

From Black-Box Models to Actionable Insights: Integrating LLMs with SHAP for Predictive Maintenance

Antonios Vozikis
Student ID: 2801404
MSc AI Vrije Universiteit Amsterdam

First version, June 1, 2024 – Current version: October 7, 2024

Contents

1	Introduction	4
1.1	Large Language Models	4
1.2	LLMs and Sensor Data	4
1.3	Scope of the Work	5
1.4	Problem Statement	5
1.5	Objectives	5
1.6	Structure	5
2	Related Work	6
2.1	Large Language Models in Natural Language Processing	6
2.1.1	Evolution of Language Models	6
2.1.2	LLMs Beyond NLP	6
2.2	Anomaly Detection in Sensor Data	6
2.2.1	Machine Learning Approaches	6
2.2.2	Deep Learning Models	7
2.3	Model Interpretability and SHAP Values	7
2.3.1	The Need for Interpretability	7
2.3.2	SHAP Values for Explainable AI	7
2.4	Combining Interpretability with LLMs	7
2.4.1	Bridging the Gap Between Models and Actionable Insights	7
2.4.2	Existing Work on LLMs for Interpretability	8
2.5	Predictive Maintenance and Battle Damage Repair	8
2.5.1	Challenges in High-Stakes Environments	8
2.5.2	Integrating SHAP and LLMs for Enhanced Decision-Making	8
2.6	Gaps in the Literature and Novel Contributions	8
2.6.1	Limitation of Current Approaches	8
2.6.2	Our Work's Unique Positioning	8
3	Dataset and Experimental Data Analysis	9
3.1	Dataset	9
3.1.1	Data Preprocessing	11
3.2	Experimental Data Analysis	11
3.2.1	Engine Count and Cycles Before Failure	11

4	Methodology	14
4.1	Overview of the Proposed Approach	14
4.2	Data Preprocessing	14
4.2.1	Process Data	14
4.2.2	Normalization	14
4.2.3	Data Splitting	16
4.3	Model Development	16
4.3.1	Model Architectures	16
4.3.2	Training and Evaluation	20
4.4	Model Interpretability	20
4.4.1	SHAP Values	20
4.5	Integration with Large Language Model	21
4.5.1	Pretrained Transformer Models	21
4.5.2	Generating Output	21
5	Results	23
5.1	Methodology Recap	23
5.2	Results	23
5.3	Model Performance	23
5.3.1	RNN model	24
5.3.2	LSTM	25
5.3.3	xLSTM	27
5.3.4	Decoder only Transformer	28
5.4	Interpretability Analysis with SHAP Values	29
5.4.1	SHAP Analysis	29
5.5	Integration with Large Language Models	30
5.5.1	LLM Reasoning and Next Steps	30
5.5.2	Instance 1	31
5.5.3	Instance 2	31
5.5.4	Instance 3	31
5.5.5	Instance 4	32
6	Discussion	33
6.1	Discussion	33
6.1.1	Integration of LLMs in Predictive Maintenance Workflows	33
6.1.2	Anomaly Detection and Model Interpretability	33
6.1.3	Key Findings from SHAP Analysis	33
6.1.4	Leveraging LLMs for Actionable Insights	34
6.1.5	Comparison of LLM Responses	34
6.1.6	Advantages of LLMs in High-Stakes Environments	34
6.1.7	Generalization to Other Domains	34
6.2	Future Work and Considerations	34
7	Conclusion	36

A	Appendix	40
A.1	Appendix	40
A.1.1	Instance 1	40
A.1.2	Instance 2	45
A.1.3	Instance 4	56

Chapter 1

Introduction

1.1 Large Language Models

Large Language Models (LLM) are a subgroup of Artificial Intelligence (AI) and the latest revolution in the field of Natural Language Processing (NLP). These kind of models are a big leap forward on how machines are able to understand and also generate human language based on a user generated prompt. The main use cases for NLP include text generation and summarizing, sentiment analysis, opinion mining and machine translation. Diving deeper, the underlying architecture of LLMs relies on transformers, which are networks that use self-attention mechanisms to process and generate text based on patterns learned from vast amounts of data. These models are pretrained on large text corpora and later finetuned for specific task to enhance their accuracy in applications of interest. Generative pretrained models (GPT), due to the enormous amount of data they have been trained on, are able to demonstrate a form of reasoning, draw conclusions and create a plan on their own on how to tackle any task at hand. Being capable of this, makes them incredibly versatile and opens up a broad spectrum of potential use cases in different sectors. While this is remarkable at first glance, their "reasoning" is pattern-based and statistical, relying on the correlations observed in the training data rather than true understanding or logical deduction. With that in mind, it is very crucial to set realistic expectations about their capabilities. The most important innovation that helped promote the field's growth was the transformer architecture, which has found many use cases in fields other than NLP, like Computer Vision, Representation Learning, Reinforcement Learning and Generative AI. These applications further demonstrate the flexibility and power of transformers in various computational tasks.

1.2 LLMs and Sensor Data

One interesting use case for LLMs includes predictive maintenance and battle damage repair, a situation that has to do with the quick assessment and fixation of critical damages on a vehicle (aircraft, vessel, etc.) during combat with the aim of keeping or restoring its operational capabilities. In such a specific use case, huge potential could be seen in how LLMs could support real-time decision-making in high-stakes settings. This, in essence, requires access to various sensors on the vehicle that deliver data about its current state in as much detail as possible. After that is done, there is a need to implement an anomaly detection framework in a way that any engine failure, for instance, or any problem occurring on the vehicle can be immediately detected. These systems have to work with a lot of precision and accuracy for the continued operation of the vehicle.

Unfortunately, while it is possible to detect the anomaly, black-box models like neural networks and transformer-based architectures themselves cannot identify which sensor is to blame for the error. This is one major challenge in diagnosis to find the root cause of the issue at hand and taking necessary steps for rectification. That is where SHAP-an acronym for SHapley Additive exPlanations- starts to become really valuable. Applying SHAP to these black-box models makes them interpretable, since it quantifies the contribution of every single sensor regarding an anomaly detected by the model, hence informing the model on which sensor is causing the problem. This can be conducive to interpretability for informed repair decisions.

1.3 Scope of the Work

In this work, we cover the use of traditional statistical analysis combined with LLMs to present an output that will be human-readable and understandable, guiding the operator through the steps available for resolving any given issue. Our key goal is not just to improve the usability and clarity of outputs in a rapidly changing environment, but also to bridge the gap between the complex models' output and the actionable steps for the operator that can be taken to prevent issues or make immediate fixes. We aim to offer operators an intuitive and more effective tool under highly dynamic, challenging conditions.

1.4 Problem Statement

While both neural networks and transformers are increasingly used in the study of anomaly detection, one major gap still remains: the satisfactory identification of which sensors are responsible for these anomalies, coupled with providing actionable solutions to the operators. To the best of our knowledge, no prior framework couples interpretability with the large language model's ability to reason and generate instructions to repair or even prevent the problem before it arises. This gap inhibits effective real-time decision-making in contexts that are critical, such as predictive maintenance and battle damage repair.

1.5 Objectives

The key objectives of this research are:

- Using a Machine Learning or Deep Learning model to predict if the next failure is about to happen in the next cycles.

- Integrating SHAP values with black-box models for enhanced interpretability in sensor data analysis.

- Utilizing LLMs to produce human-friendly outputs that will help the operators in guiding their resolution of identified issues. This framework's main goal is to enhance decision-making in a high-stakes environment.

1.6 Structure

This thesis is organized as follows:

Relevant literature with regards to LLMs, SHAP values, and anomaly detection frameworks is reviewed in Chapter 2. The dataset and methods of the experimental data analysis are described in Chapter 3. Chapter 4 describes the methodology regarding data processing, model development, and integrating the code of the SHAP values with LLMs. Chapter 5 describes the results of our experiments and analyses. Chapter 6 presents the implications of findings, limitations of the study, and suggestions for future research. Chapter 7 concludes this thesis by summarizing the contributions and key messages.

Chapter 2

Related Work

This chapter presents the related work in the field concerning LLMs, anomaly detection using machine learning and deep learning on sensor data, and explainability of black-box models through techniques like SHAP values. We will situate our research within the broad context by reviewing these contexts and highlight what aspects make this approach unique.

2.1 Large Language Models in Natural Language Processing

2.1.1 Evolution of Language Models

Large language models have revolutionized NLP and put computers in a position where they can understand and generate text like humans. All this was made possible with the introduction of a transformer architecture that allowed models to leverage long-range dependencies in text using self-attention mechanisms. This started with models such as BERT [3] and GPT [12], which achieved state-of-the-art results in most language translation, summarization, and question answering benchmarks.

2.1.2 LLMs Beyond NLP

Although the models in origin were designed for textual data, their architecture began was used in many other domains, such as computer vision [5] and reinforcement learning [2]. The fact that transformers are able to handle different types of sequential data predetermines their suitability for tasks like time series analysis [9] or anomaly detection [7]. The ability of LLMs to generate relevant text opens perspectives for interfacing complex computational outputs with human operators, enables better comprehension and allows for better decision making.

2.2 Anomaly Detection in Sensor Data

2.2.1 Machine Learning Approaches

Anomaly detection in sensory data plays an important role in almost all fields and industries for predictive maintenance and operational safety. More traditional machine learning models, such as support vector machines [13] and random forests [8], have also been used to identify complex patterns and are able to indicate system failures. These models can process large volumes of data to detect deviations from normal

behavior, but often require considerable feature engineering and may not capture temporal dependencies effectively.

2.2.2 Deep Learning Models

Deep Learning models, like for example Recurrent Neural Networks (RNNs) [11] and their variants, such as Long Short Term Memory (LSTM) [6], have been promising in handling sequential data from sensors. LSTMs can capture long term dependencies, hence making them very useful for modeling timeseries data where past events may influence future outcomes. Bidirectional LSTMs extend this capability by processing the data in both forward and backward directions. thus capturing information from the entire sequence [14]. Similarly, Transformers have been used for timeseries analysis [17]. Due to their ability to focus on different parts of the input sequence, transformer based models can be utilized to model complex relationships in time without being limited by the sequential processing as in RNNs.

2.3 Model Interpretability and SHAP Values

2.3.1 The Need for Interpretability

As the complexity of machine learning models continues to grow, it becomes important to understand decision making processes, particularly in applications for which critical consequences are at stake, such as predictive maintenance. While black box models are formidable, they shed little light on how input signals are transformed into output signals. Such transparency may inhibit trust and reduce practical utility for such models in real-life applications where an explanation will be necessary for validation and compliance [4].

2.3.2 SHAP Values for Explainable AI

SHAP (SHapley Additive exPlanations) values provide a unified framework for modeling and interpreting, based on values from game theory. Building on the concept of cooperative game theory, SHAP values define a value for each feature for a specific prediction that indicates the contribution of that feature toward the model output. By attributing the prediction to individual input features, SHAP values make it possible to understand which factors are driving the model's decisions. Shap Values can also be applied to sensor anomaly detection models and are able to point out which sensor is the most influential in predicting the failures. This information is very important for diagnosis and focused maintenance [10].

2.4 Combining Interpretability with LLMs

2.4.1 Bridging the Gap Between Models and Actionable Insights

While SHAP values provide enhanced interpretability, they supply their explanations in a somewhat technical format, making very hard for a non expert user to understand and connect the dots into discovery where did malfunction took place or is going to take place in the upcoming future. This is where LLMs come into play, since they can translate such technical information into human readable and actionable insights.

2.4.2 Existing Work on LLMs for Interpretability

Previous work has explored the use of LLMs to generate explanations for model predictions for domains such as healthcare [15] and finance [1]. These efforts focus on summarizing complex data outputs or translating them into natural language explanations. However, the integration of SHAP values with an LLM to generate context specific maintenance instructions is still relatively unexplored.

2.5 Predictive Maintenance and Battle Damage Repair

2.5.1 Challenges in High-Stakes Environments

In scenarios like predictive maintenance and battle damage repair, timely and accurate diagnosis of system failures is critical [16]. Operators need explicit recommendations about how to address problems to maintain functionality. Traditional models will detect an anomaly but cannot provide insights onto the root causes or actionable next steps for remediation.

2.5.2 Integrating SHAP and LLMs for Enhanced Decision-Making

The idea of integrating SHAP values with LLMs will ensure that the model not only can identify an anomaly but also explain such decision in a way easily understandable by operators. This allows generating step by step instructions or recommendations based on how the model interprets sensor data, thus enabling quicker and more effective responses in critical situations.

2.6 Gaps in the Literature and Novel Contributions

2.6.1 Limitation of Current Approaches

Current models in vogue for anomaly detection in sensor data are many times not transparent and provide little explanation for the prediction made. Although SHAP values provide interpretability, they inherently do not translate into actionable guidance for operators. Similarly, LLMs have been used only for general explanations rather than specific context-aware instructions in this context.

2.6.2 Our Work's Unique Positioning

To the best of our knowledge, no prior framework combines model interpretability through SHAP values with the reasoning of LLMs in order to generate maintenance instructions that are actionable and detailed. In this work, we fill this gap by developing a system interpreting the black box models output using LLMs to get human friendly guidance that enhances decision making in high stakes environments.

Chapter 3

Dataset and Experimental Data Analysis

3.1 Dataset

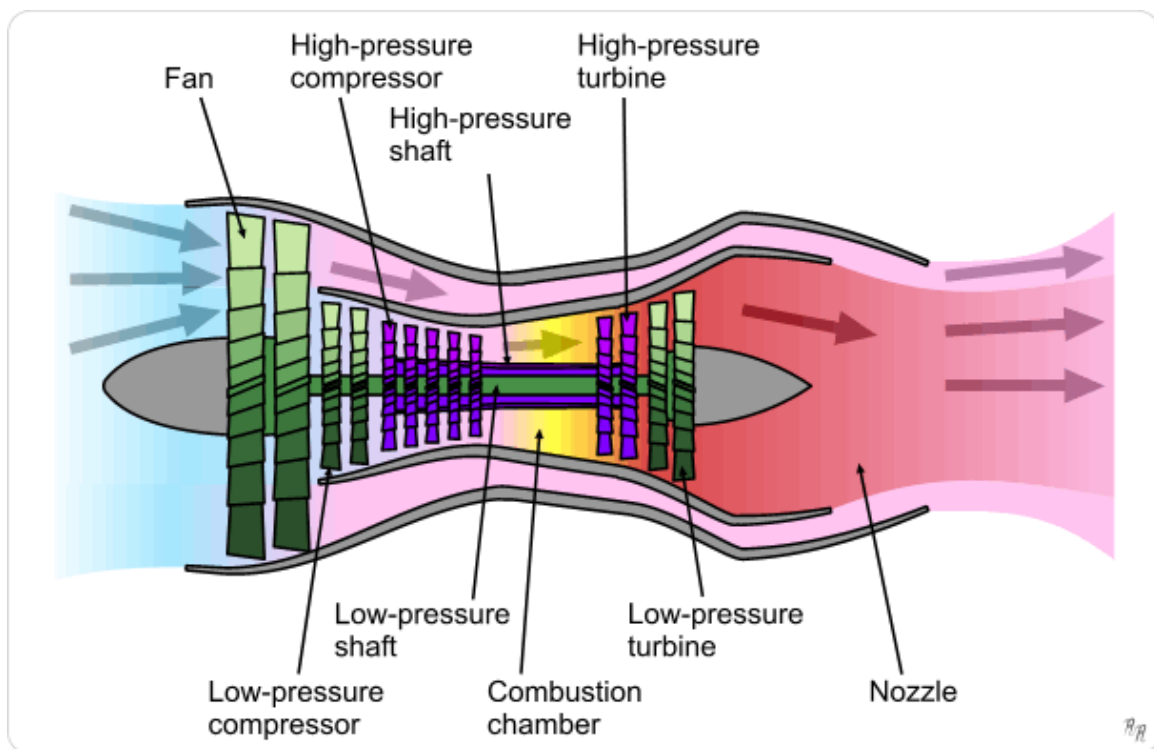


Figure 3.1: Image of the Turbofan from the dataset

[please provide the explicit reference to the dataset – is that the NASA PHM-2008-Challenge dataset?](#)
The dataset consists of multivariate time series. Each dataset is further divided into training and test

subsets. Each time series is from a different engine, but the data can be considered to be from a fleet of engines of the same type. Each engine starts with different degrees of initial wear and manufacturing variation, which are unknown to the user. There are three operational settings that have a substantial effect on engine performance. These settings are also included in the data. The data is contaminated with sensor noise. The engine is operating normally at the start of each time series and develops a fault at some point during the series. In the training set, the fault grows in magnitude until system failure. In the test set, the time series ends some time prior to system failure.

There are 3 different settings for the engine:

- Setting 1
- Setting 2
- Setting 3

and 21 sensor values corresponding to different parts of the engine as shown in Figure 3.1:

Sensor Name	Usage
Fan Inlet Temperature ($^{\circ}\text{R}$)	Measures the temperature of air entering the fan; crucial for understanding thermal efficiency and operating conditions.
LPC Outlet Temperature ($^{\circ}\text{R}$)	Records the temperature of air leaving the Low-Pressure Compressor (LPC); helps in assessing compressor efficiency and detecting overheating.
HPC Outlet Temperature ($^{\circ}\text{R}$)	Measures the temperature of air exiting the High-Pressure Compressor (HPC); vital for monitoring the core engine's thermal state.
LPT Outlet Temperature ($^{\circ}\text{R}$)	Tracks the temperature at the outlet of the Low-Pressure Turbine (LPT); important for evaluating turbine cooling efficiency.
Fan Inlet Pressure (psia)	Measures air pressure at the fan inlet; essential for evaluating the pressure ratio and airflow through the engine.
Bypass-Duct Pressure (psia)	Monitors pressure within the bypass duct; important for understanding the thrust generated by bypassed air.
HPC Outlet Pressure (psia)	Indicates the pressure of air leaving the HPC; crucial for determining compression efficiency and power output.
Physical Fan Speed (rpm)	Measures the rotational speed of the fan; directly related to air movement and thrust production.
Physical Core Speed (rpm)	Monitors the rotational speed of the engine core; critical for managing engine power and detecting abnormalities.
Engine Pressure Ratio (P50/P2)	Ratio of pressures at two points in the engine; key performance indicator for efficiency and thrust generation.
HPC Outlet Static Pressure (psia)	Monitors static pressure after the HPC; helps assess compressor performance and airflow.
Ratio of Fuel Flow to Ps30 (pps/psia)	Calculates the efficiency of fuel usage relative to pressure at station 30; aids in optimizing fuel consumption and performance.
Corrected Fan Speed (rpm)	Provides a normalized measure of fan speed, adjusted for ambient conditions; useful for consistent performance assessment.
Corrected Core Speed (rpm)	Normalized measure of core speed, adjusted for conditions; allows for consistent comparison across operations.
Bypass Ratio	Ratio of bypassed air mass flow to core air mass flow; key indicator of engine efficiency.
Burner Fuel-Air Ratio	Monitors the fuel and air mixture in the combustion chamber; essential for efficient combustion and emission control.
Bleed Enthalpy	Measures energy removed via bleed air; affects overall engine performance and is used for auxiliary systems.
Required Fan Speed	Indicates demanded fan speed for specific operational conditions; used for engine control.
Required Fan Conversion Speed	Similar to required fan speed, adjusted for specific conditions to ensure efficient operation.
High-Pressure Turbines Cool Air Flow	Monitors cooling air flow to the high-pressure turbine; vital for maintaining turbine integrity and preventing overheating.
Low-Pressure Turbines Cool Air Flow	Similar to high-pressure turbine cooling; ensures adequate cooling for the low-pressure turbine.

Table 3.1: Usage of Sensors in NASA Turbofan Engine Dataset

At the same time there is access to the number of cycles that lead the engine to failure, which offers deeper insights into the performance of the engine over time. Especially this value gives us the ability to work with, and analyze the behavior of the engine in both a supervised and an unsupervised manner. Clear benchmarks can be established for normal operation versus abnormal, and at the same time provide a more comprehensive understanding of the patterns (e.g. the sensor combination), leading up to the engine failure.

3.1.1 Data Preprocessing

Before analyzing the data, we performed preprocessing steps to clean and prepare the dataset. Luckily enough, there were no missing values, but normalizing sensor readings and splitting the data into training, validation, and test sets was required. There are 100 instances leading to engine failure for the training set and the validation set, leading to an almost 80-20 training-validation set split. As for the test set, there are already in place 20 instances, not included in the original dataset, which the model will have access to only at the end of the training process in order to calculate the accuracy of the prediction on unseen data. We will get more in depth about the steps we took for data preprocessing on the next chapter, where we outline all the methods used.

3.2 Experimental Data Analysis

The dataset used consists of a set of operational metrics, specially focusing on different engine settings, temperatures, speeds, pressures and other related parameters. The summary statistics of the dataset can be found in Table 3.2, which provides key statistical measurements, including the number of total values, mean and standard deviation, minimum and maximum values per and also the first second and third quartile for each of the parameters.

3.2.1 Engine Count and Cycles Before Failure

Examining each parameter in more detail.:

1. Engine Count and Cycles Before Failure

- **Engine Number:** This contains data from 100 engines with a total of 20361 values. The engines are of the same model or type, but each one represents a different physical unit, subject to it's own set of environmental and operational conditions. Even though all the engines have the same design and specifications, they will experience different wear and tear, maintenance practices and operational loads. As a result, there are differences in how many cycles individual engines complete before failure.
- **Number of Cycles:** Here, the number of cycles refers to how many cycles of operation each of these engines went through before failure. A normal cycle involves the whole procedure until the engine is shut down due to a malfunction. Cycles recorded this way give an indication of the operating life of the engine. Figure 3.2a on the left ranks various engine cases according to the number of cycles before failure in descending order. This shows the variability in the number of cycles various engines survived prior to a failure. Figure 3.2b merges the histogram of the cycles before failure for values close to each other and fits a normal distribution curve for better visualization. The average number of cycles before failure is $\mu = 206.31$ with a standard deviation, $\sigma = 46.11$, indicating a moderate dispersion across the engines. Most of the engines failed after about 200 cycles, although there are a few failing with much fewer

operational lifetimes. The distribution is right skewed, since most of the engines failed within a fairly tight range. Furthermore, several engines completed many more cycles before failure, which indicates that they were above average in durability.

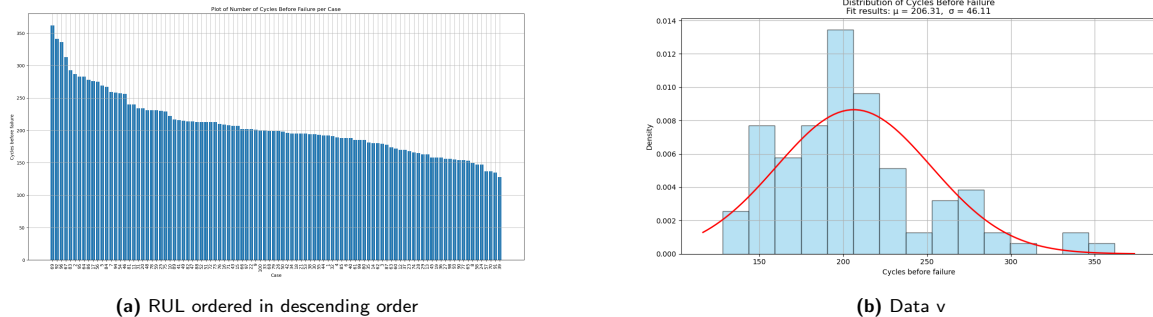


Figure 3.2: Engine Cycles Before Failure

2. Engine Settings These are preset by the operator as part of the operational procedure, right?

- Setting 1 and Setting 2: Their mean values are very close to 0 with very small standard deviation. The quartile ranges indicate very minimal variation, with values around zero.
- Setting 3: It is constant across all datapoints, indicating a parameter, which shows no change in the dataset.

3. Temperature Metrics

- Fan Inlet Temperature: Constant variable 518.67 across all datapoints, implying a controller or a non changing environmental condition for this metric.
- LPC, HPC, LPT OUTLET TEMPERATURES The spread of these temperature variates is as shown: LPC outlet temperature has a SD of 0.50, while that of the LPT outlet temperature takes an SD of 9.00. In the interim, the HPC has the highest mean value of 1590.52 that spreads more with a given standard deviation of 6.13. It, therefore, follows that the high-pressure compressor outlet conditions have variation.

4. Pressure Metrics

- Fan Inlet Temperature: Given that all the data points possess a constant variable value of 518.67, this would hint either towards a controller or some environmental condition that is unchanging for this metric.
- Bypass-Duct Pressure: This variable presents a larger spread; the average is 8143.75 psia, while the standard deviation is 19.08 psia. The minimum value is 8099.94 psia and the maximum 8293.72 psia, implying a very small spread across all dataset.
- HPC Outlet Pressure: The variability within this variable almost does not exist, because of the reading of 8.44 psia and a small deviation of near 0.

5. Speed Metrics

- Physical Fan and Core Speeds: The speed of the physical fan remains the same at 0.03 rpm, while the mean of the physical core speed is at 393.21 rpm with a standard deviation of 1.55 rpm, which is an indication of a slightly more variation in the core speed in comparison with the fan speed.
- Corrected Fan and Core Speeds: These corrected speeds show more variability, and particularly the corrected core speed, with a mean of 8143.75 rpm and a standard deviation of 19.08 rpm.

6. Other Key Metrics

- The Engine Pressure Ratio and Bypass Ratio: Relatively stable on both of the two metrics, with the engine pressure ratio always being 2388; the bypass ratio goes up and down slightly with an average of 8.44.
- Burner Fuel-Air Ratio and Bleed Enthalpy: Burner fuel-air ratio has been kept constant at 0.03; the mean of bleed enthalpy is 393.21 with a standard deviation of 1.55, which indicates that there is not much variation.
- High-Pressure and Low-Pressure Turbines Cool Air Flow: Some variability is shown, with means of 38.82 and 23.29, respectively, and standard deviations indicating slight fluctuations in cooling air flows.

	Count	Mean	Std	Min	25%	50%	75%	Max
Engine Number	20631.00	51.51	29.23	1.00	26.00	52.00	77.00	100.00
Cycles Before Failure	20631.00	206.31	46.11	52.00	104.00	156.00	243.00	362.00
Setting 1	20631.00	-0.00	0.00	-0.01	-0.00	0.00	0.00	0.01
Setting 2	20631.00	0.00	0.00	-0.00	-0.00	0.00	0.00	0.00
Setting 3	20631.00	100.00	0.00	100.00	100.00	100.00	100.00	100.00
Fan inlet temperature (oR)	20631.00	518.67	0.00	518.67	518.67	518.67	518.67	518.67
LPC outlet temperature (oR)	20631.00	642.68	0.50	641.21	642.33	642.64	643.00	644.53
HPC outlet temperature (oR)	20631.00	1590.52	6.13	1571.04	1586.26	1590.10	1594.38	1616.91
LPT outlet temperature (oR)	20631.00	1408.93	9.00	1382.25	1402.36	1408.04	1414.56	1441.49
Fan inlet Pressure (psia)	20631.00	14.62	0.00	14.62	14.62	14.62	14.62	14.62
bypass-duct pressure (psia)	20631.00	8143.75	19.08	8099.94	8133.25	8140.54	8148.31	8293.72
HPC outlet pressure (psia)	20631.00	8.44	0.04	8.32	8.41	8.44	8.47	8.58
Physical fan speed (rpm)	20631.00	0.03	0.00	0.03	0.03	0.03	0.03	0.03
Physical core speed (rpm)	20631.00	393.21	1.55	388.00	392.00	393.00	394.00	400.00
Engine pressure ratio(P50/P2)	20631.00	2388.00	0.00	2388.00	2388.00	2388.00	2388.00	2388.00
HPC outlet Static pressure (psia)	20631.00	100.00	0.00	100.00	100.00	100.00	100.00	100.00
Ratio of fuel flow to Ps30 (pps/psia)	20631.00	38.82	0.18	38.14	38.70	38.83	38.95	39.43
Corrected fan speed (rpm)	20631.00	23.29	0.11	22.89	23.22	23.30	23.37	23.62
Corrected core speed (rpm)	20631.00	8143.75	19.08	8099.94	8133.25	8140.54	8148.31	8293.72
Bypass Ratio	20631.00	8.44	0.04	8.32	8.41	8.44	8.47	8.58
Burner fuel-air ratio	20631.00	0.03	0.00	0.03	0.03	0.03	0.03	0.03
Bleed Enthalpy	20631.00	393.21	1.55	388.00	392.00	393.00	394.00	400.00
Required fan speed	20631.00	2388.00	0.00	2388.00	2388.00	2388.00	2388.00	2388.00
Required fan conversion speed	20631.00	100.00	0.00	100.00	100.00	100.00	100.00	100.00
High-pressure turbines Cool air flow	20631.00	38.82	0.18	38.14	38.70	38.83	38.95	39.43
Low-pressure turbines Cool air flow	20631.00	23.29	0.11	22.89	23.22	23.30	23.37	23.62

Table 3.2: Summary Statistics of the Dataset showing the mean, standard deviation, min max and first, second and third Quartile

Chapter 4

Methodology

In this chapter, we present the methodologies used in our research, from data preprocessing through model development and interpretability techniques, to the integration of models with large language models to produce actionable outputs.

4.1 Overview of the Proposed Approach

Figure 4.1 illustrates the procedure from data to output. Initially some data processing tasks need to be performed in order for it to be useful and ready to be fed to the model in the next step. The model can be trained to perform a regression task, predicting the number of engine cycles before failure or a classification task, splitting the data into chunks and giving the freedom to the model to understand if a chunk will lead to failure or not, based on historical data. The next section will provide more information on the necessary steps taken in order to complete this process. As already known, most of the Deep Learning models are considered black boxes, which means there is no access to the inner dynamics of the network. In order to have some form of interpretability SHAP values are utilized in order to find out which of the sensors contributes to the output of the model. Once the model has a good understanding of which sensors contributed the most to the failure, the data is then fed to a pretrained LLM, which is going to give the motivation behind to what caused and lead the engine to malfunction, and therefore provide the operator with a message on steps to be taken in order to fix, or even prevent, the problem before it arises and creates an issue.

4.2 Data Preprocessing

4.2.1 Process Data

Since we already have the data in place we need to take some steps forward and perform some neat tricks that can lead us to a desired output.

4.2.2 Normalization

Normalization is one of the most important activities for data preprocessing, especially in Machine Learning models, because it allows various features to fall into a similar scale. Sensor data may come from different sources and their ranges differ. This is why the magnitudes of those features, like the sensor

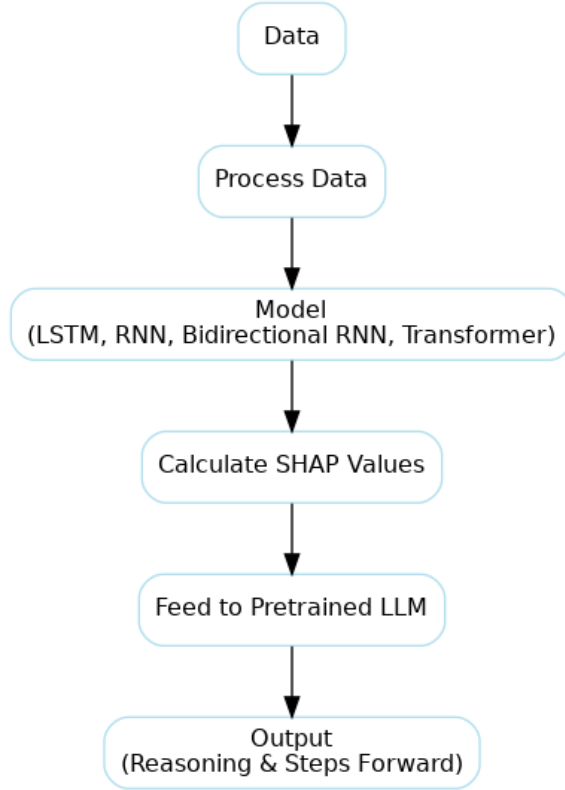


Figure 4.1: Flowchart of methods

values, are different from each other, thus the model can be biased toward learning from particular features only, ignoring others. Normalization removes bias, giving equal importance to each of the features, without allowing larger ranges and values to dominate. Furthermore, most machine learning algorithms, especially gradient-based models, work efficiently if their features are on similar scales. Models like neural networks converge faster and more accurately when the data used for training is normalized.

In our scenario, we will be using the interquartile range (IQR) normalization, a very robust measure of statistical dispersion, which is less sensitive to outliers compared to the more standard scaling techniques, like for example MinMax scaling or Z-score normalization.

The IQR is calculated as the difference between the 3rd quartile (Q3) and the 1st quartile (Q1):

$$IQR = Q3 - Q1$$

This provide us with the range of the middle 50% of the data, ignoring outliers. To normalize a data point x , we can subtract the first quartile and then divide by the IQR:

$$x_{\text{norm}} = \frac{x - Q1}{IQR}$$

This formula ensures that each feature is scaled based on its spread, making the data more comparable across different features.

IQR's robustness against outliers arises because it focuses on the central 50% of the data, between Q1 and Q3. This makes it suitable for sensor data, which may show extremities quite often either because of malfunctions or due to noise.

4.2.3 Data Splitting

Because we didn't have access to the Δt value of the dataset, meaning that we don't know how often the readings from the sensors were saved, a wise choice was to split the dataset into chunks and then, and use the model in a way that predicts if the chunk will lead to an engine malfunction or not. This methodology is very close to the sliding window, which involves moving, or even better phrased sliding, a fixed-size window across a data sequence to generate overlapping segments, which can then be used to create features that capture temporal patterns and dependencies within the data. At the same time, it helps extracting meaningful information such as trends, seasonal patterns, and auto-correlations. When sliding the window over a specific time frame, we can perform different kinds of data aggregation, like mean, max, min or median.

4.3 Model Development

Since we are dealing with multivariate timeseries data a thoughtful approach is to use a model that takes advantage of the time dependencies. We train a model to take as input a temporal window of timeseries data. This is matched by the model against the learned patterns from the training dataset to classify this input window as anomalous or normal (In other words, leading or not to an engine failure within the specified window). The anomalies detected in the dataset are interpreted as predictive indicators of engine failures that are foreseen to occur in some period in the future. There are many models available, but the state of the art includes LSTMs, bidirectional RNNs and transformer based architecture. In our scenario we are going to experiment with all three.

4.3.1 Model Architectures

Recurrent Neural Network

Recurrent Neural Networks (RNN) represent a neural network architecture tailored for sequential data. Unlike the classic feed forward networks, RNNs keep some sort of hidden state that lets them remember information from previously processed elements in a sequence, which makes them ideal for tasks where the order of the inputs is important. This is accomplished by RNNs, which allow weight sharing, meaning that the same weights are applied in a recursive manner over each input of the sequence, enabling the network to "remember" previous inputs. The major problem in using RNNs is that they suffer from issues such as vanishing gradients, which hinders them in learning long-range dependencies within sequences.

For any time t , the input sequence $\{x_1, x_2, \dots, x_T\}$ is considered to update the hidden state h_t . The hidden state is computed by combining the current input x_t and the previous hidden state h_{t-1} . More precisely, there is the element-wise multiplication of the input x_t with the input-to-hidden weight matrix W_{xh} , and the multiplication of the previous hidden state h_{t-1} with the hidden-to-hidden weight matrix W_{hh} . Then, the two results are summed and passed through a nonlinear activation function - \tanh or ReLU , - to obtain a new hidden state h_t :

$$h_t = f(W_{xh}x_t + W_{hh}h_{t-1})$$

Once the hidden state h_t is updated, it is used to generate the output for that time step. The hidden state is multiplied by the hidden-to-output weight matrix W_{hy} , yielding the output y_t :

$$y_t = W_{hy}h_t$$

The RNN architecture can be seen at Figure 4.2a

LSTM

Long Short-Term Memory can be regarded as an advanced version of RNNs that is designed in such a way that it need not face any problem while learning long-range dependencies. LSTMs are using special types of gating in order to manage the flow of information within the network, by which it may keep more important information over longer sequences and discard irrelevant data.

In a basic LSTM, there exist three major gates that regulate the memory. These include the forget gate, the input gate, and the output gate. The forget gate makes decisions on which previous information from the earlier cell state C_{t-1} needs to be forgotten. The input gate selects what new information from the current input x_t is added to the cell state. Wherein the cell state C_t is updated, while the output gate will select what portion of the updated cell state shall move to the hidden state h_t that will be used to compute the output.

The architecture of the model can be seen on Figure 4.2b and consists of:

Forget gate: Decides what to forget and what not from the previous cell state. It takes the current input x_t and the previous hidden state h_{t-1} and a sigmoid activation function is applied in order to compute the forget gate f_t :

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f)$$

Input gate: Decides what new information matters so it can get added to the cell state. It also has two parts: one of the parts is to compute which values to update (i_t), and the other to create a candidate update \tilde{C}_t :

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_C)$$

Cell state update: The cell state C_t by combining the previous cell state gets updated C_{t-1} , scaled by the forget gate, and the candidate update scaled by the input gate:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Output gate: The output gate decides which part of the updated cell state is able to influence the hidden state. This hidden state is then used to compute the output for the time step:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

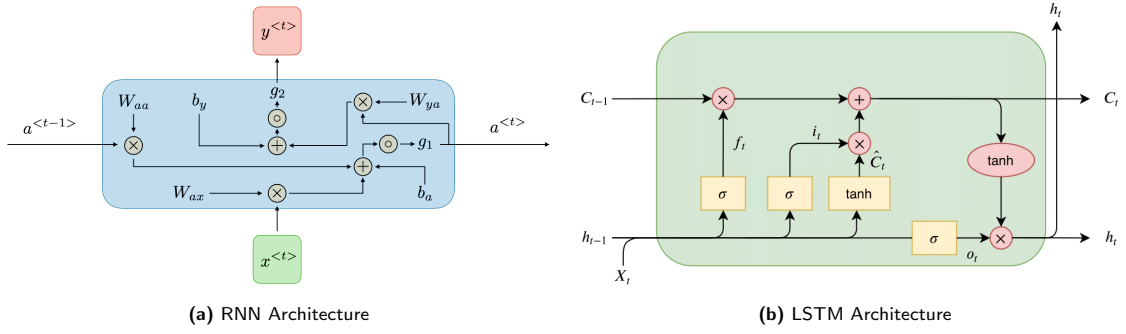


Figure 4.2: Comparison of RNN and LSTM Architectures

Bidirectional LSTM

BiLSTM processes the input sequence of a model in both forward and backward directions, hence capturing information about both past-previously observed time steps-and future-later in time-context. One forward LSTM processes the input like any other, from start to end, while one backward LSTM processes it in reverse, from end to start. Then, at every time step, the hidden states coming from both directions are concatenated to form the final output of the model. Such an architecture enhances performance for tasks where context from both directions is informative, such as speech recognition or machine translation.

xLSTM

xLSTM is based on the general architecture of LSTM, but presents some enhancements on handling long term dependencies and complicated patterns in sequential data. Memory cells and three kinds of input, forget, and output gates are organized in the manner of the traditional LSTM to control information flow and avoid problems such as exploding or vanishing gradients. What really makes xLSTM different, is that it supports much more complex sequences with extended memory capacity, among other features that allow the model to capture patterns over longer time horizon. This may also include architectural enhancements like peephole connections, allowing direct access to cell state from gates and enabling the model to better focus on information that is relevant. The basic components of the xLSTM architecture are

- **Forget Gate:** The forget gate decides for which parts of the cell state from the last cell c_{t-1} it should keep or throw away. It takes as input the hidden state at the previous step h_{t-1} and the input at the current step x_t , and applies a sigmoid activation function:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

where f_t is the forget gate output, W_f are the weights, and b_f is the bias term. The forget gate value f_t is used to scale the previous cell state.

- **Input Gate:** The input gate updates the cell state with new information. Initially, the input gate computes which of the values to update:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

The candidate gate \tilde{z}_t is calculated, and, at the same time, the input gate modulates the new candidate state for the cell:

$$\tilde{z}_t = \tanh(W_z \cdot [h_{t-1}, x_t] + b_z)$$

- **Candidate Gate:** The candidate gate proposes a new candidate state \tilde{z}_t to be added to the cell state:

$$\tilde{z}_t = \tanh(W_z \cdot [h_{t-1}, x_t] + b_z)$$

- **Cell State Update:** The new cell state c_t gets updated by a combination of the previous cell state c_{t-1} , scaled by the forget gate, and the new candidate information scaled by the input gate:

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{z}_t$$

- **Output Gate:** This gate decides which parts of the cell state are sent to the hidden state h_t and which parts are not:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

The hidden state is calculated after a non linearity is applied to the cell state:

$$h_t = o_t \cdot \tanh(c_t)$$

- **Hidden State:** Finally, the newly updated hidden state h_t is passed to the next LSTM cell in the sequence.

Some variants of xLSTM increase the number of memory cells or layers. This enables deeper representations of sequence data and more flexibility with respect to learning hierarchies of temporal dependencies. Skip connections and layer normalization may further be added to enhance gradient flow and training stability. Due to certain modifications, the xLSTM is often computationally more efficient. This means it is practical for large datasets or complex sequences, such as sensor data, financial modeling, and natural language processing tasks. Such modifications make the xLSTM well-suited for advanced prediction tasks in estimating engine failure or complex multivariate time-series analysis.

Transformer Based Models

They process the entire data simultaneously, but they are less computationally heavy in comparison with RNN's (e.g. RNNs and LSMTs) architecture, while at the same time being able to catch long term dependencies. They have the ability to detect time series anomalies in the same manner they can process sequential data for translation in text data. Positional encoding is a requirement, in order to help the model understand the order of the data, which is very crucial for accurate timeseries analysis and anomaly detection. The transformer is built with layers of self attention mechanism, multi head attention and feed forward neural networks. The self and multi head attention, when dealing with a text based task, allows the model to weigh the importance of different words in a sentence, considering their relationship regardless of their position, enabling better understanding of context. The same way works when we are dealing with timeseries data, as it processes input a sequence of continuous tokens (e.g timesteps). Those series are processed by first embedding and adding positional encodings to the token and then

generating outputs in sequence, by predicting the next token based on the previously generated ones. There are three different architectures, Encoder only, Decoder only and Encoder-Decoder, each suited to different types of tasks, depending on whether the end goal is to understand, generate or transform sequences. Figure 4.3 shows the Encoder-Decoder architecture, visualizing that Attention is the most important part of the transformer model. In our use case we use a decoder only architecture as seen in Figure 4.3b because it efficiently predicts future values in a step by step sequential manner based on previous observations.

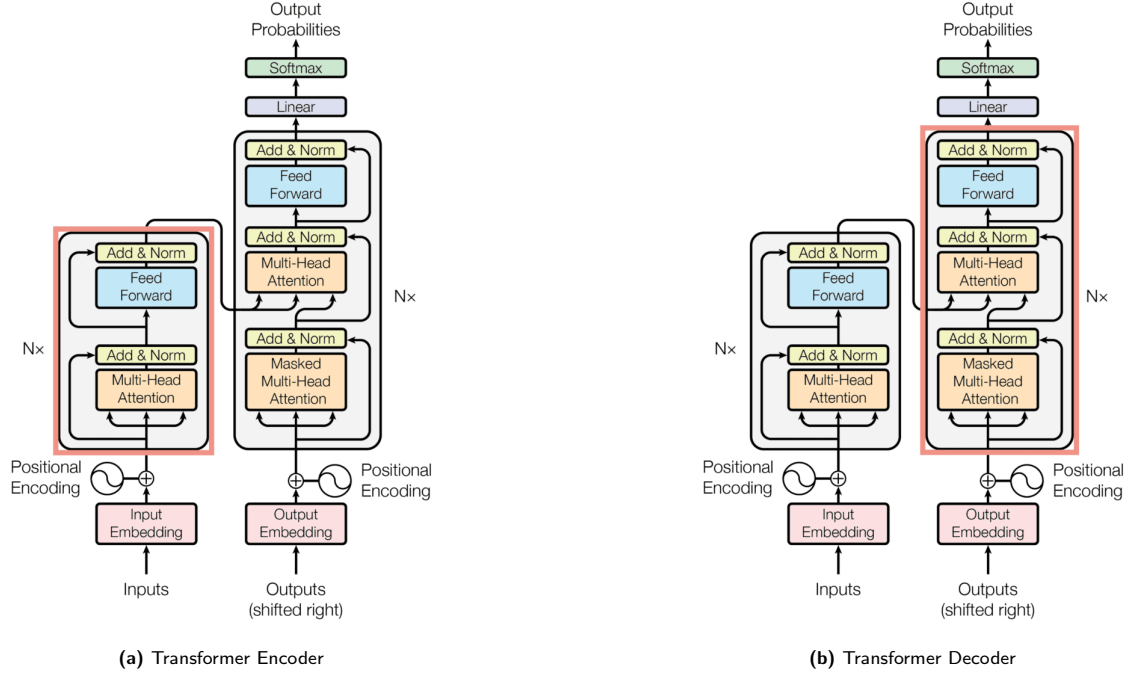


Figure 4.3: Transformer Encoder and Decoder

4.3.2 Training and Evaluation

After selecting the appropriate model architectures, we trained the models using the preprocessed data. Hyperparameter tuning was performed to optimize model performance, and evaluation metrics such as accuracy, was used to assess the models' effectiveness.

4.4 Model Interpretability

4.4.1 SHAP Values

SHAP (SHapley Additive exPlanations) values are a way to explain the output of any machine learning model. Based on Shapley values from cooperative game theory, where a way to fairly distribute the "payout" is provided (the anomaly detection in our case) among the features. It is an approach to interpreting ML models by assigning each feature a value of importance for a particular prediction.

SHAP Value ϕ_i for feature i is defined as:

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} [v(S \cup \{i\}) - v(S)]$$

where:

- N is the set of all features.
- S is a subset of N not containing feature i .
- $v(S)$ is the value of the coalition S (often taken as the model prediction for the subset of features S).
- $|S|$ is the number of features in subset S .
- $|N|$ is the total number of features.

4.5 Integration with Large Language Model

4.5.1 Pretrained Transformer Models

Large Language Models (LLMs) are a class of AI systems, which have been trained using a huge amount of text data, known as a corpus, to understand and generate human language. The main architecture behind the success of these models is transformers and the concept of attention, topics mentioned in the previous subsection. This architecture allows the LLMs to process and generate text through next-word prediction in a sequence based on the context provided by the words found earlier in the sentence being generated. LLMs can handle a wide range of tasks, including machine translation, text completion, summarizing, sentiment analysis, and a basic form of reasoning, which makes them a great candidate for our task at hand. They can also model patterns in language and generate content that is indistinguishable from human text. For our task, we will utilize a pretrained LLM, already fine-tuned on an enormous corpus, allowing us to take advantage of its knowledge and reduce the time and resources needed for further training.

Pretrained LLMs are language models, which have already been trained on large datasets before they are fine-tuned or applied to a very specific task. During the pretraining phase, the model is exposed to many different text sources, which enables it to learn grammar, vocabulary, reasoning, and facts about the world. After this phase is completed, it can be further fine-tuned on smaller, more specialized datasets to adapt to particular and more specific use cases. These LLMs that have undergone the pretraining phase are very powerful because they come with a deep understanding of the language out of the box, reducing the need for any retraining, and at the same time, this allows them to adapt very quickly to new applications. This approach enhances their efficiency and makes them more versatile.

I think we should also mention RAG here – basically what you want is to have a LLM to guide the conversation, so using all its pre-trained language skills, but to access exclusively pre-specified technical documents. So part of the prompt should be: restrict your replies to be taken from this set of technical documents.

4.5.2 Generating Output

The output is a text-based approach, directly given to the operator, achieved by efficient prompting and LLMs. Efficient prompting involves creating very clear and specific prompts, which provide necessary context and structure in order to effectively guide the LLM towards the goal of interest. After the prompt

is passed, in combination with the SHAP values, into the LLM, the LLM uses reasoning techniques to choose which of the actions are more vital for the vehicle's functionality. Finally, the recommended actions are presented to the operator, in a priority order.

Chapter 5

Results

5.1 Methodology Recap

As explained in the previous sections, multivariate data is collected from the engine's sensors of a flying vehicle, then processed before being used as an input to a model, such as LSTM, RNN, Bidirectional LSTM, xLSTM or Transformer. The primary object is to predict engine malfunctions based on the data collected. During the preprocessing stage, normalization was performed and the engine cycles were split in chunks of 50 consecutive cycles each. These chunks are used as an input to the aforementioned models and their output is a binary prediction indicating whether the engine is expected to malfunction or not in the upcoming cycles. When the model predicts a failure within the specified timeframe, SHAP values are calculated to identify the most influential of the sensors, which contributed the most to the model's prediction. This insight is then processed by a pretrained LLM, which is able to generate explanations for the failure the models predicted and suggests the optimal actions, which can be taken in order to prevent or mitigate the malfunction.

$$\underbrace{x_1, x_2, \dots, x_{50}}_{\text{chunk 1}} \underbrace{x_{51}, x_{52}, \dots, x_{100}}_{\text{chunk 2}} \cdots \underbrace{x_{n1}, \dots, x_{nT}}_{\text{chunk } n} \quad (5.1)$$

I am not sure i fully understand the chunking. Does failure occur at the last measurement x_{nT} ? Also, am i right in thinking that the last chunk might not be fully filled?

5.2 Results

In this chapter, we present the results of our experiments, including performance on the dataset of various models, an interpretability analysis using SHAP values, and integrations with Large Language Models that generate actionable insights.

5.3 Model Performance

Several models were evaluated, including RNN, LSTM, Bidirectional LSTM, xLSTM, and a Decoder-only Transformer, on their ability to predict engine failures. The performance of each model is presented below.

5.3.1 RNN model

Validation set performance

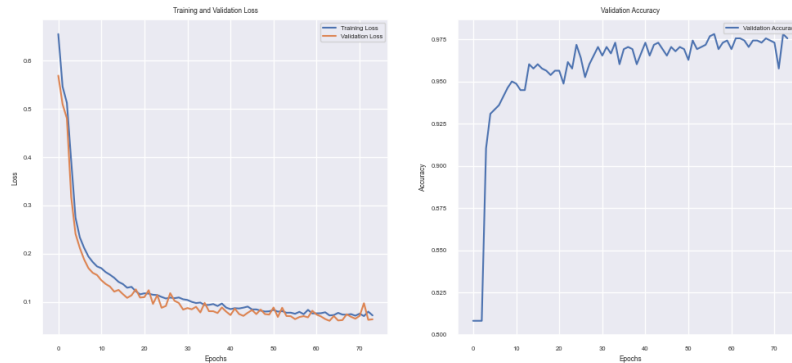


Figure 5.1: RNN Training and Validation Loss (left) and Validation Accuracy (right) over 75 epochs.

The models shows a very good performance with a decrease in both the training and the validation loss in the first epochs, leading to stabilization, as seen in figure 5.1. Effective training is indicated by the validation accuracy, which plateaued at 97%. The model appears to have managed to reach great performance with early stopping kicking in at about 70 epochs. Further training most likely will not lead to substantial improvements.

Test set performance

The model performs very good giving an accuracy of 96.47% on the test set as seen on Figure 5.1, but there are some instances where the prediction is wrong. Despite all the shortcomings, the model will give very good performance for the biggest part of the data.

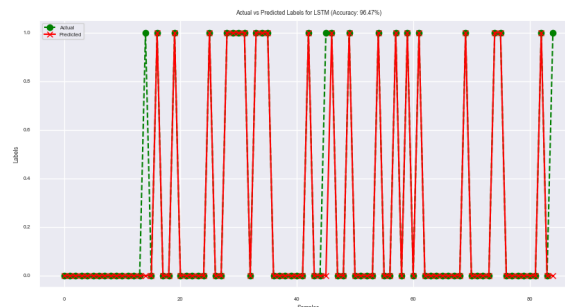


Figure 5.2: Actual vs Predicted labels for RNN model.

Use the caption of figure to explain the x- and y axis, the x-axis says: samples (if i can read correctly.. are these the engines? The y-axis indicates whether failure was predicted and actual happened? But shouldn't that be coupled to a specific chunk ?

5.3.2 LSTM

Validation set performance

The loss decreases steadily and the accuracy reaches a 97.5%, on average performance, on the validation set as seen in Figure 5.3. It seems that the LSTM achieves slightly higher accuracy compared to the regular RNN, which suggests that it might be more effective at learning the underlying patterns within the data. There is also consistence maintenance on the performance, similarly as the RNN.

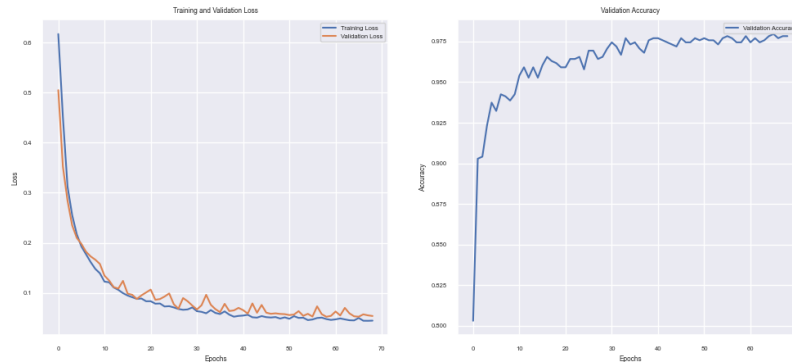


Figure 5.3: LSTM Training and Validation Loss (left) and Validation Accuracy (right) over 68 epochs.

Test set performance

This model also has the same accuracy as the RNN, as seen on Figure 5.3. The classification task is done quite well, with only few errors, which definitely makes this a strong candidate for sequential binary classification tasks.

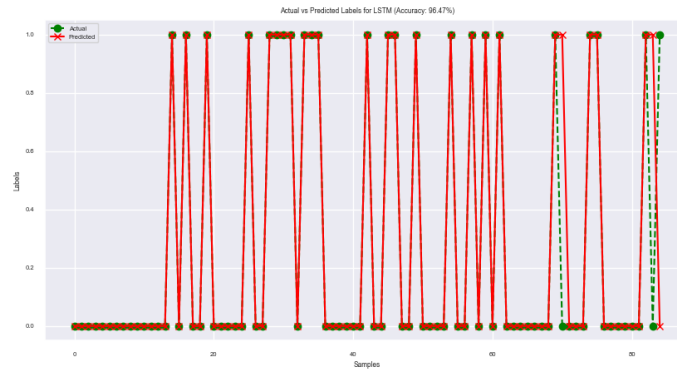


Figure 5.4: Actual vs Predicted labels for LSTM model.

Bidirectional LSTM

Validation set performance

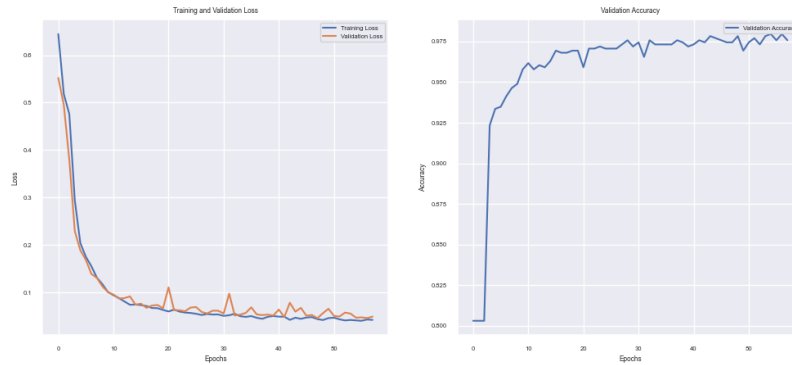


Figure 5.5: Bidirectional LSTM Training and Validation Loss (left) and Validation Accuracy (right) over 58 epochs.

The loss rapidly decreases and stabilizes with very minimal fluctuation between both the training and validation set as seen in Figure 5.5, an indication of good generalization. Accuracy reaches 97.5% in the validation set, similar to the LSTM but stability is achieved in fewer epochs. In comparison with RNN and LSTM the bidirectional LSTM is able to reach high accuracy quicker, which means it might be able to capture more complex patterns in the data effectively, leading to faster convergence and better overall performance.

Test set performance

The Bidirectional LSTM, while it is supposed to catch dependencies in both ways, performs worse than both the RNN and the LSTM. The accuracy on the test set reached 95.29% as seen in Figure 5.6

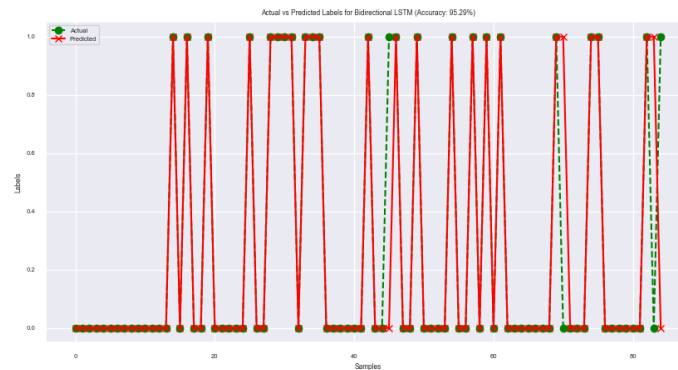


Figure 5.6: Actual vs Predicted labels for BiLSTM model.

5.3.3 xLSTM

Validation set performance

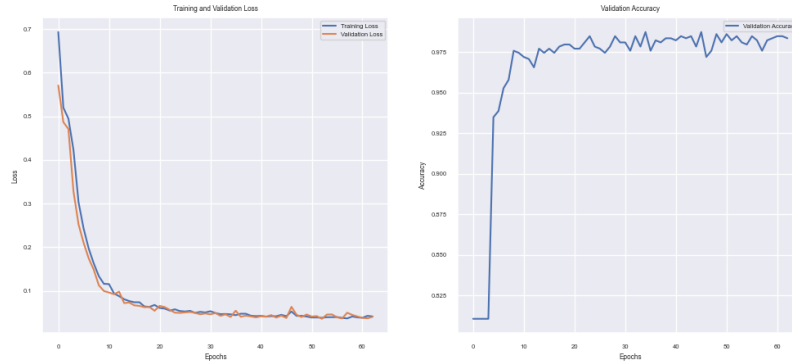


Figure 5.7: xLSTM Training and Validation Loss (left) and Validation Accuracy (right) over 50 epochs.

For the xLSTM architecture the loss decreases rapidly and stabilizes, for both the training and validation set, which indicates a strong generalization without overfitting. There is stabilization of the accuracy around 97.5% as seen in Figure 5.7. Compared to RNN, LSTM, BiLSTM and Decoder only Transformer, this model is able to demonstrate stable and efficient learning, achieving similar results to the BiLSTM but with a smoother and more consistent convergence, making it one of the most reliable models, compared to the previous ones tested.

Test set performance

Similarly, the xLSTM also performs comparably like the LSTM and the RNN. It generalizes both classes well enough but with minor miss-predictions. This would be a great alternative to use for tasks of binary classification.

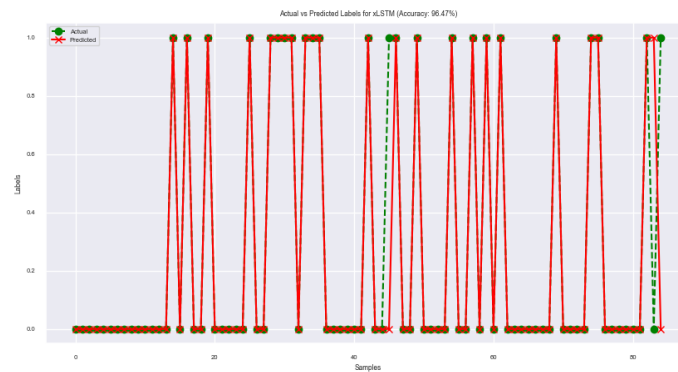


Figure 5.8: Actual vs Predicted labels for xLSTM model.

5.3.4 Decoder only Transformer

Validation set performance

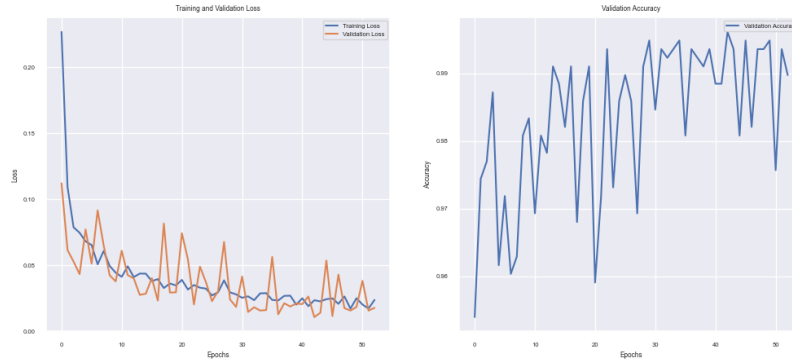


Figure 5.9: Decoder only Transformer, Training and Validation Loss (left) and Validation Accuracy (right) over 50 epochs.

While the training loss decreases and stabilizes, the validation loss is exhibiting fluctuations as seen in Figure 5.9, which could be an indicator that the model is struggling with consistency in generalization. Validation accuracy is showing high volatility with sharp peaks and big drops, in a range from 96% to just under 100%, which suggests that the model is sensitive to certain data patterns or maybe overfitting on some very specific features.

Test set performance

It performs better in this binary classification task, as it achieves a maximum accuracy of 98.82%, as seen on Figure 5.10. The Decoder only Transformer model deals with very high precision by miss-classifying only one instance. This model is the best candidate for the next step, as it achieved the highest accuracy on the test set.

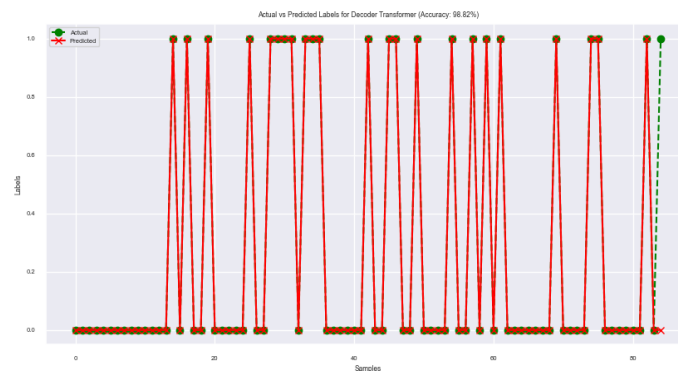


Figure 5.10: Actual vs Predicted labels for Decoder Transformer model.

5.4 Interpretability Analysis with SHAP Values

5.4.1 SHAP Analysis

Next step of the pipeline is to interpret the results and find the combination and interaction of the sensors leading to the engine malfunction. In order for this to work efficiently, SHAP values are calculated for each of the sensors. Figure 5.11 shows the SHAP values for 4 instances that within the next cycles the engines stopped working in a meaningful way.



Figure 5.11: SHAP values for 4 instances with corresponding function of each sensors

Initial explanation of the SHAP values

For the first instance, **High-Pressure Turbines Cool Air Flow** and **Low-Pressure Turbines Cool Air Flow** have the most significant impact to the model's prediction. At the same time, **Fan Inlet Temperature** and **LPT Outlet Temperature** impact negatively the prediction, an indication that they were working as expected before the engine's failure.

In the second instance, **HPC Outlet Static Pressure** has a strong positive contribution to the prediction, which means there is a strong influences to the model's output. **Physical Core Speed** and **Engine Pressure Ratio** also contribute, but to a lesser degree compared to the **HPC Outlet Static Pressure** sensor.

For the third the **HPC Outlet Pressure** has the most influence on the upcoming malfunction. At the same time, both **Physical Core Speed** and **Physical Fan Speed** contribute to the model's output. In contrast, **Low-Pressure Turbines Cool Air Flow** and **Fan Inlet Pressure** negatively affect the prediction, which suggests that those sensor readings were within the normal range, that the model doesn't associate with failure conditions.

Finally, for the last instance 4, the most dominant influence occurs from **LPC Outlet Temperature**, with the **HPC Outlet Pressure** and **Fan Inlet Pressure** also contributing, but to a lesser extent. The **Corrected Fan Speed** and **Engine Pressure Ratio** have negative contributions, indicating that they were properly working and within the normal value range before engine breakdown.

5.5 Integration with Large Language Models

5.5.1 LLM Reasoning and Next Steps

After calculating the SHAP values the next step is to use those values as input into a pretrained Large Language Model. By employing efficient prompts, the LLM can be directed to reason, analyze and provide the necessary steps to prevent or fix the engine fault that is going to arise. Supplementing the LLM with an engine manual as a knowledge base is also possible and will enhance its ability to deliver more detailed information on the normal operation, behavior and potential faults of the engine and its sensors.

Efficient Prompting

In the context of LLMs this refers to the practice of crafting very efficient prompts. Those are utilized in a way to maximize the relevance and the overall quality of the responses the model is going to provide, while at the same time minimising unnecessary ambiguity. It involves crafting very specific, concise and unambiguous prompts that the model can understand and respond accurately. Essential context is provided, avoiding details that seem unnecessary for the final response of the LLM, and refining those prompts through iteration. To illustrate and have a basic understanding on how these principles can be effectively applied let's visualize how those can be tailored for different LLMs, such as ChatGPT 4o, Sonnet 3.5 and LLaMA 3.1. The examples following will demonstrate how prompts can be crafted in order to leverage the strengths of each of the models.

Prompt example 1 *You're an expert vehicle engineer, specialized in flying vehicle engines. Based on the SHAP values provided some of the sensors have been identified as critical contributors to an upcoming engine failure. Prioritize those with the highest contribution that lead to failure. Combine the sensors with those highest SHAP values because their collaboration and interaction leads to failure. Provided the manual that shows the maintenance of each of the engine's parts, reason and conduct a root cause analysis for each of these sensors. Additionally, specify the preventive maintenance steps that should be taken, including any adjustments or re-calibrations according to the manufacturer's manual. Your output should be the top 3 steps based, prioritizing on the importance of each, and only that.*

Prompt example 2 *With your expertise in flying vehicle engines, you've identified the contributions of key sensors to potential engine failure, based on SHAP values. Prioritize those with the highest contribution that lead to failure. Combine the sensors with those highest SHAP values because their collaboration and interaction leads to failure. By using the engine's manual for maintenance, analyze those sensors to find out which was the root cause that caused the problem. Once this is done, propose three actions to be taken ASAP in order to maintain or fix the problem, including any necessary sensor adjustments, re-calibrations or replacement of parts. Prioritize them by their significance to engine longevity and the capacity to have the engine up and running with minimal downtime.*

5.5.2 Instance 1

Across the four LLMs used, the consensus is that the **High-Pressure Turbine Cool Air Flow** and the **Low-Pressure Turbine Cool Air Flow** sensors show the most significant positive SHAP values, which indicates a cooling system malfunction that may lead to overheating and damage to the turbine if left unchecked. **Corrected Fan Speed** is also a critical factor and all the models recommend re-calibration or inspection to prevent imbalances to the airflow, which could cause engine inefficiencies.

All models agree on **cooling system inspection and cleaning, sensor calibration and turbine blade inspections** as the highest priorities. They differ slightly in what they recommend, with GPT4o emphasizing the immediate overhaul of the cooling systems and re-calibration of fan speed sensors, while Claude 3.5 points to the addition of the fuel metering system for inspection, given its contribution based on SHAP. LLaMA 3.1 and Gemini Pro 1.5 address more interconnections between the cooling and airflow issues and point out that proper airflow management and good turbine blade condition are highly vital for the prevention of engine defect.

Out of the four models, there is consistency in terms of which factors are given emphasis like cooling, fan speed, and turbine performance. The general feeling among them, though, is that immediate and comprehensive cooling checks of the entire system, as well as controlling the fan speed and turbine blades, will forestall a complete engine failure.

More details about the LLM response can be found at Appendix

5.5.3 Instance 2

In the second instance, SHAP across the 4 LLMs unanimously point at the same sensors: **High-Pressure Turbine Cool Air Flow (HPT)**, **HPC Outlet Static Pressure**, and **HPC Outlet Temperature**, as the most important for the development of a failure of the engine in the nearest future.

GPT4o emphasizes more on checking the cooling system and re-calibrating the sensors, including those in the HPT cooling system and the HPC outlet pressure and temperature sensors, in order to avoid an upcoming situation in which overheating affects the performance level. Claude 3.5 puts a lot of emphasis on in-depth inspections of the high pressure compressor and turbine cooling systems, such as blade condition checking, airflow and pressure sensors that affect the maximum performance. Same concerns are shared by LLaMA 3.1, but with extra attention to the compressor blades condition and the pressure sensors. The former requires the blades to be in a good condition and the latter need to be properly calibrated. Gemini Pro 1.5 on the other hand directs our attention to issues surrounding inspection and cleaning operations in the HPT cooling system and recalculation of HPC static and outlet pressures' sensors for which proper actions should be taken.

All these models are again leaning more towards cooling and airflow problems, and though ChatGPT4o and Claude 3.5 recommend sensor re-calibration and compressor performance, the LLaMA 3.1 and Gemini Pro 1.5 lean more towards the mechanical checkup of the components like compressor blades, take-off pressure sensors, among others, for the best performance of the engine. Whichever way we might look at it, the immediate recommendations are leaning towards cooling system service and the re-calibration of the pressure sensors to avoid engine failure.

More details about the LLMs response can be found at Appendix

5.5.4 Instance 3

In the third instance, it is noticed from the SHAP analysis that the **Ratio of Fuel Flow to Ps30**, **HPC Outlet Static Pressure**, and **Core Speed, or Fan Speed**, are the top contributors to engine potential failure. Each of the models has managed to identify the crucial issues, which are very important to

maintain stability for the prevention of an engine malfunction. Those include **incorrect fuel regulation**, **pressure imbalance** and **mechanical speed control issues**.

ChatGPT4o focuses on the re-calibration of the fuel system, HPC outlet pressure sensor, and core speed sensor, underlining the proper fuel-to-pressure ratio, efficiency of airflow, and mechanical balance that will prevent degradation. Similarly, Claude 3.5's priorities of inspection of the fuel system and compressor target calibration of the fuel metering system, pressure sensors, and ensuring that fan speed is stable to avoid an imbalance that degrades performance. The LLaMA 3.1 draws particular attention to cleaning and inspection of the fuel system, compressor blades, and undertaking fan blade balancing as some of the precautionary measures. The Gemini Pro 1.5 also does not forget the fuel system calibration and advises a close check of the combustion chamber and high-pressure turbine, factoring in a possibility of thermal or mechanical failure due to imperfect fuel balancing.

All models unanimous agree on the need for fuel system upkeep and regulation of its pressure inside the HPC. Thus, as all models suggests, the consensus comes as comprehensive checks on the fuel system or re-calibration of the pressure sensors to avoid a catastrophic engine failure.

More details about the LLM response can be found at Appendix

5.5.5 Instance 4

The fourth instance reveals that the main sensors according to the SHAP values are **HPC Outlet Temperature**, **Ratio of Fuel Flow to Ps30**, and **Corrected Core Speed**. What all of those models bring to the light is thermal stress, fuel management and mechanical speed as a set of factors that need to be kept under control, so the performance doesn't deteriorate and an upcoming engine downtime is avoided.

The inspection priorities for ChatGPT4o are the wear or cooling blockages in the HPC section, re-calibration of the HPC outlet temperature sensor, and a comprehensive fuel system and core speed sensor inspection to ensure proper fuel-to-air ratios and rotational speed. In the case of Claude 3.5, a related concern deals with HPC temperature problems, re-calibration of the fuel metering system, and verifying the core speed anomalies that provide stability in engine performance. Similarly, LLaMA 3.1 includes HPC blades inspection; faulty cooling systems; and correct fuel flow and speed sensors. On the other hand, Gemini Pro 1.5 mentions the examination of HPC, fuel control system, speed control systems for variable stator vanes, and bearings.

They all agree that the temperature problem in the HPC needs to be fixed and the same goes for the fuel flow. ChatGPT and Claude focus more on calibration of the sensors, while LLaMA and Gemini indicate physical inspections of the blades, cooling passages and stator vanes. Overall, it is clear that maintenance is needed in the HPC, fuel system and core speed control in order to prevent the upcoming failure.

More details about the LLM response can be found at Appendix

Chapter 6

Discussion

In this chapter, we discuss the implications of our findings, the effectiveness of integrating Large Language Models (LLMs) with predictive maintenance workflows, and potential areas for future research.

6.1 Discussion

6.1.1 Integration of LLMs in Predictive Maintenance Workflows

During this experiment we explored the use of Large Language Models in a predictive maintenance workflow, with main focus on engines of flying vehicles and emphasized ways that LLMs can support decisions in high stake environments. We managed to couple traditional statistical methods with interpretability given by SHAP values and, at the same time, leverage the reasoning capabilities of LLMs to provide actionable maintenance steps. Integrating these, our aim was to bridge the gap between complex predictive model outputs and the practical maintenance tasks needed, either to prevent or rectify engine failures.

6.1.2 Anomaly Detection and Model Interpretability

Among the main factors influencing predictive maintenance of engines is anomaly detection, the ability for the model to recognize patterns that might lead to an engine malfunction in the future. Anomaly detection is traditionally based on black box models and represented by RNNs, LSTMs and transformers. The challenge is still to know why these models make such prediction due to their opaque internal workings. To that end, in order to overcome the opaqueness, SHAP values allow interpreting model outputs to quantify the contribution of each of the sensors in the determination of whether an anomaly has been successfully predicted or not. This level of interpretability is quite fundamental for guiding maintenance engineers by showing them specifically which sensors are triggering the alert and therefore allowing targeted intervention rather than general inspections.

6.1.3 Key Findings from SHAP Analysis

In the SHAP values from our own experiments some of the sensors were identified that always contributed to the failure of an engine: **HPC outlet temperature, fuel flow to Ps30 ratio, and core speed sensors**. Among there instances analyzed, cooling system issues, more specifically those related to high presure turbine cooling airflow and pressure imbalances, such as that of static pressure at the outlet of HPC, were among the major contributors to failures.

6.1.4 Leveraging LLMs for Actionable Insights

In our workflow, the interpretability provided by the SHAP values is extended even more by LLMs toward intuitive and actionable insights. Once the key contributing sensors have been pinpointed by those, the task of root cause analysis is assigned to LLMs along with preventive action suggestions. Several variants of LLMs were used, such as GPT 4o, Claude 3.5, LLaMA 3.1 and Gemini Pro 1.5 in reasoning over these sensor contributions and generating a prioritized list of maintenance, prevention or fixable actions. Once again we notice, that a manual was not available for the particular engine so the answers provided by the LLMs remain generic. Each of the LLMs had a structured insight into the causes and the measures required to overcome the problem arising. For all of them, recommended actions by LLMs included checks on the cooling system, fuel system re-calibration and adjustments of the core speed sensor.

6.1.5 Comparison of LLM Responses

GPT-4o emphasized the immediate need for inspection of sensors and re-calibration, while Claude 3.5 focused more on how temperature anomalies interact with inefficiencies in the fuel system. LLaMA 3.1 allowed deeper mechanical diagnostics, focusing into blades and adjusting vane application. Gemini Pro 1.5 took it further into detailed mechanical wear and blockages that would lead to reduced airflow. At the same time, these models pointed toward the possibility of using LLMs for real time decision making by providing fast and accurate responses, given the current data from the engine. What makes them valuable in high stress situations, such as military operations and time restricting maintenance scenarios, is explaining what has gone wrong so far and giving a clear action plan. Also, we firmly believe, providing a manual in combination with the prompt would lead to better results and specific actions that can be taken for the particular engine model.

6.1.6 Advantages of LLMs in High-Stakes Environments

Another exceptional advantage of LLMs is that they can distinguish meaningful data from mere technical standpoint and present in forms that will be actionable by operators. For purposes of this work, these LLMs bridge the gap between the output of the anomaly detection models and what actions must be taken by the engineers. Given the SHAP values as input, the next step for LLMs is ranking steps by the probability not to fail. This ordering ensures that attention is directed at the most critical components first, which maximizes the chances of keeping the engine operationally ready with the least possible downtime.

6.1.7 Generalization to Other Domains

While the presented study focuses on aircraft engine's this framework of anomaly detection combined with SHAP and LLMs can be generalized for all other domains of predictive maintenance. Any vehicle with a system, where sensors generate large data volumes can make use of interpretability and reasoning from this approach. Apart from that, the flexibility of those models allows them to work on various domains with minimum retraining requirement, making it very scalable for different applications in the industry.

6.2 Future Work and Considerations

While this study has demonstrated the effectiveness of the combination of SHAP values and LLMs for predictive maintenance, several questions remain open for future research and development in terms of

enhancing the applicability and robustness of this approach. Among the major future works, implementation of this framework is on real time monitoring systems. Only when combined with anomaly detection and SHAP interpretability in a live operational environment will engineers gain immediate insights and recommendations at the instance of an issue. Furthermore, inclusion of domain specific knowledge would enrich the understanding of the underlying system by LLM. Finetuning and supplementing the LLMs with technical manuals, historical service logs and domain specific guidelines, which would enable them to provide more precise and customized actions. Finally, privacy is also a concern when implementing an LLM driven predictive system, especially when sensitive operational data from engines or vehicles comes into consideration. Making sure that the data used for the training of the model and analysis is securely handled and subjected to data protection regulations in order to avoid unauthorized access of misues is crucial. Future work should also focus on privacy preserving techniques, for example, by using a locally trained LLM specific for the task at hand, which would enable model training and deployment without the release of sensitive information.

Chapter 7

Conclusion

In this work, we investigated how Large Language Models can be integrated with predictive maintenance models to extend decision making in high stake environments, specifically focusing on the flying vehicle engines. We aimed to bridge the gap between complex outputs of predictive models and practical maintenance task by coupling traditional statistical methods with interpretability techniques leveraging the reasoning of LLMs.

We started with the dataset consisting of multivariate time series of various sensor on aircraft engines of the same type. Preprocessing was done on the aforementioned data, including normalization and chunking to extract temporal patterns and dependencies. In our experiments, we covered a wide range of models that were capable of modeling multivariate time series: RNNs, LSTMs, Bidirectional LSTMs, xLSTMs, and Decoder only Transformer architectures.

Our best models were trained on a binary classification tasks to predict whether an engine is likely to fail in the cycles to come. Compared with other models, the Decoder-only Transformer model showed the best performance on the test set, which indicates that the Transformer-based architecture works well for complex temporal dependencies in the data.

To interpret the predictions of these black box models, we employed SHAP values. SHAP values helped quantify the contribution of each sensor to the predictions of the model by indicating which of those sensors contributed the most to the malfunctioning of the engines. This level of interpretability is very helpful to identify and understand the root cause and sensors' interactions, which led or will lead in an engine failure.

Beyond the SHAP values, LLMs were integrated into the output as a means to generate actionable maintenance steps. Utilizing efficient prompting and the reasoning powers of LLMs, such as GPT-4o, Claude 3.5, LLaMA 3.1, and Gemini Pro 1.5, the SHAP values were transformed into actionable steps, which can be taken, as soon as possible, by the operator. These LLMs not only summarized the most contributing sensors, but also executed root cause analysis, distinguishing which of the sensors and their interactions play the most crucial role. At the same time, they proposed prioritized maintenance actions that can prevent or at least guide the maintenance personnel in the steps that need to be taken so that the engine can be fixed.

This study points to a potential integration of LLMs with predictive maintenance models to enhance decision-making in high-stakes settings. By coupling SHAP values with reasoning capability from an LLM, actionable insights are provided on complex model outputs into practical maintenance tasks. This technique not only does it make black box models more interpretable but also highly scalable and applicable across different industries. The envisioned framework thus calls for real implementation in real-time, integrating domain knowledge, and addressing privacy concerns for maximum benefit.

The combination of state of the art predictive models and techniques which make the outputs of those models interpretable in combination with the reasoning of LLMs bring predictive maintenance to a very promising direction. This integrated approach can potentially improve operational efficiency, safety, and decision-making in high-stakes environments substantially. Further research and development work shall focus on refining the proposed framework and working out identified challenges. Regarding the applicability in various industrial settings, an examination is also important.

[Add a reference to your github and the dataset.](#)

Bibliography

- [1] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, et al. "Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI". In: *Information Fusion* 58 (2020), pp. 82–115.
- [2] Lili Chen et al. "Decision Transformer: Reinforcement Learning via Sequence Modeling". In: *Advances in Neural Information Processing Systems*. Vol. 34. 2021.
- [3] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *NAACL-HLT*. 2019, pp. 4171–4186.
- [4] Finale Doshi-Velez and Been Kim. "Towards A Rigorous Science of Interpretable Machine Learning". In: *arXiv preprint arXiv:1702.08608* (2017).
- [5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, et al. "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale". In: *International Conference on Learning Representations (ICLR)*. 2021.
- [6] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-Term Memory". In: *Neural Computation* 9.8 (1997), pp. 1735–1780.
- [7] Dong Li et al. "MAD-GAN: Multivariate Anomaly Detection for Time Series Data with Generative Adversarial Networks". In: *International Conference on Artificial Neural Networks*. 2019, pp. 703–716.
- [8] Andy Liaw and Matthew Wiener. "Classification and Regression by randomForest". In: *R news* 2.3 (2002), pp. 18–22.
- [9] Bryan Lim et al. "Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting". In: *International Journal of Forecasting* 37.4 (2021), pp. 1748–1764.
- [10] Amy McGovern, Ryan Lagerquist, David John Gagne II, et al. "Making the Black Box More Transparent: Understanding the Physical Implications of Machine Learning". In: *Bulletin of the American Meteorological Society* 100.11 (2019), pp. 2175–2199.
- [11] Larry Medsker and Lakhmi Jain. *Recurrent Neural Networks: Design and Applications*. CRC Press, 2001.
- [12] Alec Radford et al. "Improving Language Understanding by Generative Pre-Training". In: *OpenAI* (2018).
- [13] Bernhard Schölkopf et al. "Estimating the Support of a High-Dimensional Distribution". In: *Neural Computation* 13.7 (2001), pp. 1443–1471.
- [14] Mike Schuster and Kuldip K Paliwal. "Bidirectional Recurrent Neural Networks". In: *IEEE Transactions on Signal Processing* 45.11 (1997), pp. 2673–2681.

- [15] R. P. Singh et al. "Internet of Things (IoT) Applications to Fight Against COVID-19 Pandemic". In: *Diabetes & Metabolic Syndrome: Clinical Research & Reviews* 14.4 (2020), pp. 521–524.
- [16] Ernst J Wolff. "Real-Time Data Analytics for Predictive Maintenance". In: *Proceedings of the Annual Conference of the PHM Society*. Vol. 6. 1. 2014.
- [17] George Zerveas et al. "A Transformer-Based Framework for Multivariate Time Series Representation Learning". In: *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2021, pp. 2114–2124.

Appendix A

Appendix

A.1 Appendix

A.1.1 Instance 1

ChatGPT4o

From the SHAP values shown in the image:

- **High-pressure turbines Cool air flow and Low-pressure turbines Cool air flow:** These sensors have significant contributions to the issue, indicating potential problems in the cooling system of the turbines, which could lead to overheating and subsequent engine failure.
- **Corrected Fan Speed (rpm):** High SHAP values for this sensor suggest airflow or speed issues, potentially leading to imbalances in engine operation or inefficient fan performance.

Given the high SHAP values associated with cooling airflow and fan speed, the root cause of the issue likely lies within the turbine cooling system and the fan speed regulation. Overheating and incorrect fan speeds can lead to imbalances, engine stress, and failure if not addressed.

Based on the analysis, here are three prioritized actions to take:

1. Immediate Cooling System Inspection and Overhaul

Action: Inspect and, if necessary, overhaul the cooling systems for both high-pressure and low-pressure turbines. This includes checking the air ducts, fans, and cooling pathways for blockages, leaks, or wear.

Significance: Cooling is critical to prevent turbine overheating, which can cause immediate and catastrophic engine failure. Ensuring proper cooling will extend engine life and reduce the risk of failure.

Downtime: Moderate; depending on the extent of the overhaul, but critical to perform ASAP.

2. Fan Speed Sensor Calibration and Maintenance

Action: Recalibrate or replace the corrected fan speed sensor. Additionally, inspect the fan and its motor for any signs of imbalance or wear.

Significance: Correct fan speed is crucial for optimal airflow through the engine. Imbalances can reduce cooling efficiency or lead to mechanical damage. Addressing this will ensure efficient engine operation and airflow balance.

Downtime: Minimal to Moderate; recalibration might require a short downtime, but sensor replacement or fan motor servicing could take longer.

3. Turbine Performance Monitoring and Component Replacement

Action: Monitor the performance of both the high-pressure and low-pressure turbine components. If signs of wear or inefficiency are detected, consider replacing or servicing these components.

Significance: Maintaining optimal turbine performance is crucial for engine integrity. Proactive replacement of degraded components can prevent more severe damage down the line.

Downtime: Moderate; depending on the availability of replacement parts, but necessary for long-term engine health.

Priority	Sensor	Action	Significance
1. Immediate Cooling System Inspection and Overhaul	High-pressure turbines Cool air flow Low-pressure turbines Cool air flow	Inspect and, if necessary, overhaul the cooling systems for both high-pressure and low-pressure turbines. This includes checking the air ducts, fans, and cooling pathways for blockages, leaks, or wear and tear.	Cooling is critical to prevent turbine overheating, which can cause immediate and catastrophic engine failure. Ensuring proper cooling will extend engine life and reduce the risk of failure.
2. Fan Speed Sensor Calibration and Maintenance	Corrected Fan Speed (rpm)	Recalibrate or replace the corrected fan speed sensor. Also, inspect the fan and motor for signs of imbalance or wear.	Correct fan speed is essential for optimal airflow and engine performance. Imbalances can reduce cooling efficiency and lead to mechanical issues.
3. Turbine Performance Monitoring and Component Replacement	High-pressure turbines Low-pressure turbines	Monitor the performance of the high-pressure and low-pressure turbines. If signs of inefficiency or damage are detected, consider replacing or servicing the components.	Maintaining optimal turbine performance is crucial for engine integrity. Proactive replacement of degraded parts can prevent more severe damage down the line.

Table A.1: Proposed Actions for Engine Maintenance and Repair

Claude 3.5 Sonnet

Top 3 prioritized steps for preventive maintenance, considering SHAP values and sensor interactions:

1. Inspect and service the high-pressure turbine cooling system:

Reasoning: The high-pressure turbine cool air flow has the highest positive SHAP value, indicating it's the most significant contributor to the potential failure. In jet engines, proper cooling of the high-pressure turbine is critical for several reasons:

- It allows the turbine to operate at higher temperatures, improving engine efficiency.
- It prevents thermal damage to turbine blades, which are exposed to extreme temperatures.
- Inadequate cooling can lead to rapid deterioration of turbine components, potentially causing catastrophic failure.

The high SHAP value suggests that abnormal cooling air flow (either too much or too little) is strongly correlated with the predicted failure. This could indicate issues like blockages, leaks, or malfunctioning control valves in the cooling system.

2. Evaluate and calibrate the fuel metering system:

Reasoning: The "Ratio of fuel flow to Ps30" also shows a high positive SHAP value. This ratio represents the relationship between fuel flow and combustor inlet pressure. An abnormal ratio could indicate:

- Fuel system issues, such as incorrect fuel metering or faulty fuel nozzles.
- Problems with the engine's pressure sensing system.
- Potential combustion instabilities.

Improper fuel-air ratios can lead to a range of problems, from reduced efficiency to engine surge or stall. The high SHAP value suggests that this parameter is deviating from its expected range, potentially contributing to the predicted failure.

3. Assess low-pressure turbine cooling effectiveness:

Reasoning: The low-pressure turbine cool air flow also shows a significant positive SHAP value, though less than the high-pressure turbine. This indicates that cooling issues are affecting multiple engine stages. The low-pressure turbine, while operating at lower temperatures than the high-pressure turbine, still requires effective cooling to maintain efficiency and component life.

The high SHAP value here, combined with the high-pressure turbine cooling issue, suggests a systemic problem with the engine's cooling system. This could be due to:

- Degradation of thermal barrier coatings.
- Clogged cooling passages.
- Malfunctioning cooling air control systems.

By addressing these three areas, we're targeting the most statistically significant contributors to the predicted failure. The combination of cooling system issues and fuel metering anomalies suggests a complex interplay of factors that could lead to engine malfunction or failure if not addressed promptly.

llama 3.1

Top 3 Critical Contributors to Engine Failure (Based on SHAP Values):

Based on the SHAP values, the top 3 critical contributors to the upcoming engine failure are:

1. High-Pressure Turbines Cool Air Flow (HPT CAF): This sensor measures the flow of cool air into the high-pressure turbine, which is essential for maintaining optimal engine performance and preventing overheating.

Root Cause Analysis: A malfunction in the HPT CAF sensor may indicate a blockage or restriction in the cool air flow path, causing the turbine to overheat and leading to engine failure.

Priority	Sensor	Action	Significance
1.	High-pressure turbine cool air flow	Inspect and service the high-pressure turbine cooling system. Address any blockages, leaks, or malfunctioning control valves.	Ensures proper turbine cooling, preventing thermal damage and improving engine efficiency. Failure in cooling could lead to catastrophic engine failure.
2.	Ratio of fuel flow to Ps30 (pps/psia)	Evaluate and calibrate the fuel metering system. Check for incorrect fuel metering or faulty fuel nozzles, and inspect pressure sensors.	Maintains proper fuel-air ratios, crucial for efficient combustion and preventing engine surge or stall. Deviations in this ratio could result in reduced performance or failure.
3.	Low-pressure turbine cool air flow	Assess the low-pressure turbine cooling system. Check for thermal barrier degradation, clogged cooling passages, or malfunctioning cooling control systems.	Ensures efficiency and longevity of the low-pressure turbine by maintaining proper cooling, preventing degradation and component failure.

Table A.2: Top 3 Prioritized Steps for Preventive Engine Maintenance for instance 1

Preventive Maintenance Step: Inspect and clean the HPT CAF sensor and its associated ducting to ensure unobstructed airflow. Check for any blockages, such as debris or ice buildup, and clear them as needed.

2. Corrected Fan Speed (CFS): This sensor measures the speed of the engine's fan, which is critical for maintaining optimal airflow and engine performance.

Root Cause Analysis: A malfunction in the CFS sensor may indicate a problem with the fan's speed control system, causing the fan to operate outside its optimal range and leading to engine instability.

Preventive Maintenance Step: Check the fan speed control system, including the fan speed sensor, controller, and actuator, for any faults or malfunctions. Calibrate or replace the faulty component as needed to ensure optimal fan speed control.

3. Low-Pressure Turbines Cool Air Flow (LPT CAF): This sensor measures the flow of cool air into the low-pressure turbine, which is essential for maintaining optimal engine performance and preventing overheating.

Root Cause Analysis: A malfunction in the LPT CAF sensor may indicate a blockage or restriction in the cool air flow path, causing the turbine to overheat and leading to engine failure.

Preventive Maintenance Step: Inspect and clean the LPT CAF sensor and its associated ducting to ensure unobstructed airflow. Check for any blockages, such as debris or ice buildup, and clear them as needed.

These three sensors are interconnected, and their collaboration and interaction can lead to engine failure. Therefore, it is essential to inspect and maintain all three sensors and their associated systems to prevent engine failure.

Gemini Pro 1.5

Top 3 Preventative Maintenance Steps Based on SHAP Values for Engine Failure Prediction:

High SHAP values for High-Pressure Turbine Cool Air Flow, Corrected Fan Speed, and Low-Pressure Turbine Cool Air Flow suggest a critical interaction between these

Priority	Feature	Step	Significance
1.	High-Pressure Turbines Cool Air Flow (HPT CAF)	Inspect and clean the HPT CAF sensor and its associated ducting to ensure proper airflow and prevent overheating.	Prevents overheating and maintains optimal engine performance by ensuring sufficient cool air flow to the high-pressure turbine.
2.	Corrected Fan Speed (CFS)	Check the fan speed control system, including sensor, controller, and actuator, and calibrate or replace faulty components.	Ensures the engine fan operates within optimal speed range, reducing engine instability and enhancing performance.
3.	Low-Pressure Turbines Cool Air Flow (LPT CAF)	Inspect and clean the LPT CAF sensor and its associated ducting to ensure proper airflow and prevent overheating.	Maintains optimal engine performance by ensuring adequate cool air flow to the low-pressure turbine, preventing engine failure.

Table A.3: Top 3 Preventive Maintenance Steps for Engine Failure Based on SHAP Values

three factors that could lead to engine failure. Here's a prioritized approach for preventative maintenance:

1. Inspect and Clean Cooling Air Passages (High-Pressure & Low-Pressure Turbines):

Reasoning: Insufficient cooling airflow to both the high-pressure and low-pressure turbines is a major red flag. This suggests potential blockages or restrictions in the cooling air passages, leading to turbine overheating and potential catastrophic failure.

Action: Thoroughly inspect all cooling air passages for both turbine stages. This includes checking for:

- **Debris:** Foreign object damage (FOD) like dirt, dust, or small parts can clog passages.
- **Carbon build-up:** Combustion byproducts can accumulate and restrict airflow.
- **Damaged seals:** Inspect seals around the turbine casings for leaks that reduce cooling efficiency.

Preventative Measure: Implement a rigorous schedule for cleaning cooling air passages based on engine usage and environmental conditions.

2. Verify & Calibrate Fan Speed Sensors and Control Systems:

Reasoning: "Corrected fan speed" implies the engine control system is attempting to adjust fan speed to compensate for some issue. A faulty sensor reading or control system malfunction could lead to improper fan operation. This can disrupt the intended airflow through the engine, impacting both cooling and thrust, potentially leading to a dangerous stall or surge.

Action:

- **Sensor Calibration:** Verify the accuracy of the fan speed sensors against manufacturer specifications. Recalibrate or replace if necessary.
- **Control System Diagnostics:** Conduct a thorough diagnostic check of the engine control system, focusing on the fan speed control loop. Look for:
 - Software glitches
 - Faulty actuators
 - Wiring issues

Preventative Measure: Regularly schedule calibration checks for fan speed sensors and comprehensive diagnostics for the engine control system.

3. Analyze Turbine Blade Condition and Clearance:

Reasoning: The combination of high SHAP values for all three factors points towards potential turbine blade distress. Insufficient cooling airflow can cause warping or damage to turbine blades. If the fan speed is also compromised, this could further exacerbate the issue by altering the pressure and flow dynamics across the turbines, leading to blade rubbing or even failure.

Action:

- **Borescope Inspection:** Conduct a visual inspection of both high-pressure and low-pressure turbine blades for signs of:
 - Overheating (discoloration, warping)
 - Cracks or damage
 - Erosion or wear
- **Clearance Measurement:** Verify turbine blade clearances are within manufacturer tolerances.

Preventative Measure: Implement a scheduled inspection program for turbine blades based on engine cycles and operating conditions. Consider more frequent inspections if the engine operates in harsh environments or experiences frequent high-power demands.

Important Note: These steps are a starting point for your investigation. Access to the engine maintenance manual, detailed sensor data, and historical maintenance logs are crucial for a comprehensive root cause analysis and the development of effective preventative maintenance procedures.

Priority	Feature	Step	Significance
1.	High-Pressure & Low-Pressure Turbines Cool Air Flow	Inspect and clean cooling air passages for blockages, debris, and leaks, ensuring proper airflow.	Prevents overheating, maintains turbine efficiency, and reduces the risk of catastrophic failure by ensuring optimal cooling.
2.	Corrected Fan Speed (CFS)	Verify and calibrate fan speed sensors, perform control system diagnostics, check for wiring or actuator issues.	Ensures proper fan speed control, preventing disruptions in airflow, and mitigating the risk of engine stall or surge.
3.	Turbine Blade Condition and Clearance	Perform borescope inspections of turbine blades for damage, overheating, and wear. Measure and verify blade clearances.	Ensures the structural integrity of turbine blades, reducing the risk of blade failure or performance degradation due to insufficient cooling.

Table A.4: Top 3 Preventive Maintenance Steps for Engine Failure Based on SHAP Values

A.1.2 Instance 2

ChatGPT4o

- **High-pressure turbines Cool air flow:** This sensor shows the highest contribution, indicating potential issues with the turbine cooling system. Insufficient cooling could lead to turbine overheating, causing blade deformation, cracking, or material degradation due to thermal stress, potentially leading to engine failure.
- **HPC Outlet Static Pressure (psia):** A significant SHAP value for HPC outlet static pressure suggests imbalances in the high-pressure compressor. This can result in improper

airflow regulation, leading to overheating and inefficiency. The issue may stem from sensor inaccuracies, blockages, or damage within the compressor stages.

- **HPC Outlet Temperature (°R):** Deviations in outlet temperature indicate inefficiencies in the high-pressure compressor. High outlet temperature may result from improper compression, potentially due to worn compressor blades, and may exacerbate thermal stress on engine components.

Given the high SHAP values for these factors—cooling, pressure regulation, and temperature control—the root cause likely lies in the turbine cooling system and the HPC's airflow and pressure management. If left unchecked, these issues can cause significant damage, accelerating engine degradation and increasing the risk of failure.

Based on the analysis, here are three prioritized actions to take:

1. Cooling System Inspection and Calibration

Action: Conduct a thorough inspection of the high-pressure turbine's cooling system, ensuring that all cooling air passages are free of blockages and wear. Recalibrate the cooling flow sensors to maintain optimal airflow as per the manufacturer's specifications.

Significance: Proper cooling is critical for preventing turbine overheating. Effective maintenance of the cooling system will reduce thermal stress and mitigate the risk of blade damage or engine failure.

Downtime: Minimal; inspections and recalibrations can be done during routine maintenance cycles.

2. HPC Outlet Pressure Sensor Calibration

Action: Recalibrate or replace the HPC outlet static pressure sensor. Inspect the compressor blades and stages for signs of damage or wear that could affect pressure regulation and airflow efficiency.

Significance: Proper regulation of HPC outlet pressure is essential for efficient airflow and engine performance. Addressing pressure imbalances will reduce the risk of overheating and pressure-related inefficiencies.

Downtime: Moderate; sensor calibration and component inspections are necessary for long-term performance.

3. HPC Outlet Temperature Sensor Calibration and Mechanical Inspection

Action: Recalibrate the HPC outlet temperature sensor. Inspect the compressor blades for wear or damage and ensure the temperature control systems are functioning within prescribed limits.

Significance: Accurate temperature regulation is essential for maintaining engine stability and preventing thermal overload. Regular inspection of temperature sensors and related components ensures the engine operates within safe thermal limits, avoiding failures.

Downtime: Moderate; sensor calibration and blade inspection can prevent catastrophic engine damage.

Claude 3.5 Sonnet

Top 3 prioritized steps for preventive maintenance based on SHAP analysis:

Priority	Sensor	Action	Significance
1. Cooling System Inspection and Calibration	High-pressure turbines Cool air flow	Inspect cooling air passages for blockages or damage. Recalibrate the cooling flow sensors to ensure optimal performance and airflow.	Proper cooling is essential to avoid overheating and material degradation in the high-pressure turbine, reducing the risk of failure.
2. HPC Outlet Pressure Sensor Calibration	HPC outlet Static pressure (psia)	Recalibrate or replace the HPC outlet pressure sensor. Inspect the high-pressure compressor for damage or blockages that may cause pressure imbalances.	Correct pressure regulation ensures efficient engine performance, minimizing the risk of overheating and performance degradation.
3. HPC Outlet Temperature Sensor Calibration and Mechanical Inspection	HPC outlet temperature (°R)	Recalibrate the temperature sensor and inspect the compressor blades for wear or damage to ensure proper temperature regulation.	Maintaining accurate outlet temperature is crucial to prevent thermal stress and ensure engine stability, avoiding potential failures.

Table A.5: Proposed Actions for Engine Maintenance and Repair Instance 2

1. High-pressure turbines Cool air flow:

Reasoning: This feature has the longest bar extending to the right, indicating the highest positive SHAP value. It suggests that the cooling system for the high-pressure turbine is critically important to the engine's performance and potential failure risk.

Step: Thoroughly inspect and service the high-pressure turbine cooling system. Check for blockages in cooling air passages, verify the integrity of seals and ducting, and ensure proper operation of cooling air control valves. Consider recalibrating the cooling air flow rates if necessary.

2. HPC outlet Static pressure:

Reasoning: This feature has the second-longest bar extending to the right, indicating a high positive SHAP value. Abnormal pressure at the high-pressure compressor outlet can signify compressor efficiency issues or potential blockages.

Step: Conduct a detailed inspection of the high-pressure compressor, focusing on blade condition, clearances, and potential obstructions. Check and calibrate pressure sensors. Analyze compressor performance maps to ensure operation within design parameters.

3. HPC outlet pressure:

Reasoning: This feature also shows a significant positive SHAP value, closely following the HPC outlet Static pressure. The presence of both static and total pressure as important factors underscores the critical nature of the high-pressure compressor's performance.

Step: Perform a comprehensive analysis of the high-pressure compressor's aerodynamic performance. This includes checking for surge margin, verifying variable geometry systems (if applicable), and ensuring proper bleed valve operation. Consider borescope inspections to check for any internal damage or deposits affecting compressor efficiency.

These steps target the most statistically significant contributors to the predicted engine behavior based on the SHAP values in the provided image. The focus on high-pressure turbine cooling and compressor performance suggests that these areas are critical for maintaining engine health and preventing potential failures.

Illama 3.1

Top 3 Preventive Maintenance Steps for Engine Failure (Based on SHAP Values):

Priority	Feature	Step	Significance
1.	High-pressure turbine Cool air flow	Thoroughly inspect and service the high-pressure turbine cooling system. Check for blockages, leaks, and recalibrate cooling air flow rates.	Ensures proper cooling of the high-pressure turbine, preventing overheating and extending component life. Critical to maintaining engine performance and reducing failure risk.
2.	HPC outlet Static pressure	Conduct detailed inspection of the high-pressure compressor, focusing on blade condition and pressure sensor calibration.	Ensures efficient compressor performance by maintaining proper pressure, preventing blockage and efficiency losses, which can degrade engine performance.
3.	HPC outlet pressure	Perform comprehensive analysis of the high-pressure compressor's aerodynamic performance, including surge margin checks, bleed valve operation, and potential internal damage inspections.	Guarantees the high-pressure compressor is operating efficiently, maintaining the proper balance of pressures to prevent engine failure and enhance longevity.

Table A.6: Top 3 Prioritized Steps for Preventive Maintenance Based on SHAP Analysis for instance 2

Given the absence of a manual, I will rely on general engineering knowledge and industry standards to provide a prioritized root cause analysis and preventive maintenance steps for the top 3 contributors to engine failure.

Rank 1: High-Pressure Turbines Cool Air Flow (Contribution to failure: High)

Root Cause Analysis: The high-pressure turbine (HPT) cool air flow is critical for maintaining the turbine's structural integrity and preventing overheating. A malfunction in the cool air flow system can lead to increased temperatures, reduced turbine efficiency, and ultimately, engine failure. Possible causes include:

- Blockages or restrictions in the cool air flow ducts or nozzles
- Malfunctioning or faulty temperature sensors
- Inadequate cooling air supply

Preventive Maintenance Steps:

- Inspect the cool air flow ducts and nozzles for any blockages or restrictions. Clean or replace them as needed.
- Check the temperature sensors for proper function and calibrate or replace them if necessary.
- Verify that the cooling air supply is adequate and functioning correctly.

Rank 2: HPC Outlet Static Pressure (Contribution to failure: Medium-High)

Root Cause Analysis: The high-pressure compressor (HPC) outlet static pressure sensor monitors the pressure at the exit of the compressor. An abnormal reading can indicate a problem with the compressor or the engine's overall performance. Possible causes include:

- Compressor blade damage or erosion
- Compressor surge or stall
- Faulty or clogged pressure sensors

Preventive Maintenance Steps:

- Inspect the compressor blades for damage or erosion. Replace them if necessary.
- Check the compressor's operating conditions to prevent surge or stall. Adjust the compressor's bleed valves or inlet guide vanes as needed.
- Verify that the pressure sensors are functioning correctly and calibrate or replace them if necessary.

Rank 3: HPC Outlet Pressure (Contribution to failure: Medium)

Root Cause Analysis: The HPC outlet pressure sensor monitors the pressure at the exit of the compressor. An abnormal reading can indicate a problem with the compressor or the engine's overall performance. Possible causes include:

- Compressor performance degradation
- Compressor seal leaks
- Faulty or clogged pressure sensors

Preventive Maintenance Steps:

- Perform a compressor performance test to identify any degradation. Adjust or replace the compressor components as needed.
- Inspect the compressor seals for leaks. Replace them if necessary.
- Verify that the pressure sensors are functioning correctly and calibrate or replace them if necessary.

Important Note: These steps are general recommendations and may not be specific to the manufacturer's requirements. It is essential to consult the manufacturer's manual or documentation for specific maintenance instructions.

Priority	Feature	Step	Significance
1.	HPT Cool Air Flow (°R)	Inspect ducts and nozzles for blockages, check temperature sensors for functionality, verify cooling air supply.	Ensures adequate cooling, preventing overheating and maintaining turbine efficiency, reducing the risk of engine failure.
2.	HPC Outlet Static Pressure (psia)	Inspect compressor blades for damage, adjust bleed valves to prevent stall, calibrate static pressure sensor.	Prevents compressor issues like surge or reduced efficiency, maintaining engine performance and reducing the risk of failure.
3.	HPC Outlet Pressure (psia)	Perform a compressor performance test, inspect seals for leaks, calibrate pressure sensors.	Ensures accurate pressure monitoring, preventing performance degradation and ensuring the engine operates efficiently.

Table A.7: Top 3 Preventive Maintenance Steps for Engine Failure Based on SHAP Values

Gemini Pro 1.5

Top 3 Preventative Maintenance Steps for Impending Engine Failure (Based on SHAP Values):

Given: High-Pressure Turbine (HPT) Cool Air Flow, HPC Outlet Static Pressure, and HPC Outlet Pressure sensors show the highest SHAP values, indicating their critical contribution to the predicted engine failure.

Assumptions: Since a manual is unavailable, we'll assume these sensors are crucial for maintaining optimal HPT performance and cooling. We'll prioritize based on potential failure severity and impact on flight safety.

Here are the top 3 preventative maintenance steps:

1. Inspect and Clean/Replace HPT Cooling Air Flow System (Highest Priority):

Reasoning: HPT Cool Air Flow having the highest SHAP value suggests a severe issue with the turbine's cooling system. This is critical as insufficient cooling can lead to rapid material degradation, blade deformation, and catastrophic engine failure.

Root Cause Analysis: Possible causes include blockage in the cooling air ducts, a malfunctioning cooling air fan, or a faulty HPT Cool Air Flow sensor itself.

Preventative Action:

- **Inspect** all cooling air ducts for obstructions like debris, dirt, or foreign object damage (FOD).
- **Check** the cooling air fan for proper operation and airflow.
- **Verify** the HPT Cool Air Flow sensor's accuracy and replace it if faulty.

2. Inspect and Calibrate HPC Outlet Static Pressure Sensor:

Reasoning: HPC Outlet Static Pressure directly impacts the pressure ratio across the HPT, influencing its efficiency and thrust output. An inaccurate reading could lead to improper fuel-air mixture, reduced performance, or even a compressor stall.

Root Cause Analysis: Potential causes include sensor drift, blockage in the static pressure port, or damage to the sensor itself.

Preventative Action:

- **Inspect** the static pressure port for any blockage and clean if necessary.
- **Calibrate** the HPC Outlet Static Pressure sensor according to manufacturer specifications (even if a manual isn't available, general calibration procedures for similar sensors can be consulted).

3. Inspect and Calibrate HPC Outlet Pressure Sensor:

Reasoning: HPC Outlet Pressure is another critical parameter for engine performance and safety. A faulty sensor could lead to incorrect thrust calculations, potential overspeeding, or improper engine control responses.

Root Cause Analysis: Similar to the static pressure sensor, potential causes include sensor drift, blockage, or damage.

Preventative Action:

- **Inspect** the pressure tap and connecting lines for any blockage or leaks.
- **Calibrate** the HPC Outlet Pressure sensor to ensure accurate readings.

Important Note: These steps provide a starting point. Without a specific engine manual, consult with experienced aircraft mechanics and seek information on similar engine models for more detailed procedures. Always prioritize safety and err on the side of caution when dealing with potential engine failures.

Priority	Feature	Step	Significance
1.	HPT Cooling Air Flow (°R)	Inspect cooling air ducts for debris or damage, check cooling air fan operation, and verify sensor accuracy.	Ensures adequate cooling of the HPT, preventing material degradation and catastrophic engine failure caused by overheating.
2.	HPC Outlet Static Pressure (psia)	Inspect and clean static pressure ports, calibrate the sensor for accurate pressure readings.	Maintains correct pressure ratio across the HPT, ensuring optimal performance and preventing compressor stall or reduced efficiency.
3.	HPC Outlet Pressure (psia)	Inspect pressure tap for blockages or leaks, calibrate the sensor to ensure proper thrust calculations and engine control.	Ensures accurate engine control and prevents improper responses like overspeeding or thrust miscalculations, improving safety and performance.

Table A.8: Top 3 Preventative Maintenance Steps for Impending Engine Failure Based on SHAP Values

ChatGPT4o

From the SHAP values shown in the image:

- **Ratio of fuel flow to Ps30 (pps/psia):** This sensor shows the highest contribution, indicating potential issues in fuel regulation. Problems here can lead to inefficient combustion, fuel starvation, or over-fueling, affecting engine performance and increasing the risk of failure.
- **HPC outlet Static pressure (psia):** A significant SHAP value here suggests pressure imbalances in the high-pressure compressor, which can result in airflow inefficiencies and overheating. This could be caused by sensor issues, blockages, or compressor degradation.
- **Physical core speed (rpm):** Deviations in core speed point to potential mechanical imbalances or wear in the turbine or compressor, leading to instability in power generation and engine performance.

Given the high SHAP values for fuel flow, pressure regulation, and core speed, the root cause of the issue likely lies within the fuel delivery system, pressure management in the HPC, and mechanical control of engine speed. These inefficiencies, if not corrected, will accelerate engine degradation and could result in failure.

Based on the analysis, here are three prioritized actions to take:

1. Fuel System Inspection and Calibration

Action: Inspect the entire fuel delivery system, including fuel injectors, pumps, and control units, for any blockages, wear, or malfunction. Recalibrate the fuel flow system to ensure accurate fuel-to-pressure ratios as per the manufacturer's specifications.

Significance: Efficient fuel flow is critical for maintaining combustion quality and engine power. Proper calibration will prevent misfueling and reduce engine stress, avoiding potential failures.

Downtime: Minimal; system checks and recalibration are essential and can be done relatively quickly.

2. HPC Outlet Pressure Sensor Calibration

Action: Re-calibrate or replace the HPC outlet static pressure sensor. Inspect the compressor for signs of wear, blockages, or damage that may be causing pressure imbalances.

Significance: Proper pressure regulation ensures efficient airflow through the engine, critical for optimal performance. Addressing this will reduce the risk of overheating and pressure-related failures.

Downtime: Moderate; sensor calibration and component inspection are necessary for long-term performance.

3. Core Speed Sensor Calibration and Mechanical Inspection

Action: Re-calibrate the core speed sensor and inspect mechanical components like the compressor and turbine for any wear or imbalance that could affect engine speed.

Significance: Core speed is a key factor in engine stability and power generation. Ensuring the speed sensor and related components are functioning correctly prevents mechanical failures and ensures engine reliability.

Downtime: Moderate; sensor calibration and mechanical inspection are necessary to prevent catastrophic engine damage.

Priority	Sensor	Action	Significance
1. Fuel System Inspection and Calibration	Ratio of fuel flow to Ps30 (pps/psia)	Inspect the fuel system for blockages, wear, or leaks. Recalibrate the system to maintain the proper ratio of fuel flow to pressure.	Ensuring proper fuel flow is critical for efficient combustion and preventing misfueling, which can cause power loss and engine failure.
2. HPC Outlet Pressure Sensor Calibration	HPC outlet Static pressure (psia)	Re-calibrate or replace the HPC outlet pressure sensor. Inspect the high-pressure compressor for signs of wear or blockage that could cause pressure imbalances.	Proper pressure regulation is crucial for efficient engine operation. Imbalances can lead to overheating and significant engine stress.
3. Core Speed Sensor Calibration and Mechanical Inspection	Physical core speed (rpm)	Re-calibrate the core speed sensor and inspect the mechanical components for wear, ensuring proper speed regulation.	Maintaining accurate core speed is essential for engine power generation and stability, preventing catastrophic failures.

Table A.9: Proposed Actions for Engine Maintenance and Repair for Instance 3

Sonnet Claude 3.5

Top 3 prioritized steps for preventive maintenance based on SHAP analysis:

1. Ratio of fuel flow to Ps30 (pps/psia):

Reasoning: It suggests that the fuel-to-air ratio is critically important to the engine's performance and potential failure risk.

Step: Thoroughly inspect and calibrate the fuel metering system. Check fuel nozzles for clogs or uneven spray patterns. Verify the accuracy of fuel flow sensors and pressure transducers. Adjust the fuel control unit to optimize the fuel-to-air ratio across various operating conditions.

2. HPC outlet Static pressure (psia):

Reasoning: This feature has the second-longest bar extending to the right, indicating a high positive SHAP value. Abnormal static pressure at the high-pressure compressor outlet can signify compressor efficiency issues or potential blockages.

Step: Conduct a detailed inspection of the high-pressure compressor, focusing on blade condition, clearances, and potential obstructions. Calibrate pressure sensors and analyze compressor performance maps to ensure operation within design parameters.

3. Physical fan speed (rpm):

Reasoning: This feature shows a significant negative SHAP value, indicated by the long bar extending to the left. Abnormal fan speeds can impact overall engine performance and efficiency.

Step: Investigate the fan section thoroughly. Check for any damage to fan blades, foreign object damage, or imbalance. Verify the integrity of the fan shaft and bearings. Inspect and calibrate the fan speed sensors. Analyze the relationship between fan speed and other engine parameters to ensure proper engine matching.

These steps target the most statistically significant contributors to the predicted engine behavior based on the SHAP values in the provided image. The focus on fuel system performance, compressor outlet conditions, and fan operation suggests that these areas are critical for maintaining engine health and preventing potential failures. The interplay between these factors indicates a complex system behavior that requires careful analysis and maintenance.

Priority	Feature	Step	Significance
1.	Ratio of fuel flow to Ps30 (pps/psia)	Thoroughly inspect and calibrate the fuel metering system. Check fuel nozzles, verify fuel flow sensors, and adjust the fuel control unit for optimized fuel-to-air ratio.	Ensures optimal combustion efficiency, reducing the risk of engine performance degradation due to fuel-to-air imbalance, which is critical for preventing engine failures.
2.	HPC outlet Static pressure (psia)	Conduct detailed inspection of the high-pressure compressor, calibrate pressure sensors, and ensure proper operation by analyzing compressor performance maps.	Maintains compressor efficiency by keeping static pressure within design parameters, preventing blockages and improving overall engine health.
3.	Physical fan speed (rpm)	Inspect the fan section, checking for damage, imbalance, and sensor accuracy. Ensure the fan operates within the expected range, matching engine demands.	Ensures stable engine operation by maintaining proper fan speeds, reducing the risk of performance degradation or failure due to fan-related issues.

Table A.10: Top 3 Prioritized Steps for Preventive Maintenance Based on SHAP Analysis for instance 3

llama 3.1

Top 3 Preventive Maintenance Steps for Engine Failure (Based on SHAP Values):

Given the SHAP values, I've prioritized the top 3 steps to address the critical contributors to the engine failure. Since there is no manual, I'll provide general guidance based on industry standards and best practices.

Top 3 Steps:

1. Ratio of Fuel Flow to Ps30: The ratio of fuel flow to Ps30 (compressor pressure at 30% of the compressor's length) has the highest contribution to the engine failure. This suggests an issue with fuel management and compressor performance.

Root Cause Analysis: Potential causes include:

- Fuel system malfunction or blockage
- Incorrect fuel flow settings
- Compressor performance degradation

Preventive Maintenance Steps:

- Inspect and clean the fuel system to ensure proper flow
- Check and adjust fuel flow settings according to the engine's design specifications
- Perform compressor performance tests to identify potential issues

2. HPC (High-Pressure Compressor) Outlet Static Pressure: The HPC outlet static pressure has a significant contribution to the engine failure. This suggests an issue with compressor performance and potential instability.

Root Cause Analysis: Potential causes include:

- Compressor blade damage or wear
- Incorrect compressor vane settings
- Airflow blockage or restriction

Preventive Maintenance Steps:

- Inspect compressor blades for damage or wear and replace if necessary
- Check and adjust compressor vane settings to ensure optimal performance
- Inspect the airflow path for blockages or restrictions and clean as needed

3. Corrected Fan Speed: The corrected fan speed has a moderate contribution to the upcoming engine failure. This suggests an issue with fan performance and potential imbalance.

Root Cause Analysis: Potential causes include:

- Fan blade imbalance or damage
- Incorrect fan speed settings
- Bearing wear or failure

Preventive Maintenance Steps:

- Inspect fan blades for imbalance or damage and rebalance or replace as needed
- Check and adjust fan speed settings to ensure optimal performance
- Inspect fan bearings for wear or failure and replace if necessary

By addressing these top 3 contributors, you should be able to mitigate the risk of engine failure. However, please note that a comprehensive maintenance manual and manufacturer's guidance are essential for specific and accurate maintenance procedures.

Gemini 1.5 Pro

Top 3 Preventive Maintenance Steps for Impending Engine Failure:

Based on the provided SHAP values, the ratio of fuel flow to Ps30 is the most significant contributor to the potential engine failure. This suggests a strong likelihood of issues within the fuel control system or the engine core's ability to handle the incoming fuel flow. While HPC outlet static pressure and corrected fan speed also contribute, their lower SHAP values indicate a likely secondary role influenced by the primary issue.

Therefore, focusing on the fuel flow and its relation to Ps30 (likely a pressure sensor in the combustion chamber or high-pressure turbine) is paramount.

Priority	Feature	Step	Significance
1.	Ratio of Fuel Flow to Ps30 (pps/psia)	Inspect and clean the fuel system, check and adjust fuel flow settings, and perform compressor performance tests.	Ensures proper fuel management and compressor performance, addressing the most critical contributor to engine failure.
2.	HPC Outlet Static Pressure (psia)	Inspect compressor blades for wear, check vane settings, and clean airflow paths.	Maintains compressor performance and prevents instability, reducing the risk of engine inefficiency or failure.
3.	Corrected Fan Speed (rpm)	Inspect fan blades for imbalance, check fan speed settings, and inspect bearings for wear.	Ensures proper fan performance, preventing imbalances that could degrade engine efficiency and increase failure risks.

Table A.11: Top 3 Preventive Maintenance Steps for Engine Failure Based on SHAP Values

Here are the top 3 preventive maintenance steps, prioritizing the most critical:

1. Fuel Control System Inspection and Calibration:

Rationale: The high SHAP value for "Ratio of fuel flow to Ps30" strongly suggests an imbalance. This could be due to a faulty fuel control unit, clogged fuel nozzles, incorrect fuel pressure, or a leak in the fuel lines.

Action:

- **Inspect:** Thoroughly inspect the entire fuel control system, including the fuel pump, control unit, fuel lines, and fuel nozzles for any signs of damage, wear, blockage, or leaks.
- **Test:** Conduct performance tests on the fuel control unit and fuel pump to ensure they are operating within the manufacturer's specified parameters.
- **Calibrate:** Using a calibrated fuel pressure gauge, adjust the fuel pressure regulator to ensure the correct fuel pressure is supplied to the engine.

2. Combustion Chamber and High-Pressure Turbine Inspection:

Rationale: The "Ps30" sensor likely monitors pressure in a critical area like the combustion chamber or high-pressure turbine. An imbalance in the fuel-to-pressure ratio could point to an issue in these areas affecting combustion efficiency and potentially damaging engine components.

Action:

- **Inspect:** Visually inspect the combustion chamber for any signs of overheating, cracking, or damage to the liner, fuel nozzles, or igniters. Inspect the high-pressure turbine blades for signs of wear, damage, or deposits.
- **Clean:** If deposits are found, carefully clean the affected components according to the manufacturer's recommendations.

3. HPC Outlet Static Pressure and Fan Speed Analysis:

Rationale: While these factors have a lower SHAP value, their contribution shouldn't be ignored. They could be symptoms of the primary issue or indicate a separate developing problem.

Action:

- **Analyze:** Compare the current HPC outlet static pressure and corrected fan speed readings to the engine manufacturer's specifications.
- **Investigate:** If deviations are found, further investigate potential causes, such as issues with the high-pressure compressor blades, variable geometry system, or foreign object damage.

By focusing on these top three steps, you address the most likely root causes of the impending engine failure based on the SHAP value analysis.

Priority	Feature	Step	Significance
1.	Ratio of Fuel Flow to Ps30 (pps/psia)	Inspect and test the fuel control system, including fuel pump, control unit, lines, and nozzles. Calibrate fuel pressure regulator.	Ensures proper fuel flow to prevent engine performance issues or failure due to imbalances in fuel-to-air ratios.
2.	Ps30 Pressure Sensor (psia)	Inspect the combustion chamber and high-pressure turbine blades for damage or deposits. Clean components if necessary.	Addresses potential damage to critical engine components caused by incorrect combustion efficiency, reducing failure risks.
3.	HPC Outlet Static Pressure and Fan Speed	Analyze current static pressure and fan speed, investigate any deviations from manufacturer specifications.	Ensures compressor and fan efficiency, reducing secondary risks associated with incorrect pressure or fan speed readings.

Table A.12: Top 3 Preventive Maintenance Steps for Engine Failure Based on SHAP Values

A.1.3 Instance 4

ChatGPT4o

From the SHAP values shown in the image:

- **HPC Outlet Temperature (°R):** This sensor shows the highest contribution, indicating significant thermal stress within the high-pressure compressor section. High outlet temperatures can suggest inefficiency, compressor wear, or blocked cooling passages, which may accelerate engine degradation.
- **Ratio of Fuel Flow to Ps30 (pps/psia):** A high SHAP value here indicates possible fuel system issues, such as incorrect fuel delivery or pressure regulation. Improper fuel flow can lead to inefficient combustion, causing performance losses and increasing the likelihood of engine failure.
- **Corrected Core Speed (rpm):** Deviations in corrected core speed highlight potential mechanical or control issues affecting the rotational speed of the compressor and turbine sections. This can lead to power generation inefficiencies and imbalanced operation, potentially causing engine instability.

Given the high SHAP values for these critical sensors, the root cause likely lies in a combination of thermal, fuel flow, and mechanical control issues. These inefficiencies, if not addressed, will accelerate engine wear and increase the risk of failure.

Based on the analysis, here are three prioritized actions to take:

1. High-Pressure Compressor (HPC) Inspection and Temperature Sensor Calibration

Action: Inspect the HPC section for signs of wear, debris, or cooling blockages. Recalibrate or replace the HPC outlet temperature sensor to ensure accurate monitoring. Pay close attention to compressor blade wear and cooling efficiency.

Significance: Addressing high temperatures is critical to prevent overheating and reduce thermal stress on the compressor and turbine stages, which can lead to catastrophic engine failure.

Downtime: Moderate; requires detailed inspection and possible component replacement.

2. Fuel System Inspection and Calibration

Action: Inspect the entire fuel system, including fuel injectors, pumps, and control units, for blockages, wear, or malfunctions. Recalibrate the fuel flow system to ensure accurate fuel-to-pressure ratios as per the manufacturer's specifications.

Significance: Efficient fuel flow is critical for maintaining combustion quality and engine power. Proper calibration will prevent misfueling and reduce engine stress, avoiding potential failures.

Downtime: Minimal; system checks and recalibration are essential and can be done relatively quickly.

3. Core Speed Sensor Calibration and Mechanical Inspection

Action: Recalibrate the core speed sensor and inspect mechanical components like the compressor and turbine for any wear or imbalance that could affect engine speed.

Significance: Core speed is a key factor in engine stability and power generation. Ensuring the speed sensor and related components are functioning correctly prevents mechanical failures and ensures engine reliability.

Downtime: Moderate; sensor calibration and mechanical inspection are necessary to prevent catastrophic engine damage.

Priority	Sensor	Action	Significance
1. HPC Inspection and Temperature Sensor Calibration	HPC outlet temperature (°R)	Inspect HPC section for wear, cooling issues, or debris. Recalibrate temperature sensor and assess cooling efficiency.	High temperatures lead to thermal stress and potential component failure. Addressing this prevents overheating and compressor degradation.
2. Fuel System Inspection and Calibration	Ratio of fuel flow to Ps30 (pps/psia)	Inspect the fuel system for blockages, wear, or leaks. Recalibrate the system to maintain the proper ratio of fuel flow to pressure.	Ensuring proper fuel flow is critical for efficient combustion and preventing misfueling, which can cause power loss and engine failure.
3. Core Speed Sensor Calibration and Mechanical Inspection	Corrected core speed (rpm)	Recalibrate the core speed sensor and inspect the mechanical components for wear, ensuring proper speed regulation.	Maintaining accurate core speed is essential for engine power generation and stability, preventing catastrophic failures.

Table A.13: Proposed Actions for Engine Maintenance and Repair Instance 4

Claude 3.5 Sonnet

Top 3 prioritized steps for preventive maintenance based on SHAP analysis:

1. Investigate and address HPC outlet temperature issues:

Reasoning: The HPC (High Pressure Compressor) outlet temperature has by far the highest positive SHAP value, significantly larger than any other feature. This indicates it's the most critical factor contributing to potential engine failure. Abnormally high temperatures at the HPC outlet can indicate:

- Compressor efficiency loss
- Potential damage to compressor components
- Risk of thermal damage to downstream components

This extreme temperature deviation could lead to accelerated wear, reduced engine efficiency, or even catastrophic failure if not addressed promptly.

2. Evaluate and calibrate the fuel metering system:

Reasoning: The "Ratio of fuel flow to Ps30" shows the second highest positive SHAP value. This ratio represents the relationship between fuel flow and combustor inlet pressure. An abnormal ratio suggests:

- Potential fuel system malfunctions
- Combustion inefficiencies
- Risk of engine surge or stall

Improper fuel-air ratios can significantly impact engine performance and reliability.

3. Assess corrected core speed anomalies:

Reasoning: The corrected core speed (rpm) shows a notable negative SHAP value, indicating it's lower than expected. This could suggest:

- Potential power section issues
- Mismatched compressor-turbine operation
- Degraded overall engine performance

Abnormal core speeds can indicate underlying mechanical problems or control system issues that need attention.

These steps target the most statistically significant contributors to the predicted failure based on the SHAP values in the provided image. The combination of temperature anomalies, fuel system issues, and core speed discrepancies suggests a complex interplay of factors that could lead to engine malfunction if not addressed.

llama 3.1

Top 3 Preventive Maintenance Steps Based on SHAP Values:

Based on the provided SHAP values, I have verified and prioritized the top contributors to the upcoming engine failure. Here are the top 3 steps to address the potential failure:

Step 1: Investigate HPC Outlet Temperature Anomaly (SHAP value: High)

The High-Pressure Compressor (HPC) outlet temperature has the highest SHAP value, indicating a significant contribution to the predicted engine failure. This anomaly suggests a potential issue with the compressor's performance or cooling system.

Root Cause Analysis: Possible causes include:

Priority	Feature	Step	Significance
1.	HPC outlet temperature (°R)	Investigate and address HPC outlet temperature issues. Inspect for compressor efficiency loss and thermal damage to components.	Prevents accelerated wear, improves efficiency, and reduces the risk of catastrophic engine failure due to high temperature.
2.	Ratio of fuel flow to Ps30 (pps/psia)	Evaluate and calibrate the fuel metering system. Ensure proper fuel-air ratios by checking fuel system malfunctions and combustion inefficiencies.	Improves combustion efficiency, reduces the risk of engine surge or stall, and prevents performance degradation.
3.	Corrected core speed (rpm)	Assess corrected core speed anomalies. Investigate power section, compressor-turbine operation, and overall engine performance.	Maintains proper engine operation by addressing potential core speed discrepancies, improving stability and performance.

Table A.14: Top 3 Prioritized Steps for Preventive Maintenance Based on SHAP Analysis for instance 4

- Compressor blade damage or fouling
- Cooling system malfunction or blockage
- Incorrect compressor bleed valve setting

Preventive Maintenance:

- Inspect the HPC compressor blades for damage or fouling and clean or replace as necessary.
- Check the cooling system for any blockages or leaks and ensure proper coolant flow.
- Verify the compressor bleed valve setting and adjust according to manufacturer's guidelines (if available).

Step 2: Adjust Fuel-to-PS30 Ratio (SHAP value: Medium-High)

The ratio of fuel to PS30 (Power Setting 30) has a significant SHAP value, indicating an imbalance in the fuel-to-air mixture. This can lead to engine performance issues, reduced efficiency, and increased emissions.

Root Cause Analysis: Possible causes include:

- Faulty fuel flow sensor or metering unit
- Incorrect fuel-to-air ratio setting
- Clogged fuel nozzles or injectors

Preventive Maintenance:

- Check the fuel flow sensor and metering unit for accuracy and replace if necessary.
- Verify the fuel-to-air ratio setting and adjust according to manufacturer's guidelines (if available).
- Inspect and clean the fuel nozzles and injectors to ensure proper fuel flow.

Step 3: Verify Corrected Speed and HPC Settings (SHAP value: Medium)

The corrected speed and HPC settings have a moderate SHAP value, indicating a potential issue with engine performance or compressor efficiency.

Root Cause Analysis: Possible causes include:

- Incorrect engine speed or compressor setting
- Faulty speed sensor or tachometer
- Compressor performance degradation

Preventive Maintenance:

- Verify the engine speed and compressor setting and adjust according to manufacturer's guidelines (if available).
- Check the speed sensor and tachometer for accuracy and replace if necessary.
- Monitor compressor performance and adjust the HPC settings to optimize efficiency.

These three steps address the most critical contributors to the predicted engine failure. By investigating and addressing these issues, the risk of engine failure can be significantly reduced.

Priority	Feature	Step	Significance
1.	HPC Outlet Temperature (°R)	Inspect HPC blades for damage or fouling, check the cooling system for blockages or leaks, and recalibrate the bleed valve.	Prevents excessive temperatures caused by compressor performance or cooling issues, reducing the risk of engine failure.
2.	Ratio of Fuel to PS30 (pps/psia)	Inspect and recalibrate the fuel flow sensor and metering unit, clean or replace clogged fuel nozzles, and verify the fuel-to-air ratio setting.	Maintains optimal fuel-air mixture, ensuring engine efficiency, and reducing emissions and the risk of overheating.
3.	Corrected Speed and HPC Settings	Verify engine speed and HPC settings, inspect speed sensor accuracy, and monitor compressor performance.	Ensures accurate speed control and compressor efficiency, enhancing engine performance and reducing the risk of failure.

Table A.15: Top 3 Preventive Maintenance Steps Based on SHAP Values

Gemini 1.5 Pro

Top 3 Preventive Maintenance Steps Based on SHAP Values:

Given the high SHAP values indicating potential engine failure, here's a prioritized action plan focusing on the top 3 contributors:

1. **HPC Outlet Temperature:** The disproportionately high SHAP value for HPC outlet temperature points to a critical issue within the High-Pressure Compressor (HPC) section. This could indicate:

Root Cause Analysis:

- **Most Likely:** Deteriorating turbine blade health (creep, cracking, erosion) leading to reduced efficiency and increased exhaust gas temperature.
- **Other Possibilities:** Faulty cooling air flow, incorrect fuel-air mixture, damaged combustion chamber lining, sensor malfunction.

Preventive Maintenance:

- **Immediately inspect:** HPC turbine blades for damage, fouling, or wear. Verify cooling air passages are clear.

- **Replace or repair:** Any damaged components according to manufacturer specifications.
- **Recalibrate:** HPC outlet temperature sensor following manufacturer procedures.

2. **Ratio of Fuel to PS30:** This value likely refers to the fuel flow in relation to a specific engine pressure (PS30). A high SHAP value here suggests:

Root Cause Analysis:

- **Most Likely:** Fuel control unit malfunction, clogged fuel nozzles, incorrect fuel pressure, leak in fuel lines.
- **Other Possibilities:** Faulty pressure sensor (PS30), inaccurate fuel flow measurement.

Preventive Maintenance:

- **Inspect:** Fuel control unit for proper operation, fuel lines for leaks, and fuel nozzles for clogging.
- **Test:** Fuel pressure and compare to manufacturer specifications.
- **Clean or replace:** Any faulty components as needed.

3. **Corrected Speed and HPC (Lower End):** This combination suggests an issue with the HPC's operation at lower speeds or during transitions, potentially impacting overall engine performance.

Root Cause Analysis:

- **Most Likely:** Variable stator vane malfunction, foreign object damage (FOD) to the HPC's first few stages, bearing wear.
- **Other Possibilities:** Control system error affecting speed management.

Preventive Maintenance:

- **Inspect:** HPC variable stator vanes for proper movement and freedom, check for FOD in the initial stages.
- **Evaluate:** HPC bearing health and lubrication.
- **Test:** Engine control system for accurate speed control throughout the operating range.

Important Note: This analysis is based on limited information. Access to the aircraft's maintenance manual and further data analysis are crucial for a complete and accurate diagnosis.

Priority	Feature	Step	Significance
1.	HPC Outlet Temperature (°R)	Inspect HPC turbine blades for damage or wear, ensure cooling air passages are clear, recalibrate temperature sensor.	Prevents excessive temperatures caused by turbine blade deterioration or cooling system faults, reducing the risk of engine failure.
2.	Ratio of Fuel to PS30 (pps/psia)	Inspect fuel control unit, check fuel lines for leaks, clean fuel nozzles. Test fuel pressure against manufacturer specifications.	Ensures proper fuel delivery, preventing overheating, and maintains the correct fuel-to-air ratio to avoid engine performance issues.
3.	Corrected Speed and HPC	Inspect variable stator vanes, check for FOD in HPC stages, evaluate bearing health, and ensure proper speed control.	Enhances overall engine performance by ensuring that the HPC operates correctly, particularly during transitions or at lower speeds.

Table A.16: Top 3 Preventive Maintenance Steps Based on SHAP Values