Raytheon Team 1

Pierce Warburton, Gahn **Mungarndee, Jack Secor, Sharath Jegannathan, Bernie Gandin, Jason Vo, Siddharth Rao, Eric Barnfather**

PURDUE UNIVERSITY | The Data Mine

Raytheon Technologies

Introduction

The overarching aim of this project has been to calculate the remaining useful life and predict part failure for a turbofan engine. Looking at Nasa Prognostic Turbofan Data and Purdue Solar Panel Data we worked to establish a pipeline to ingest, transform, and analyze data. The Turbofan data was multiple run to failure tests on virtual large aircraft engines while the Solar Panel data was simply power output and axillary data from solar panels mounted on the roof of KNOY here at Purdue. The work we have completed at this point is the ETL process (stands for Extract, Transform, and Load), the data lake (where the unaltered data is stored), an initial data analytics report, and some anomaly detection although incorporating it within the existing pipeline remains firmly in future work.

API Integration

The API team developed interfaces to access weather, regional power generation, and simulated solar data. These data are sourced on a weekly schedule to ensure a constant new data source for analysis. They are automatically ingested into the database.

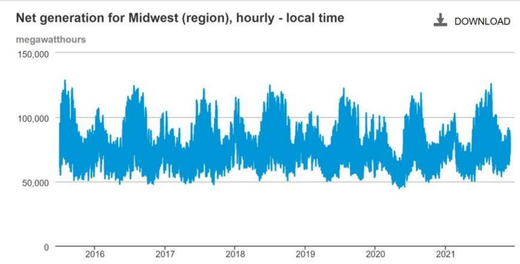

Figure 1: Net Power Generation for Midwest (EIA)

ETL Function

The ETL team was focused on creating functions to allow conversions of raw data to usable formats and eventually loading them to our database. We incorporated features such as:

- Compatibility with h5, .mat, csv, JSON, and txt files
- Rudimentary error handling
- Automation for ingesting data and generating analytical reports
- Event logs for loading, extracting, and errors

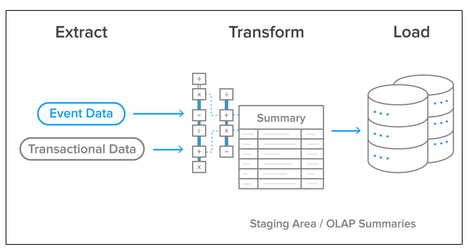

Figure 2: Example of the ETL process

Data Analysis Report

As part of the Data Analytics sub-team, our goal was to create an initial data analytics report that could be loaded into the database for the customer to view. Our goals and requirements were to make a generic report that the customer could access to get a clear overview of the data. With the conditions set, we had to use various python libraries such as Pandas, Matplotlib, and Seaborn to analyze, plot, and share data with the customer. However, we did not stop there; we went further and created static HTML websites for the simplicity of our customers. The HTML Website does not include any code and is a report with various sections of graphs and plots formatted for readability. A new HTML website file is generated and added to the database every day. All prior HTML websites are also available within our database. In addition, every HTML website is formatted to include the date and report name.

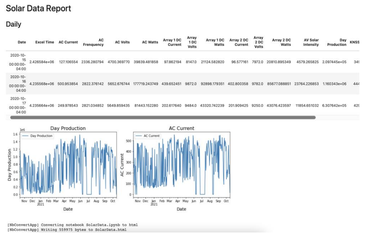

Figure 3: Static HTML Website

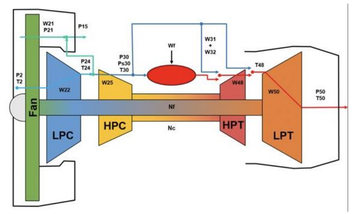

Figure 4: Turbofan Engine

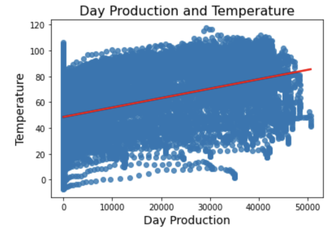

Figure 5: Seaborn Solar Data Plot

Database Selection

We explored several different database solutions.

- SQLite – Abandoned for limits of the serverless database file
  - Challenging to collaborate on – no user permissions.
  - OS file permissions not intuitive for pipeline
- MongoDB – Selected for simplicity and power
  - Much more efficient for implementing to pipeline (cloud-based vs file based)
  - Powerful authentication system
  - Intuitive API to for easy access when conducting data analysis

Infrastructure

Our goal was to implement all the pipeline pieces into a cohesive "enterprise" solution with multiple interfaces to ingest data and an intuitive dashboard to review the processed data.

- Batch job approach unviable – we needed the ability to host services (MonogDB, web server) instead of pure computing
- Switched to Docker – an OS-level virtualization platform-as-a-service solution.
  - Ability for web-powered services – currently hosting MongoDB
- Focus shifted from deploying pre-existing artifacts to developing our own throughout the year.

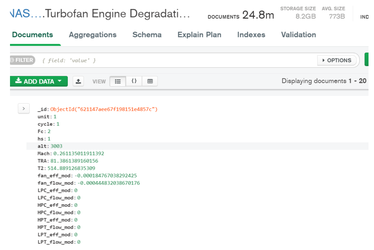

Figure 6: MongoDB populated with turbofan data

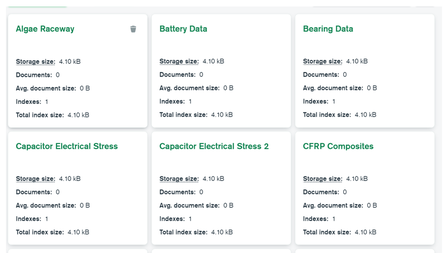

Figure 7: MongoDB populated with solar data

Infrastructure Solution

Our final product utilized MongoDB hosted on the Geddes cluster with the follow features.

- User authentication
- Easy CRUD and appending functionality
- Synthesis of ETL function and Python Flask App
- Integration of API function and Data Analytics Report
- Five NASA Prognostic datasets
  - Including three datasets of the aforementioned Turbofan data

Python Flask App

In order to create a more customer friendly option for the ETL process we developed a Python Flask web application capable of parsing input files. This web application allows authorized users to drop data into the database in a simpler way than through the ETL function. It supports both csv and .mat files

Behind the scenes the code is identical to the ETL function, this flask app simply improves the User Experience. Our next steps are to add interactive bootstrap components such as a toolbar and submission button to make the application more accessible to the end-user.

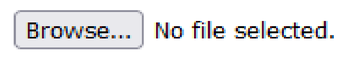

Figure 8: Python Flask App

Anomaly Detection

Anomaly detection consists of a series of statistical analysis to calculate and log outliers and anomalies within a raw data signal. We used a sliding window analysis on the Turbofan Data set from the NASA Prognostics Repository. We explored using different statistical moments such as Mean, Variance, Skewness, and Kurtosis. Through these analyses the team were able to isolate several sensors that show promise for future Remaining Useful Life calculations, such as SmFan and Wf.

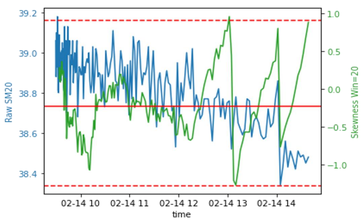

Figure 9: Turbofan Skewness Analysis

Future Work

Our future goals are to focus on improving our anomaly detection by evaluating which parts constitute each failure. This would help us evaluate which anomalies are more indicative of failures. We also wish to incorporate Principal Component Analysis (PCA). PCA reduces the dimensionality of datasets and increases interpretability but at the same time minimizes information loss. This will help us train the machine learning model to predict when an anomaly will occur.

Acknowledgments

We'd like to thank the good people at Raytheon, specifically our Mentor Mike Douglass, the always helpful Data Mine Staff Dr. Ward, Heather Goodwin, and Maggie Betz, and Purdue Professor William Hertz for allowing us to gain access to the Solar Data