SSC24-WV-04

From Ground to Orbit: Enhancing Satellite Autonomy with AI-Powered Anomaly Detection

James Murphy, Fangyijie Wang, Muhammad Deedahwar Mazhar Qureshi
Réaltra Space Systems Engineering,
The Realtime Building, Clonshaugh Business Park, Dublin, Ireland; +353 1 6875 723
Jmurphy@realtra.space

Brian Mac Namee
School of Computer Science, University College Dublin
University College Dublin, Belfield, Dublin, Ireland; +353 1 716 2315
Brian.MacNamee@ucd.ie

ABSTRACT

Recent advances in space technology have prompted a surge in the deployment of small satellite constellations by companies such as Starlink and OneWeb. Now numbering in the thousands, these constellations have significantly increased the burden on satellite operators who must monitor and manage them to ensure their safe and reliable functioning. In response to this heightened operational demand, autonomous small satellite operations have become a focal point for investment, innovation and exploration. The latest hardware breakthroughs in Edge AI have paved the way for applying artificial intelligence (AI) techniques directly on board these small satellite systems, heralding a new era of AI-empowered autonomous satellites. Space agencies such as ESA and NASA have taken notice of these advancements and are actively pursuing the development of on-board AI systems. This paper investigates the application of AI to monitor satellite telemetry data on board small satellites to enable real-time detection and immediate response to anomalies. We have curated a unique dataset derived from EIRSAT-1, Ireland's first domestically produced satellite, as a testing and validation resource for these ML models and the future development of AI-enabled small satellites. This dataset consists of a training set developed during ground testing and containing artificial anomalies induced to train satellite operators, a validation dataset containing real anomalies encountered during the qualification campaign, and an early flight test dataset collected since the satellite was launched on December 1st, 2023. This paper presents an in-depth analysis of the efficacy of several ML techniques when applied to the EIRSAT-1 dataset using flight-ready hardware. This study not only showcases the capabilities of these ML techniques in an operational environment but also sets the stage for future research and development in autonomous satellite systems.

INTRODUCTION

Satellite systems are critical in modern technology, facilitating communication, navigation, and observation. Ensuring their reliability and efficiency is paramount, given their operational complexity and the high cost associated with mission failures. One key aspect of maintaining satellite health is detecting and managing anomalies such as hardware malfunctions, software errors, and environmental impacts.

EIRSAT-1

EIRSAT-1 (Educational Irish Research Satellite-1) [1], Ireland's first satellite (see Figure 1), is an excellent focus for machine learning-based anomaly detection systems research. EIRSAT-1 was developed and launched by students and researchers at University College Dublin (UCD). This project was part of the European Space Agency's (ESA) "Fly Your Satellite!" program [1]. EIRSAT-1 development began in 2017 and it launched onboard a SpaceX Falcon 9 on December 1st, 2023. The primary objective of EIRSAT-1 is to serve as an educational tool, providing students and researchers with direct experience in satellite design, construction, testing, and operation.

EIRSAT-1 is a 2U CubeSat equipped with three payloads designed for scientific and technological experiments in space [1]: the Gamma-ray Module (GMOD) instrument is designed to detect and analyze gamma-ray bursts (GRBs); the ENBIO Module (EMOD) experiment tests advanced thermal control coatings developed by the Irish company ENBIO; and the Wave-Based Experiment tests novel technologies for space communications.

EIRSAT-1 incorporates telemetry, command, and data handling systems, enabling communication with ground stations and transmission of scientific data back to Earth [1]. This allows ground analysis of data the satellite system provides from both nominal spacecraft operations and the onboard payloads.

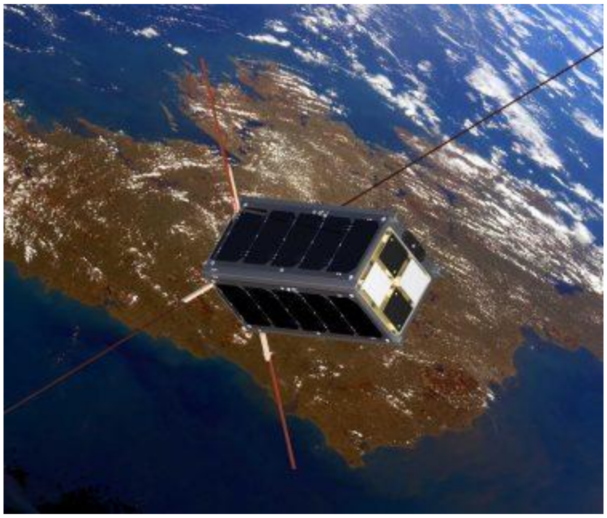

Figure 1: EIRSAT-1.

Deployment Scenario

Multiple datasets are generated during the development and qualification process for a satellite. Firstly, a dataset is designed to train satellite operators on system outputs to enable them to recognize when issues arise on board [2]. In the case of EIRSAT-1, this is called the Flight-Test dataset. This dataset is designed to contain artificial anomalies deliberately introduced to simulate various fault conditions that satellite operators might encounter. The Flight-Test dataset is essential in the initial phase of developing an anomaly detection model. A machine learning model can learn to identify patterns and signatures associated with anomalies (or nominal behaviour) using this dataset. The controlled environment from which the Flight-Test dataset is generated allows for model parameters to be systematically evaluated and tuned, ensuring that the model can recognize a wide range of potential issues.

This paper introduces the EIRSAT-1 anomaly detection dataset. This paper aims to publish a cleaned and "training ready" version of the datasets gathered from the EIRSAT-1 development, qualification and flight campaigns to enable others to continue the work in the future. This dataset is generated from the test campaign and initial flight phase of EIRSAT-1. This dataset is split into 3 different phases: the Flight-Test dataset, the TVAC (Thermal Vacuum Test) dataset, and the Flight dataset. The Flight-Test dataset was generated using artificially induced anomalies to test how the satellite reacts to potential issues; it is also used to train satellite operators in handling any potential anomalies the satellite suffers. The TVAC dataset contains real anomalies that occurred during the Thermal Vacuum test. The EIRSAT-1 dataset also contains an initial Flight dataset generated in the first several months of operation. As well as introducing the dataset, his paper also explores a development process where a model can be trained on the Flight-Test dataset, validated on the TVAC dataset, and deployed on the unlabeled Flight dataset, utilizing the metrics from the previous two datasets to assume performance metrics on the unlabeled flight dataset.

Category	Channels (Primary, Failsafe)	Frequency	No of Fields (Primary, Failsafe)
Housekeeping	3, 15	50 sec	94, 68
ADCS	5, 17	300 sec	34, 29
TED	4, 16	600 sec	77, 46
Power	7, 19	60 sec	91, 72

Table 1: The categories of data contained in the EIRSAT-1 datasets with the mapping of categories to channels from primary and failsafe data lines

DATASETS

This section describes the different parts of the EIRSAT-1 anomaly detection dataset, and how they can be used for building anomaly detection models. The data in the three different parts of the dataset described in the previous section (Flight-Test, TVAC, and Flight) is derived from two sources: the primary software data line on board and the failsafe data line. The primary software data line is the main data output of the satellite and contains all information intended to be transmitted to the ground. The failsafe data line serves as a backup with reduced functionality, such as no access to payloads, and is activated in case of multiple power cycles. Both data lines utilize aggregators to gather data from the spacecraft and store it in downlinked channels. The channels are then classified into four major categories (as shown in Table 1): Housekeeping, ADCS (Attitude Determination and Control System), TED (Telemetry-enhanced Statistics), and Power based on the fields they include. These fields contain readings and statistics from different components of the spacecraft, which are described in more detail in the remainder of this section.

Each category consists of one primary channel and one failsafe channel, which is only activated once the primary channel is inactive. This means that only one channel from any category is active at once. Due to resource constraints on the backup system, the failsafe channels do not carry the payload entries which are included on the primary channel, resulting in a smaller feature set. Additionally, each category has a different frequency for logging entries into the system, which

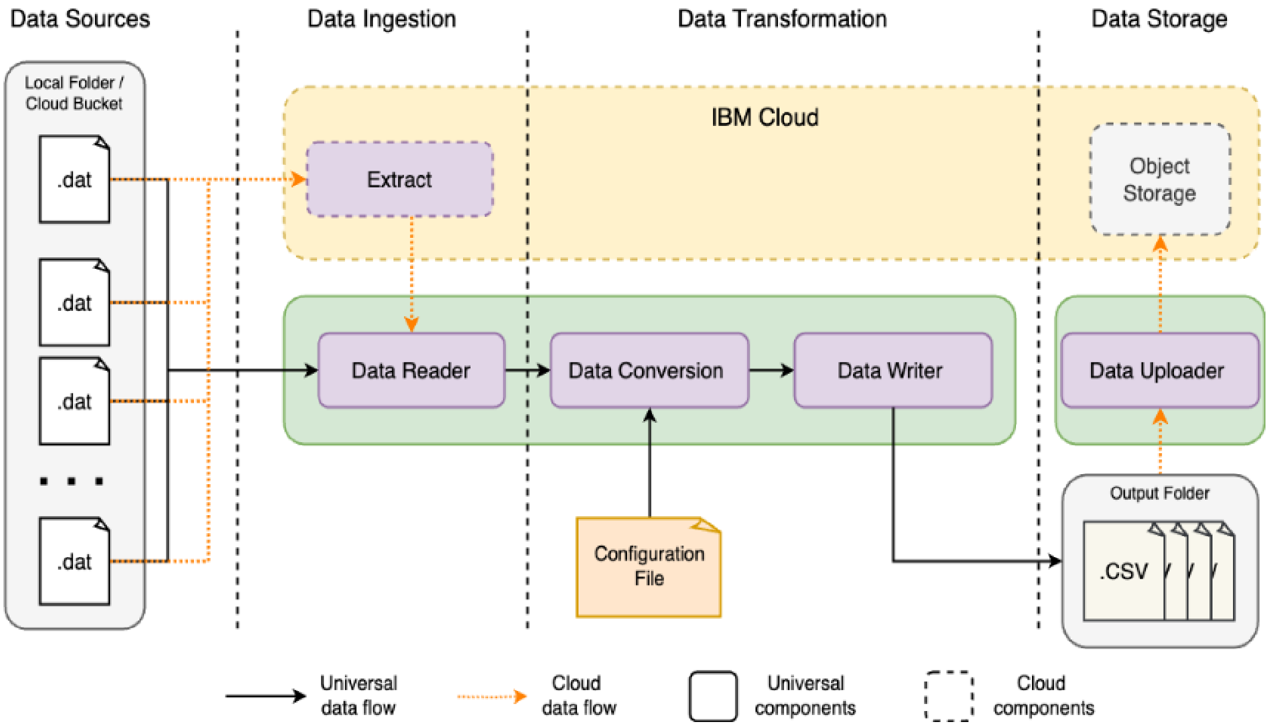

Figure 2: Overview of our data pipeline and data flows.

means that not all categories output the same volume of data in any given time duration.

Structure of the Dataset

The complete dataset for EIRSAT-1 can be found here: https://zenodo.org/records/11551556. The dataset is distributed in three parts:

Thermal Vacuum (TVAC) Test Dataset: The TVAC dataset is collected from the initial qualification test campaign. It contains data from both the primary and failsafe data lines, so all eight channels have data in them. The campaign suffered some real anomalies, which will be discussed later.

Flight Test Dataset: The Flight Test dataset was collected from the second test campaign when EIRSAT was used in a flight simulation. During this test, the simulation was injected with some artificial anomalies, but it also suffered some real anomalies, which will also be discussed later. It also contains data from both the primary and failsafe data lines.

Flight Dataset: The Flight Dataset is the dataset collected from the EIRSAT flight commissioning phase, which lasted four months from December 2023 to March 2024. EIRSAT did not go into a failsafe mode during the commissioning, so these channels had no data. Thus, the data in this dataset contains only the four primary channels for each month. The data was downlinked for each month, thus resulting in four sub-datasets for December, January, February, and March. Since this is an actual flight dataset, there are no annotated anomalies in this dataset.

Channel Definitions

The spacecraft has eight channels (four from the primary data line and four from the failsafe data line), which are grouped into four categories. Each category covers specific properties of the spacecraft and includes the channels that report those properties. Due to the different frequencies of each channel, the distribution of data across the channels is not uniform, with specific periods of time having more data than others. For each category, the failsafe data line includes a subset of the parameters reported by the primary data line. For example, in the 'Housekeeping' category, Channel 15 comes from the primary data line, and Channel 3 comes from the failsafe

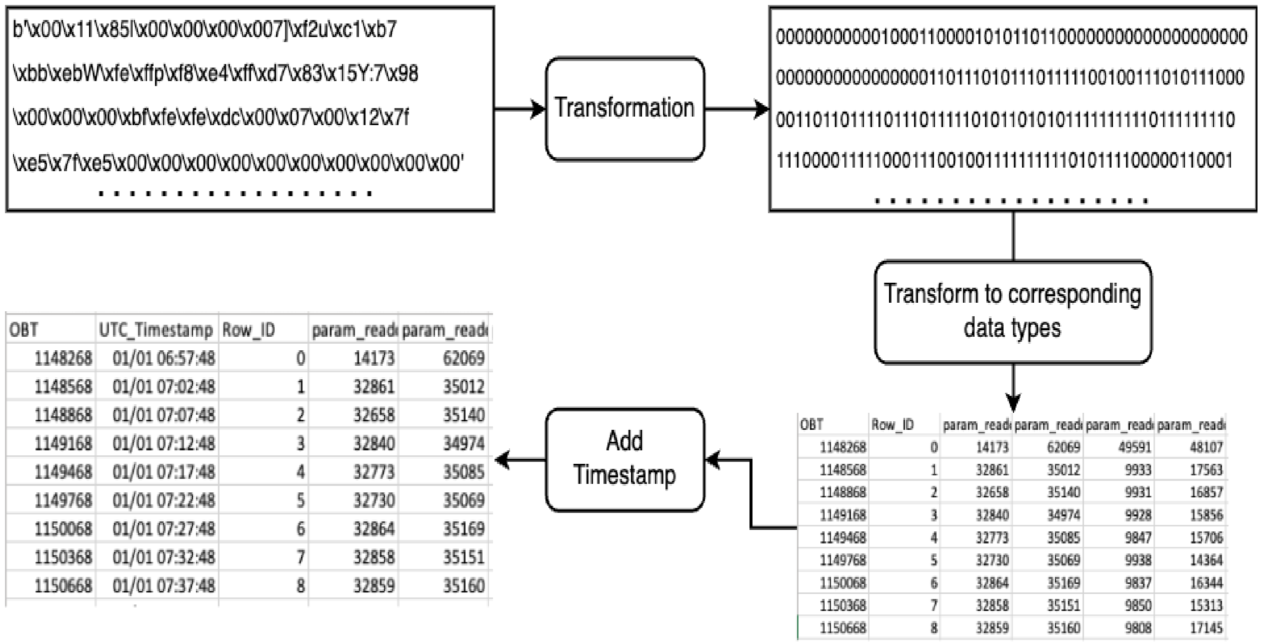

Figure 3: An example of Channel05 raw data and saved CSV data.

data line (as shown in Table 1). Each category contains the following information:

- *Housekeeping*: This category contains information about the spacecraft's general health, including parameters for temperatures, health, and the status of different components. It also contains many features related to the anomalies that occurred during the test campaigns.
- *Power:* This category contains information about electrical power generation, storage, and distribution. It includes parameters related to the voltages and currents across different components of the spacecraft and many features related to the anomalies suffered during the test campaigns.
- *Telemetry-enhanced Statistics (TED):* TED mainly contains error counters and other spacecraft statistics/readings for different components, which can be useful for detecting changes in the state of the subsystems reflected by changing error counts.
- *Attitude Determination and Control Subsystems (ADCS):* This category mainly contains housekeeping statistics from the attitude determination and control subsystems. It includes the smallest number of parameters of any category and is not useful for detecting the anomalies reported in the TVAC and Flight-Test campaigns.

Each row in the data begins with a timestamp or 'On-Board Time (OBT)' value, which serves as an identifier for each individual data point across all datasets [2].

Preprocessing Data Pipeline

Before training any ML models using these datasets, an automatic data pipeline[1] was constructed to preprocess the raw EIRSAT-1 data. The data is saved in a CSV file for better readability after preprocessing. The pipeline offers two execution modes: local and cloud, enabling execution on either a local machine or the IBM cloud. Figure 2 provides an overview of the data pipeline. When given a target file name, the pipeline retrieves the corresponding data file and reads the bytes directly into memory. In cloud mode, the pipeline will first download the raw data files from IBM Cloud Object Storage using the given file name to find the file path within the cloud environment. From end to end, the pipeline transforms the EIRSAT-1 raw data into a data table according to the metadata definition.

[1] The pipeline was constructed using Python 3.9, ibm_boto3 [3] and the Pandas [4] library.

Dataset	Anomaly	Instances	Samples
TVAC	Anomaly 1	2	496
TVAC	Anomaly 2	4	3039
Flight Test	Anomaly 1	1	30
Flight Test	Anomaly 2	2	16055
Flight Test	Anomaly 3	1	92
Flight Test	Anomaly 4	1	1096
Flight Test	Anomaly 5	1	164

Table 2: The data distribution of each anomaly in the dataset.

Figure 3 illustrates an example of converting raw data into a usable CSV data table. Initially, the raw data is extracted from the Channel05 file as bytes. Following this, the pipeline transforms the bytes into an 8-bit data representation. Subsequently, the 8-bit data is organized into a data frame and converted into appropriate data types such as integer, float, string, and bit as defined by a meta-data file. Each row in the data begins with the On-Board Time (OBT) value, which serves as an identifier for each individual data point across all datasets [2]. We augment the data frame with a timestamp column to facilitate downstream time series analysis. This is achieved by using the OBT values as Unix time and integrating them as Universal Time Coordinated (UTC) in the data frame.

Anomalies

Anomaly identification is an essential aspect of these datasets. For both the Thermal Vacuum (TVAC) Test Campaign and the Flight Test Campaign, information regarding known anomalies is recorded in the form of timestamp ranges provided by the scientists who carried out the test. Broadly, each annotated anomaly is categorized as an injected anomaly (triggered due to a forced change in conditions) or a real anomaly (occurring naturally in the system). All the variables mentioned with anomalies here are parameters in the dataset. The Excel file in the dataset folder shows the category to which these parameters belong. Some anomalies occurred more than once and for a varying duration of time, as shown in Table 2. The instances with the anomaly in the dataset have been labelled as '1' for their respective column.

TVAC Dataset

During the TVAC [6] test campaign, two anomalies were encountered.

1. *Anomaly 1:* During vibration testing, a damaged solder joint on the battery board and changing pressure in TVAC stopped the communication between the battery and the On-Board Computer (OBC) from working. The noticeable change in the data for this anomaly is that the Battery Voltage parameter (platform.BAT.batteryVoltage) drops to 0.
2. *Anomaly 2:* This is a communication issue on the I2C Line between the onboard computer and the communications subsystem. The primary indicator for this anomaly is that the Temperature value shown in the parameter (platform.CMC.temperaturePA) abruptly goes to 0. It is important to note that this value can naturally go to 0 if the sensor is facing away from the sun in colder conditions. Therefore, it is also essential to notice the other Common Mode Current (CMC) parameters in the dataset.

The TVAC anomalies are real anomalies that occurred during the test campaign [6].

Flight-Test Dataset: The following anomalies arose and were annotated during the flight-test campaign.

1. *Anomaly 1:* This is an anomaly with the Electrical Power Subsystem (EPS) Watchdog module, which manages power management and distribution in the satellite. It is related to all the platform.EPS parameters in the dataset. This anomaly occurs when all these parameters go to 0 for four minutes before the spacecraft power cycles (which may initiate failsafe).
2. *Anomaly 2:* This is an injected Low Battery Anomaly. The battery voltage was dropped below the configurable voltage of 7.5V. Regarding the dataset, this value approximates 830 units in the associated parameter in the dataset (platform.EPS.busVoltages[0]).
3. *Anomaly 3:* This is an OBC Reset Sequence triggered when the system enters failsafe mode. The key indicator for this anomaly is the OBT.Uptime parameter, which would reset to 0.
4. *Anomaly 4:* This is a more difficult gyro-error anomaly to identify. Noticing discrepancies in the values requires examining multiple

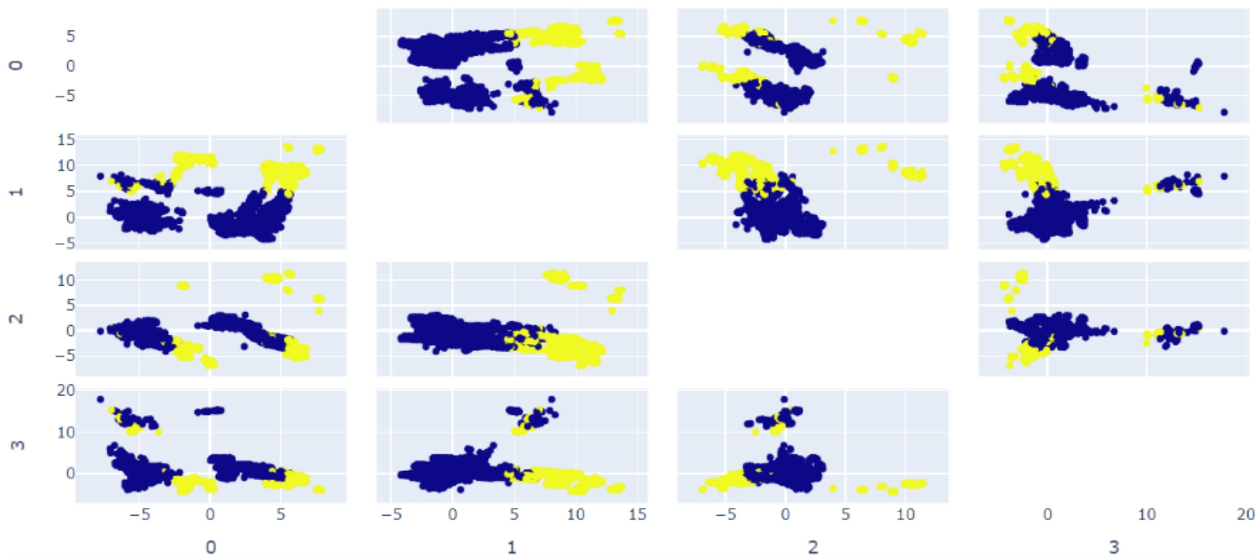

Figure 4: The PCA distribution obtained from the TVAC Dataset on Channel 15.

platform ADCS parameters in the dataset (e.g., ADCS.rawGyroRate).

5. *Anomaly 5:* This is an injected replication of Anomaly 2 in the TVAC test. An injected I2C CMC anomaly (an anomaly in the transceiver that interfaces between the OBC and the I2C) causes an OBC reboot. The indicator for the anomaly is the decrease in the same parameter as Anomaly 2 in TVAC (platform.CMC.temperaturePa) to 0. However, like Anomaly 2 in the TVAC test, this value can naturally go to 0 as well, so other CMC transceiver parameters must also be noticed.

Training a Supervised Machine Learning Model on the Dataset

We trained a supervised machine-learning model on all the channels belonging to the Primary Data line of each category. It is important to note that since the anomalies marked in both datasets are mutually exclusive except for Anomaly 2 (TVAC) and Anomaly 5 (Flight Test), the supervised ML models are not expected to replicate performance across both datasets.

First, we combine all the anomalies in our dataset into a single feature (Anomaly_Bin) that represents the binary existence of an anomaly in the dataset. Thus, we treat this as a binary classification problem targeted at the existence of anomalies. We isolate individual channels and drop the columns with missing values to obtain data from a particular channel (primary/failsafe data line of a particular category).

In each dataset, numerous features do not provide much information since their values do not vary much throughout the dataset. These features would not be helpful in model training and can affect the model's performance. Thus, we reduce the number of features using Principal Component Analysis (PCA) [7] to retain 95% variance across the data. This leads to a significant decrease in the number of features for training. An example of cluster distribution of the top 5 features is shown in Figures 4 and 5. Here, the X and Y axis refer to the top five principal components obtained from PCA. We map these components against each other to show the distribution between the anomalous and non-anomalous data.

The features obtained from PCA are passed to a Random Forest Classifier [8]. We use 5-fold cross-validation [10] to validate the model performance on the training set. The model performance for each channel on both datasets is shown in Table 3. The support column indicates the number of samples in the test set for the anomalous class.

The model performance in Table 3 unveils some interesting insights. The model trained on Channel 15 (the primary data line of the Housekeeping category) shows the best anomaly detection performance. This is consistent with the fact that it carries many important parameters (features) with respect to the anomalies found in the dataset. On the contrary, Channel 17 (the

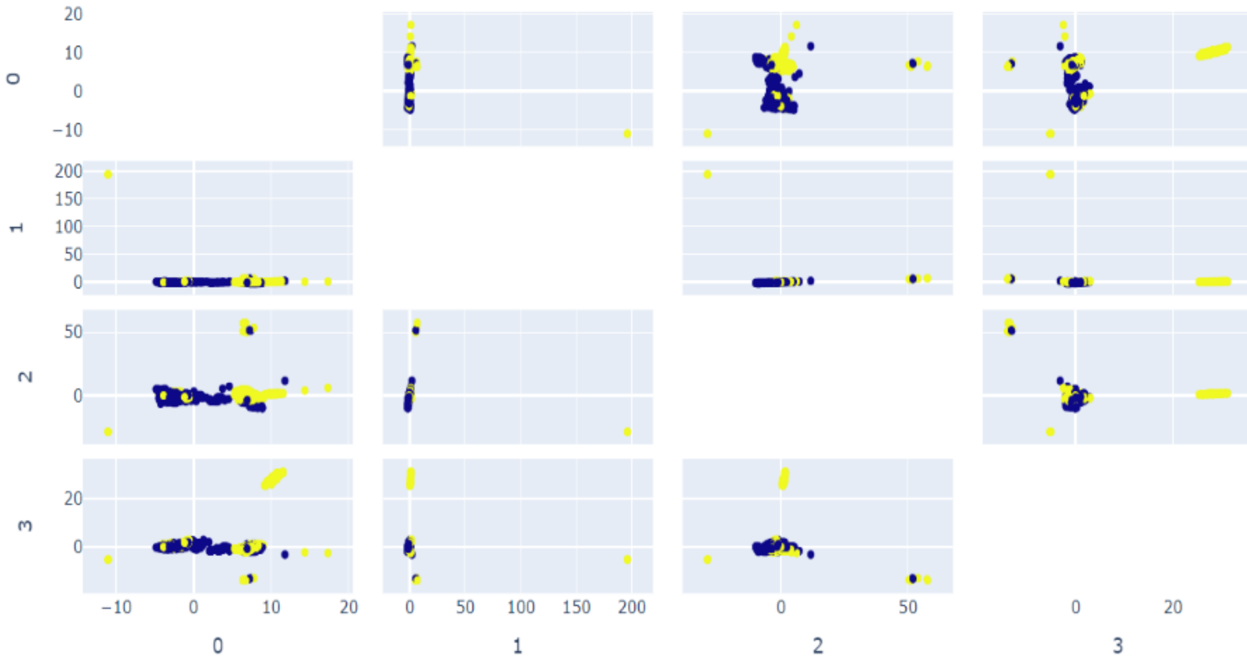

Figure 5: The PCA distribution obtained from the Flight Test Dataset.

primary data line of the ADCS Category) carries fewer features that are mostly irrelevant to most labelled anomalies. This justifies the model's poor performance on this channel.

Due to the difference in the number of parameters (features) reported by the primary and failsafe data lines, channels from the two data lines corresponding to the same category (e.g. Channels 15 and 3 correspond to the primary and failsafe data lines for the Housekeeping category) have mismatched feature set. To use both channels together, we would have to employ a strategy to fill in the missing (NaN) values for parameters that are included in the primary (e.g. Channel 15) but not in the corresponding failsafe (e.g. Channel 3) data line. Such a strategy can also allow us to use all the data in a particular category. A further extension could be to fill in all the missing values in the complete dataset. Since the merged dataset includes data from all channels with missing values for the parameters not reported by a particular channel, accurately filling these values would remove the distinction between channels and enable the complete dataset to be used. This is a possible future direction for this dataset.

Another challenge here is that while anomalies were encountered in both campaigns, they were not the same anomalies thus affecting generalizability across the two test campaign datasets that we use for model training. We validated the Flight-Test dataset on the TVAC dataset for Channel 15 to capture the transferability. Although the anomalies suffered across both datasets are different, a high F-score (0.97) on the non-anomalous data shows the ability of the model to generalize on non-anomalous data well. For the anomalous data, the model obtained an F-Score of 0.56.

Dataset	Channel	Precision	Recall	F1-Score	Support
TVAC	**15**	**1.00**	**0.95**	**0.98**	**482**
	16	1.00	0.82	0.90	17
	17	0.88	0.21	0.33	34
	19	1.00	0.74	0.85	237
Flight Test	**15**	**0.99**	**0.87**	**0.93**	**2089**
	16	0.85	0.87	0.86	174
	17	0.03	0.03	0.03	347
	19	0.91	0.83	0.87	1725

Table 3: Per Channel performance on binary classification of anomalous data in the TVAC and Flight-Test datasets.

Dataset	F1-Score
TVAC	0.732
Flight-Test	0.717

Table 4: Results for the autoencoder on the TVAC and Flight Test Datasets

Training a Semi-Ssupervised Machine Learning Model

Continuing from the supervised model, we also trained an unsupervised machine learning model to compare performance. An unsupervised model carries the advantage of not being dependent on labels, which removes the limitations of model generalizability due to differing anomalies in the TVAC and Flight Test datasets. The model is trained on the Primary data line for the Housekeeping category (Channel-15). We did not reduce the number of features for this task, resulting in an 85-feature input. Here, our aim is to fit the model on non-anomalous data so that it reconstructs non-anomalous data well but gives a greater reconstruction error when reconstructing anomalous data. The model is compiled using the Adaptive Motion Estimator (Adam) optimizer, and the reconstruction loss is calculated from the mean-squared error (MSE).

The model is a dense autoencoder with a one-layer Encoder and Decoder. The Encoder is a dense layer of 20 neurons activated by ReLU with a dropout value of 0.5. This is followed by the decoder layer, which has a size of 85 neurons (the same as the input) and is activated by the sigmoid function. The architectural specifications of the model are determined by fine-tuning through Keras Tuner. Upon training, the model achieves a ROC score of 0.78.

The model is evaluated on both the Flight-Test and the TVAC Dataset. The resultant performance of the model across the datasets is shown in Table 4. Here, the F1-Score on TVAC and the Flight-Test dataset are similar, which shows that the model can generalize well on the non-anomalous data across both datasets.

Deployment on Flight Board

We also deployed the autoencoder model on the commercial version of the Ubotica UB0100, the Intel Myriad-X. This board is currently flying on the European Space Agency's Phi-Sat-1 mission for Cloud Segmentation [11] and is due to fly on the Phi-Sat-2.

Channel	Dec	Jan	Feb	Mar
15	23376	38663	27368	21614
16	812	2897	556	571
17	1163	2115	328	2165
19	3084	14823	4407	5659
Total	**28435**	**58498**	**32659**	**30009**

Table 5: Data for each month in the Flight dataset.

The board evaluates 585 data frames per second using this model. A data frame in this context is one feature window being passed through the model. For comparison, the same model, when executed on an 11th-generation Intel Core i7 CPU, evaluates 10,897 frames per second. This marked difference is expected given the resource constraints on the flight board but shows promise with the capabilities of model deployment on a board that is currently in flight on a CubeSat.

Details of the Flight Dataset

The Flight dataset consists of real-time data obtained from the EIRSAT-1 flight mission from December to March. During the entire flight duration, the satellite did not go into failsafe mode, so the failsafe data line does not contain any data. All data from the flight is recorded in the four Primary channels for each category (15, 16, 17, 19).

The flight data has no annotations for anomalies that would have been encountered during flight; hence, it is an unlabeled dataset. Due to the downlink constraints in communications, there are also gaps in the data. This means that data from one part of the month might have been downloaded in another month. The timestamp (On-Board Time) parameter 'OBT' can be consulted to create a coherent time series. For example, in the Channel 15 CSV file for December, the OBT value jumps from 105093 to 261456. The data for the missing time stamps was later downlinked in January. Thus, the missing time stamps can be found at the beginning of the Channel 15 CSV file for January. The number of timestamps downlinked in each month is seen in Table 5.

This dataset can provide a good exercise for users and the community in general for trying out anomaly detection on an actual flight dataset without any labels. The table thus warrants an unsupervised approach with a greater focus on non-anomalous data [9].

FUTURE WORK

The purpose of this paper is twofold. Firstly, we introduce a new complete dataset to the community with two accompanying labelled test campaign datasets that can be used to validate the performance of anomaly detection models. Secondly, we use a PCA-based Random Forest Classifier to validate the annotations present in the testing datasets. In the future, we will build on this work and attempt to label the in-flight dataset in multiple phases thoroughly. We have also released this dataset publicly to allow others in the community to work on it. Our current work naturally builds up to this goal by enabling us to use the trained classifiers to generate soft labels, which would be helpful given the model's generalizability (as evaluated across the TVAC and Flight-Test datasets).

Furthermore, we will attempt to bypass the issues of annotation quality that arise in such datasets by adopting an unsupervised or semi-supervised ML approach. We will focus on the non-anomalous samples and the multivariate trends that can make this identification easier. An alternative approach is to adopt a recurrent neural network or an LSTM that is more capable of dealing with the temporal aspects of the data.

Finally, the missing values when the data is mapped temporally over the timestamp (OBT), especially in the case of channels from the failsafe data line, is an interesting problem which would require some interpolation or an alternative method to enable the use of maximum data for the ML model. When applied to all data samples in the merged form of the data (when mapped by OBT), such interpolations could further reduce the data's complexity by minimising the differences across different channels.

CONCLUSION:

In this paper, we have introduced three new datasets: two from test campaigns and one from in-flight data under real conditions. Using these datasets we have trained and evaluated a number of machine learning models, demonstrating the potential of the EIRSAT-1 datset for further research and application. Our work represents an initial stage the community can build upon, with plans to label the in-flight dataset further and refine the machine learning model for enhanced performance.

Additionally, we have presented a transformation methodology specifically designed to prepare satellite telemetry data for time series analysis. To support this, we have developed a data pipeline that facilitates the transformation process, offering flexibility through two execution modes: local and cloud-based. Our dataset is available on Zenodo, and we are inviting collaboration and further advancements in this field. This work lays a foundational framework for future developments and improvements in satellite telemetry data analysis.

References

1. Murphy, D., Flanagan, J., Thompson, J., Doyle, M., Erkal, J., Gloster, A., ... & Walsh, S. (2018, April). EIRSAT-1-the educational Irish research satellite. In Proceedings of the 2nd Symposium on Space Educational Activities (Vol. 8859). Budapest, Hungary: Budapest University of Technology and Economics.
2. Walsh, S., Murphy, D., Doyle, M., Reilly, J., Thompson, J., Dunwoody, R., ... & McBreen, S. (2021). Development of the EIRSAT-1 CubeSat through functional verification of the engineering qualification model. Aerospace, 8(9), 254.
3. IBM. (2023, June). Working on data science projects. Retrieved April 15, 2024, from https://opendatahub.io/docs/working-on-data-science-projects/#_using_data_science_projects
4. Collins, E. R. (2024, February 23). 5 Best Ways to Convert Python Bytes to Binary String. Retrieved May 7, 2024, from https://blog.finxter.com/5-best-ways-to-convert-python-bytes-to-binary-string/
5. IBM. (2024, February). S3. Retrieved May 13, 2024, from https://ibm.github.io/ibm-cos-sdk-python/reference/services/s3.html#multipartupload
6. Dunwoody, R., Reilly, J., Murphy, D., Doyle, M., Thompson, J., Finneran, G., ... & McBreen, S. (2022). Thermal vacuum test campaign of the EIRSAT-1 engineering qualification model. Aerospace, 9(2), 99.
7. Maćkiewicz, Andrzej, and Waldemar Ratajczak. "Principal components analysis (PCA)." *Computers & Geosciences* 19.3 (1993): 303-342.
8. Breiman, Leo. "Random forests." *Machine learning* 45 (2001): 5-32.
9. Choi, Kukjin, et al. "Deep learning for anomaly detection in time-series data: Review, analysis, and guidelines." *IEEE access* 9 (2021): 120043-120065.
10. Schaffer, Cullen. "Selecting a classification method by cross-validation." Machine learning 13 (1993): 135-143.
11. Furano, G., Meoni, G., Dunne, A., Moloney, D., Ferlet-Cavrois, V., Tavoularis, A., ... & Fanucci, L. (2020). Towards the use of artificial intelligence on the edge in space systems:

Challenges and opportunities. IEEE Aerospace and Electronic Systems Magazine, 35(12), 44-56.