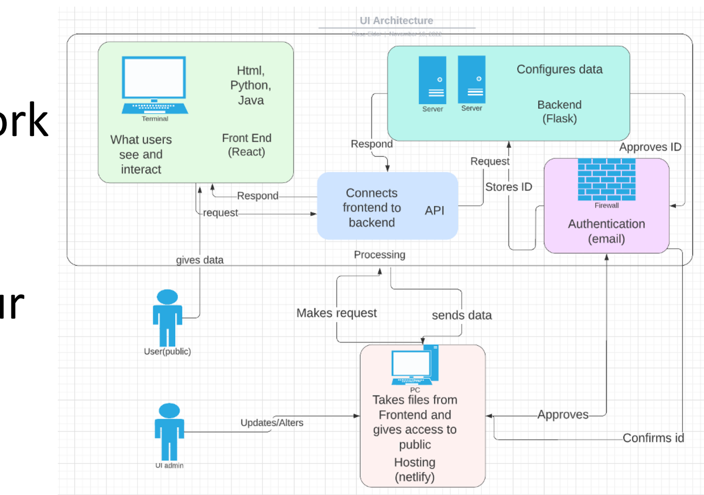
CNNs

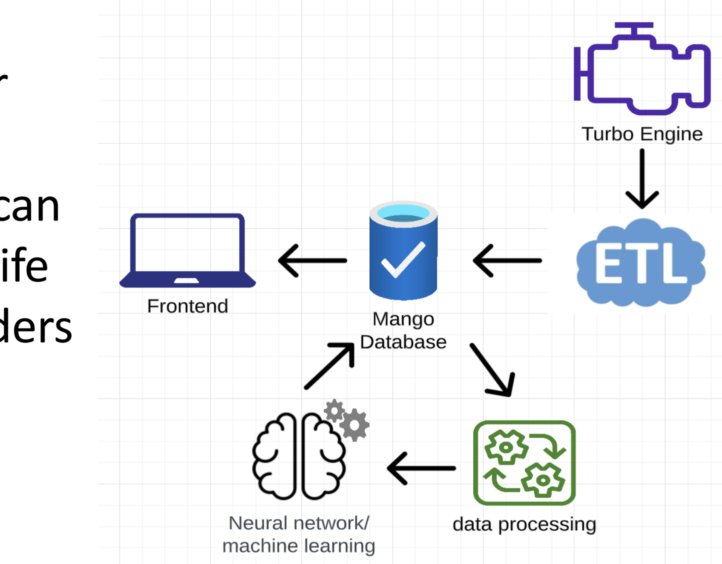
CNNs were employed here to analyze chunks of the data at a time, reducing them too classifiable embeddings. While a classification approach garnered some results, the regression approach was unsuccessful. This was a combination of poor pre-processing of the data from time series to images and inadequate model architecture

User Interface

Website

Our website is a medium to concentrate our work into a palatable format that is accessible to a nontechnical customer. The site is open to the public so anyone who wishes to view it may. Our UI architecture is designed to accept new data from the user, securely connect the user to the pipeline, and use processed data to display data visualizations. We also have encryption implemented and did some research into user authentication, leaving the selection and implementation of it for future work. The UI can display our work on finding remaining useful life in cohesive graphs that will help our stakeholders make better judgments about scheduling maintenance, optimize mission efficiency, and help avoid downtime.





Future Work

Enterprise Solution Future Work

To compose a unified solution the team plans to restructure individual processes into a singular data pipeline capable of preforming all the steps from ingestion to RUL prediction autonomously This will include hosting it on the Data Mine's resources to fully automate the ETL process and working with the website to display visualizations throughout the process

RUL Prediction Future Work

To enhance RUL prediction, the successful LSTM and classifier models will be expanded through continued hyperparameter turning and architectural improvements through research and testing. Moreover, a Physics Integrated Neural Network (PINN) will be developed to simulate ideal engine behavior, determine more detailed degradation functions,

Tools/Libraries

The primary tool used were Python through Anaconda environments.. The libraries utilized were Pandas, TensorFlow, Matplotlib, and Numpy. The database was constructed using MongoDB and Docker while the UI was built using React, Flask, and Netlify hosting.

Acknowledgements

We'd like to acknowledge and give our warmest thanks to Raytheon, specifically our mentor Michael Douglass, and the Data Mine Staff, Dr. Ward and Margaret Betz for their constant support.

The Data Mine Corporate Partners Symposium 2023