Imperial College London
Department of Earth Science and Engineering
MSc in Geo-energy with Machine Learning and Data Science

Independent Research Project
Project Plan

Developing a Fine-Tuned Time Series Model for Automated Test Documentation Generation and Anomaly Detection

Ashwin Vel
Email: ashwin.vel24@imperial.ac.uk
GitHub username: `irp-av1824`
Repository: `https://github.com/ese-ada-lovelace-2024/irp-av1824`

Supervisors: Andy Nelson
Prof. Martin Blunt

June 2025

Contents

1 AI Acknowledgment Statement

The following generative AI tool was utilized in preparing this report:

- **Tool Name and Version**: ChatGPT-o3
- **Publisher or Provider**: OpenAI
- **URL of the Service**: `https://chat.openai.com`
- **Usage Description**: I've used ChatGPT to generate ideas on how to apply anomaly detection in time series data and to modify user interface screenshots to include additional functionality boxes clearly aligned with existing UI design patterns.
- **Original Work Confirmation**: I confirm that all the submitted work in this report is my own, despite the assistance received from the generative AI tool stated above.

2 Abstract

Manual documentation of well completion tools testing is often time-consuming, inconsistent, and prone to errors, which could lead to tool failure. This project aims to automate anomaly detection and quality reporting for well-completion tools, integrating AVEVA DataHub streams into a streamlined web application. The methodology involves developing an Isolation Forest-based unsupervised machine learning model trained on historical, unlabeled sensor data. A Python API will serve anomaly results to an ASP.NET web interface, enabling one-click PDF report generation. If feasible, an LSTM Auto-Encoder will enhance detection accuracy. Expected outcomes include quicker anomaly identification, reduced manual analysis effort, and standardized, reliable compliance reports demonstrating how advanced anomaly detection can improve operational efficiency and data-driven decision-making in the energy sector.

3 Introduction

3.1 Background and Problem Description

Currently, the test data from well completion tools are being stored in AVEVA Datahub such as battery voltage, upstream pressure and temperature motor speeds. As of now, there isn't any way for engineers who do the testing to automatically validate if the

results they've obtained from these test deviates from normal values, which could either signal potential tool failure or a mistake in the testing process. This process is largely manual which is by plotting or downloading the CSV files. Most of the time, this step is skipped altogether as testers heavily rely on their engineering judgement if these values are in acceptable range, increasing the chance of human oversight. This leads to delayed responses and potentially operational risk.

The main goal of my project is to build a web-based system that can detect an anomaly automatically and generate a PDF compliance report. My main challenge in this project is that the data is unlabelled, so there are no ground truth to tell if the value is an anomaly. This research is trying to explore if machine learning can be used to reliably detect anomalies in unlabelled time-series data and have these findings automatically populated in a compliance document.

3.2 Review of Existing Work

3.2.1 Challenges in Anomaly Detection

The main challenges in anomaly detection for time series data is that anomalies are rare. Because of this, it's hard to evaluate how good a model is, since there's usually no proper ground truth. The best we can do is get domain experts to label some of the data, but even that is subjective. In unsupervised anomaly approaches where anomalies show up as deviations from predictions, it is found that most thresholding methods (mean error and standard deviations) don't work well in practice [1]. This shows there's still a lot of room for improvement in how we set thresholds in anomaly detection.

3.2.2 Model Selection

Isolation Forest (IF) and Support Vector Machines (SVM) are among the most commonly used models for anomaly detection in industrial sensor data, particularly when labelled data is scarce. Isolation Forest, in particular, is favored due to its linear time complexity, low memory usage, and simple implementation [2].

However, traditional models like IF may produce high false positive rates in datasets with strong seasonal patterns. In such cases, deep learning-based approaches such as Long Short-Term Memory (LSTM) autoencoders [3] and Deep Support Vector Data Description (Deep SVDD) [4] offer better accuracy but with significantly higher computational demands and complex hyperparameter tuning requirements.

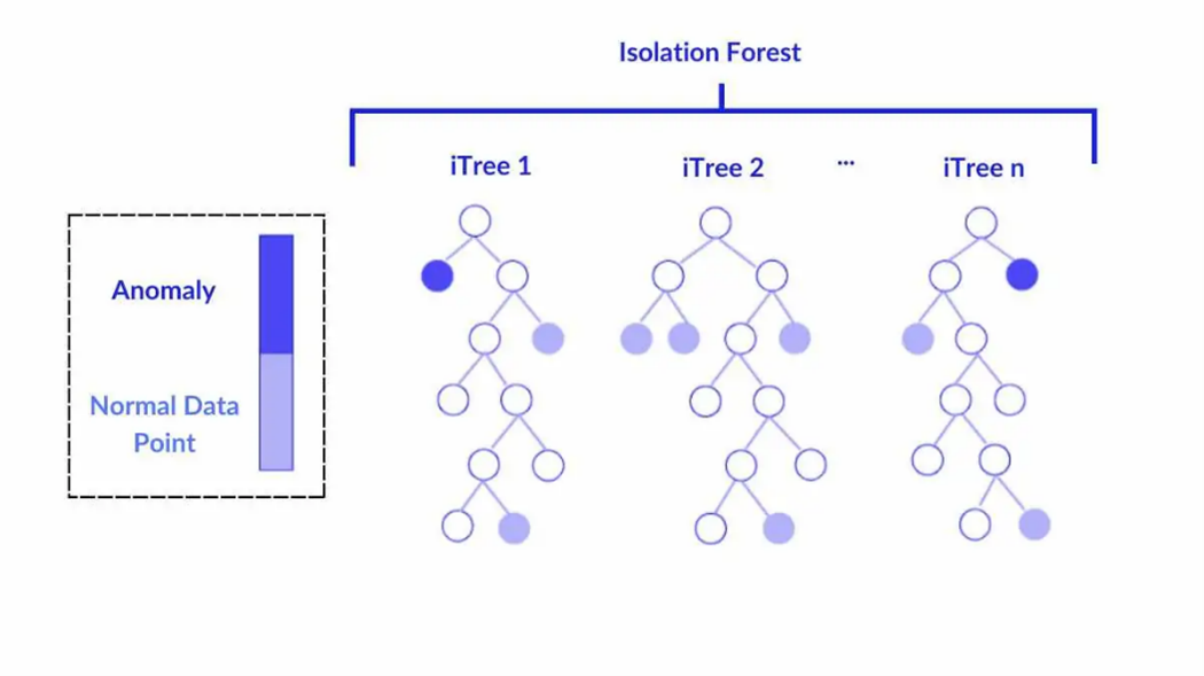

Figure 1: Isolation Forest Architecture

The novelty of this project lies in developing an integrated web-based platform that brings together DataHub ingestion, unsupervised anomaly detection, and one-click PDF report generation within the existing TAQA interface for well completion tool validation.

3.2.3 Oil & Gas Case Studies

IF are introduced in recent oilfield application where the algorithm is used to predict ESP failure in water injection wells. Results show IF could predict accurately 21 out of 45 ESP failure events when tested on historical data [5]. Febrita et al. developed a deep learning method using LSTM to detect anomaly using real time data [6]. If the delta of Loss MAE between historical and current data exceeds a set threshold, it flags an anomaly.

Other methods include using a stacked spatial-temporal autoencoder (Sst-AE) to enhance spatiotemporal features in industrial multi-sensor data [7]. An adaptive dynamic thresholding method trains the model, and anomalies are identified by comparing enhanced features against cluster centers derived from high-dimensional K-means clustering.

3.3 Objectives

The project will include software development, API integration, and the application of a time-series model for anomaly detection in an industry setting. The key objective:

- Creating a fully operational web-application allowing users to generate test documentation based on minimal user input.
- The application needs to integrate with data storage though external APIs.
- Anomaly detection module for identifying irregularities in test outcomes and comparing test results to historical baseline.

3.4 Significance

Documentation of quality test results is a critical step in ensuring well completion tools comply with industry standards. Manual documentation is often time-consuming and prone to errors. Automating this process will not only save time but also enhance the reliability of compliance checks.

Additionally, leveraging historical test data for anomaly detection can enhance the system's capabilities in detecting faulty equipment and design irregularities, reducing human-prone judgment error.

4 Methodology

4.1 System Overview

The purpose of this tool is to create a one-click PDF report generator from the data of the well completion tool tested with embedded anomaly flags.

Figure 2: Web-System Architecture

As seen above, the website will be written primarily in C# using .NET MVC framework to integrate with the current deployed web API. However, the anomaly detection module will be written in Python and these models would be wrapped using FastAPI so that the models can be exposed to .NET site calls with its own landing page.

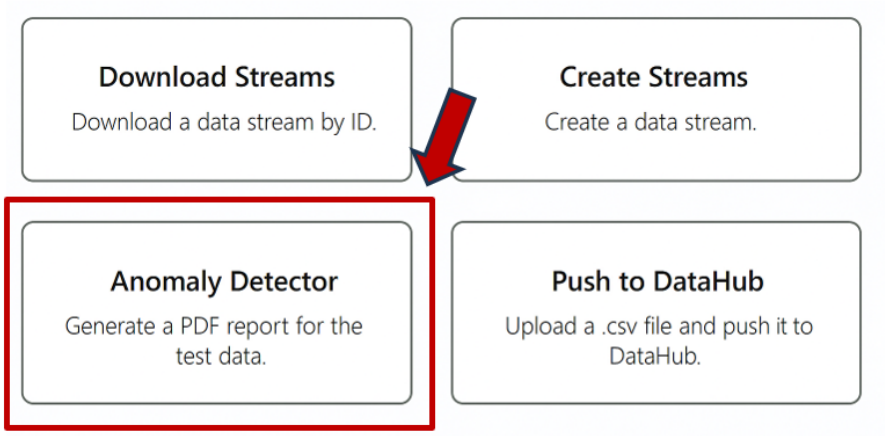

Figure 3: Mock-up of DataHub API Page

4.2 Data Collection

The primary data source for this project will be from AVEVA DataHub since all the historical data from the past tests had been stored there. The data can be downloaded from the Download Stream landing page as illustrated in the Figure 3. Access token (TenantID, ClientID) for AVEVA DataHub will be read from the `appsettings.json` file, so the C# backend will create an authenticated client. Future work can look into streamlining the login credentials directly into the webpage when the client has no access to company secured network.

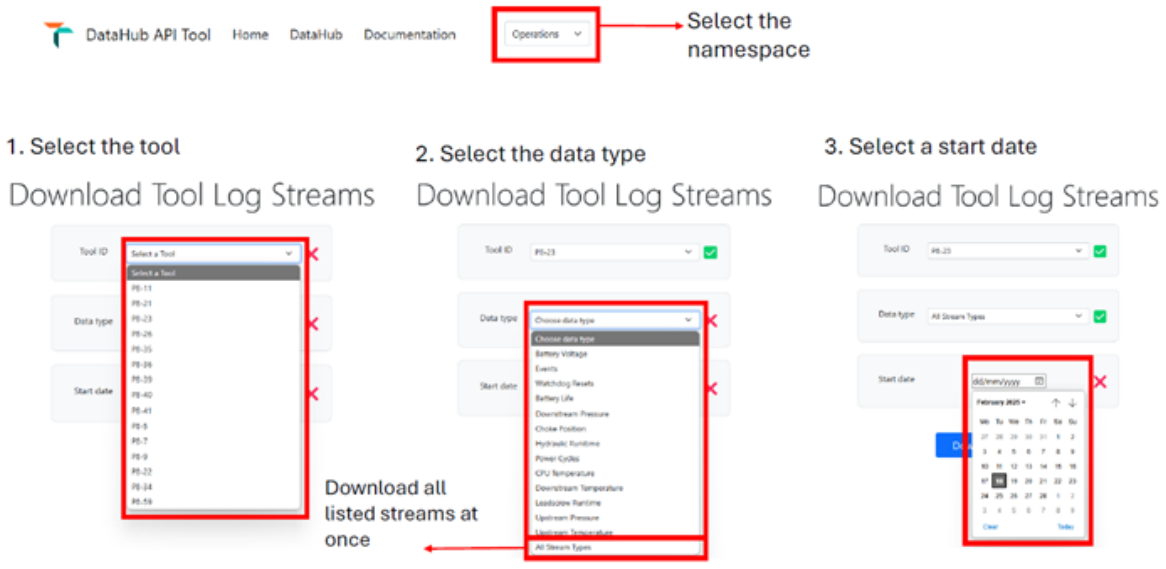

Figure 4: Screenshot from DataHub API 'Download Stream'

For the training data, I have observed there is a huge repository of data available. Training on all streams would be time consuming and not part of the project scope. For

proof of concept, 2–3 concrete streams of data (eg: `Tool.P8-01.Log.Battery-Voltage`) will be selected. In terms of expected volume, 1 year worth of data will be pulled and downloaded into CSV format.

4.3 Pre-processing

Since this is a time series model, there are some key pre-processing that I plan to hit to make sure the data is ready:

1. Timestamp normalizations – As the same tools might be tested at different sites with multiple time zones (this could avoid daylight shift errors)
2. Resampling to a uniform Δt (eg: 60 seconds – exact to be decided post EDA) – Quick look a the streams and we saw the data sampling timing were irregular.
3. Detrending/seasonal decomposition – Some data will show strong cyclical patterns (battery voltage). This will significantly help isolate residuals
4. Unit harmonization (psi to bar, etc.) – Different types of sensors will produce different types of unit readings.

4.4 Model Architecture

4.4.1 Isolation Forest (Classical ML)

The project will focus mainly on classical machine learning methods for anomaly detection and the algorithm of choice would be Isolation Forest because this is an unsupervised learning project. The training will be split with model fitting with 80% of the older data and validate with 20% of the newer data. Then, the model will be able to flag anomaly points where the anomaly score $> 95\%$.

For comparison, a one-class SVM algorithm will be tested as fallback. Whichever model hits > 0.80 recall and > 0.70 precision on a hand-labelled test will be selected.

4.4.2 LSTM with Auto-Encoder

LSTM serves as an alternative to work on unstructured time series data. This will be a stretch model if time permits. The key steps:

1. Choose a clean baseline window (14 days of anomaly free data)
2. Train the LSTM auto-encoder only on that window
3. Score the full timeline by feeding all the new timeline and computing the reconstructed MSE
4. Declare it anomalous when MSE $>$ T, where T is MSE at 99th percentile

To justify the use of LSTM, I would compare PR-AUC to IF and adopt it only if a 5% gain over the baseline model and inference < 200 ms.

4.5 Web Development

The final piece of this project would be developing a web app that would be client facing with a landing page. Model will be packaged using FastAPI microservice, containerized with Docker so that ASP.NET remains untouched. We will run the container in the same Windows host via Docker Desktop. Key here is we will expose a single endpoint using POST method that accepts the user input in JSON format and returns the model output.

5 Expected Outcomes/Deliverables

The project is expected to fine-tune a time series model to detect anomalous well completion tools and automate test results documentation. This can be further broken down to smaller deliverables as below:

1. Anomaly Detection Using Time-Series Models
2. Develop equipment performance metrics
3. Automated Documentation Pipeline

6 Project Plan

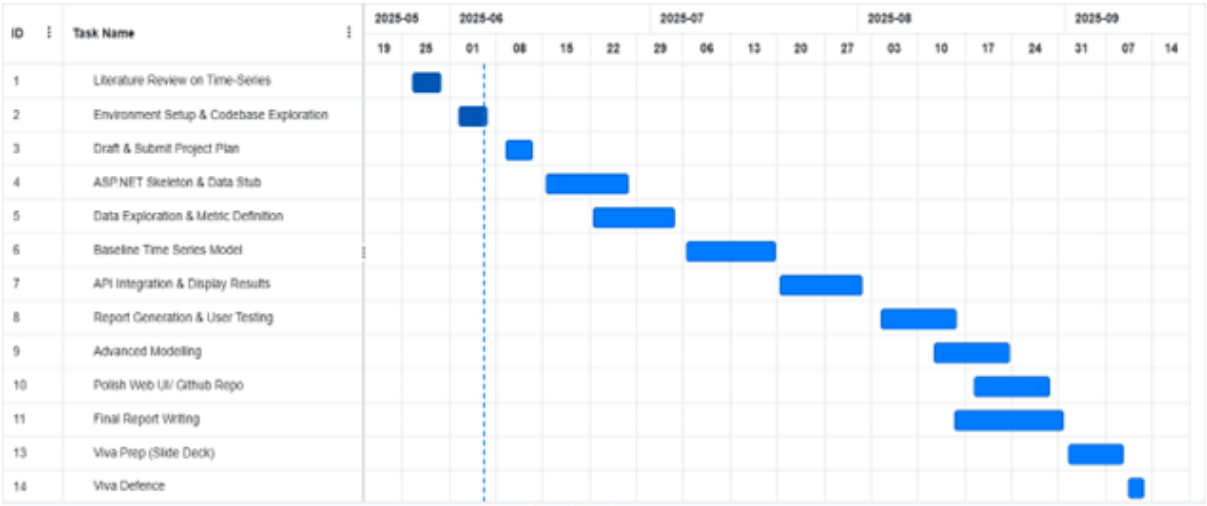

Figure 5: Gantt Chart showing the rough schedule for project plan at each stage.

7 References

1. Montelongo, M., Williamson, A., Florez, J., & Knight, J. (2025, March). Detecting low-quality intervals in surface logging data using AI-based anomaly detection (SPE-225579-MS). In *Proceedings of the SPE/IADC Middle East Drilling Technology Conference and Exhibition*, Manama, Bahrain. Society of Petroleum Engineers. https://doi.org/10.2118/225579-MS

2. Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation forest. In *Proceedings of the IEEE International Conference on Data Mining (ICDM '08)* (pp. 413–422). IEEE.

3. Malhotra, P., Vig, L., Shroff, G., & Agarwal, P. (2016). LSTM-based encoder-decoder for multi-sensor anomaly detection. In *Proceedings of the 2016 ICML Workshop on Anomaly Detection*. arXiv:1607.00148.

4. Ruff, L., Vandermeulen, R. A., Görnitz, N., Deecke, L., Siddiqui, S. A., Binder, A., Müller, E., & Kloft, M. (2018). Deep one-class classification. In *Proceedings of the 35th International Conference on Machine Learning (ICML '18)* (pp. 4393–4402). PMLR.

5. Reddicharla, N., Ali, M. A. S., Alshehhi, S. S., Elmansour, A., & Vanam, P. R. (2023, March). ESP failure prediction in water supply wells using unsupervised learning. In *Proceedings of the Gas & Oil Technology Showcase and Conference*, Dubai, UAE. Society of Petroleum Engineers. https://doi.org/10.2118/214010-MS

6. Wardana, F. K., Saputra, A. B., & Santoso, A. A. N. (2025, May). Leads: A deep learning approach to revolutionizing gas plant maintenance with advanced anomaly detection technology. In *Proceedings of the SPE Conference at Oman Petroleum & Energy Show*, Muscat, Oman. Society of Petroleum Engineers.

7. Jiang, L., Xu, H., Liu, J., Shen, X., Lu, S., & Shi, Z. (2022). Anomaly detection of industrial multi-sensor signals based on enhanced spatiotemporal features. *Neural Computing and Applications, 34*(11), 8465–8477. https://doi.org/10.1007/s00521-022-07101-y