Multi-Layered Machine Learning for Rapid LEO Object Characterization leveraging Global Radar Data

Harry She
LeoLabs
Chandler Phelps, Owen Marshall, Erin Dale
LeoLabs

ABSTRACT

Determining the physical characteristics and maneuverability capabilities of a satellite, as well as its orbital similarity to other spacecraft, is essential for enhancing tracking accuracy, collision avoidance, and overall space situational awareness. Existing methods for identifying an object as a Maneuverable or Non-Maneuverable Payload, Rocket Body or Debris Fragment are labor-intensive, time-consuming, and were not designed for launches with large manifests. As of February 2024, there are several hundred unknown objects in the United States public catalog. With the number of active satellites anticipated to grow from 8,000 to 58,000 by 2030, it is becoming increasingly crucial to employ rapid, accurate, and automated methods of characterization to stay abreast of the forecasted launch cadence.

LeoLabs uses a set of highly performant XGBoost classifiers trained with data collected from its global radar network to determine object type in LEO (low Earth orbit) within 72 hours of association.

Our approach involves processing raw radar data from LeoLabs' ground-based sites for thousands of objects in the public catalog. We then extract features that accurately represent the physical characteristics and morphology of a satellite, including, but not limited to, radar cross section (RCS), the signal-to-noise ratio (SNR), measurement count, and other pertinent transformations and aggregations. We train an XGBoost model on historical data collected from all passes from LeoLabs' radar sites for a fixed period, on a per target basis. When fed with these extracted features of a given target, the model produces an object type – Payload, Debris, or Rocket Body.

Objects classified by this model as a Payload are further processed to determine if they are Maneuverable or Non-Maneuverable. This involves a processing pipeline that predicts when an object has deviated from natural decay during its historical timeline, in combination with other derived features that score the magnitude and prevalence of such maneuvers.

Model performance is then evaluated using a hold-out test dataset of 20% of the targets from the public catalog. These models achieved up to 95% accuracy against the reported values in the public catalog for certain object types, as well as hand-labelled maneuverability status by expert analysts. Furthermore, we utilize an unsupervised machine learning technique to map objects into a 2-dimensional space such that clusters that appear tend to correspond to physical characteristics and maneuverability status of the object. We have investigated using this technique to cluster newly launched objects with existing spacecraft in low Earth orbit. This algorithm is useful for determining similarity and uncovering structure in relevant high-dimensional extracted features that may be otherwise hard to dissect.

These methods can provide valuable assistance to expert human analysts as they can use such plots to better reason about the spacecraft morphology, behavior, and intent of specific categories of satellites they are knowledgeable about, allowing them to infer the characteristics of new objects by comparing the 2-dimensional location of that target to neighboring targets with known characteristics. The results of investigation have already proved fruitful for myriad practical applications, ranging from identifying if an unknown object is part of an existing family of space systems to inferring the role and mission of a given satellite. These insights enhance the accuracy of a broader, holistic object characterization process.

We propose that focusing development effort on such capabilities are fundamental in establishing a well characterized, predictable LEO space environment for the wider space community - both public and private. By extension, this also motivates data-driven space domain awareness to admonish adversarial action in space through enhanced attribution, as well as enabling progress towards formulating effective regulations that promote space sustainability and improve compliance in LEO.

1. INTRODUCTION

1.1 Background and Motivation

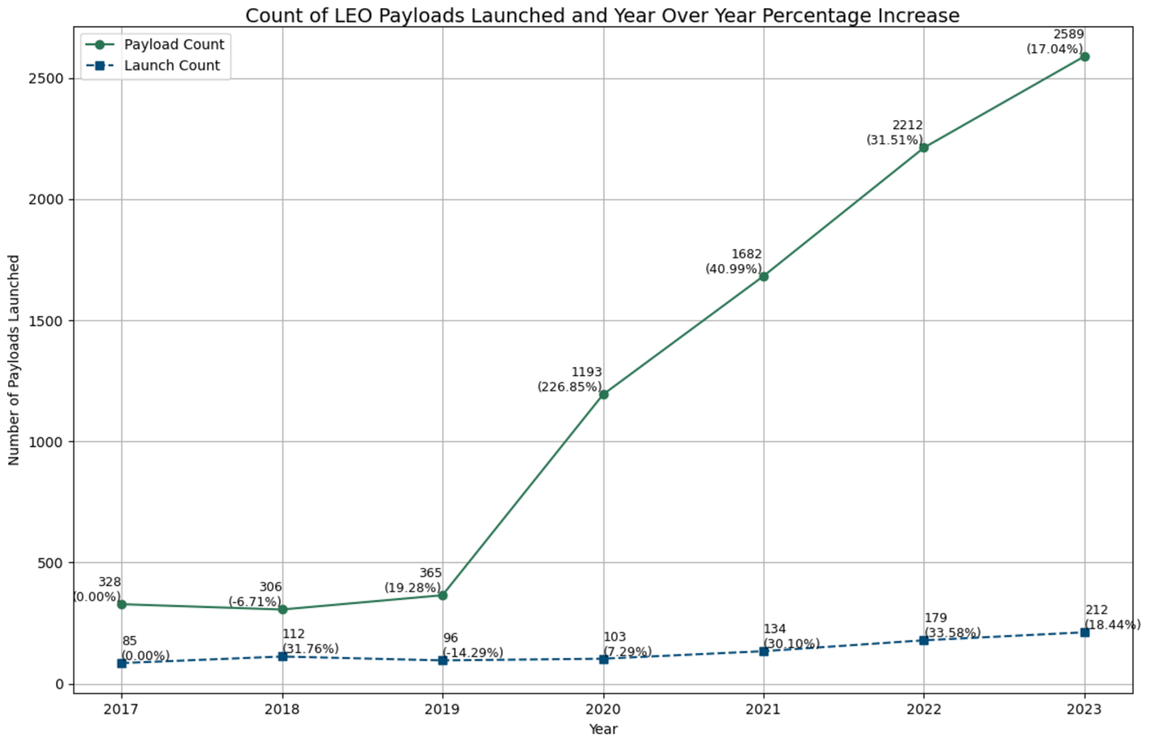

Fig. 1: Count of LEO Payloads launched from 2019 to 2023 [5]

Modern life heavily relies on key operational satellites for a range of essential services, including communications, global navigation, surveillance, remote sensing, and scientific research. The conveniences, safety, and communication capabilities provided by these satellite systems have become so integral that it is difficult to envision life without them.

However, the rapid growth in demand for these services has led to increasing congestion in Low Earth Orbit (LEO). The broadening democratization and commercialization of space, driven by a surge in satellite deployment, has also contributed to this congestion. Both well-established and merging launch companies have significantly accelerated the frequency of launches, with some rideshare missions now capable of deploying dozens to hundreds of satellites into LEO in a single launch.

This rapid increase in satellite deployment is reflected in the numbers: from 2019 to 2023, the number of payloads launched into LEO per year grew from less than 400 to over 2500 in the public satellite catalog (SATCAT) as shown in Figure 1 [5].

Furthermore, sovereign nations are racing to secure space supremacy by making substantial investments in satellite constellations and developing foundational infrastructure to assert dominance in LEO. While some countries have established national regulations and protocols for satellite registration and tracking, there are no consistent international standards governing the disclosure of payload details, especially those related to military or intelligence applications. This regulatory gap creates an environment where nations may withhold information to protect national security interests or strategic advantages [2].

As a result, we observe that the number of ‘unknown’ objects in the SATCAT cataloged by the 18th Space Defense Squadron (18 SDS) as part of the Space Surveillance Network, is increasing every year; from 2019 to 2023, unknowns rose from approximately 400 to over 1400 in 2023 [10].

With the number of launched payloads projected to increase to 58,000 by 2030, coupled with a poorly regulated space environment, it is becoming increasingly crucial to employ rapid, accurate, and automated methods of characterization to keep pace with the anticipated launch cadence [12].

1.2 Scope and Objectives

LeoLabs was founded with the mission to preserve the safety and viability of LEO through data transparency. As the industry-leading mapping platform for space, we recognize that an information deficit is what overwhelmingly allows unsafe or irresponsible actions in LEO to go unchecked. In recent years, our efforts have been focused on closing this deficit and providing data products that provide insight to analysts and operators.

In pursuit of this goal, we have developed a Multi-Layered Machine Learning system to rapidly characterize unknown objects in LEO shortly after launch. The objective of the tool is to best characterize what type of object it is (Maneuverable Payload, Non-Maneuverable Payload, Debris, or Rocket Body), and give an estimate of object size (through hard body radius); equipping analysts and operators with pertinent clustering information and insights to help determine an object's similarities with other known objects in the public catalog.

2. METHODOLOGY

2.1 Data Sources

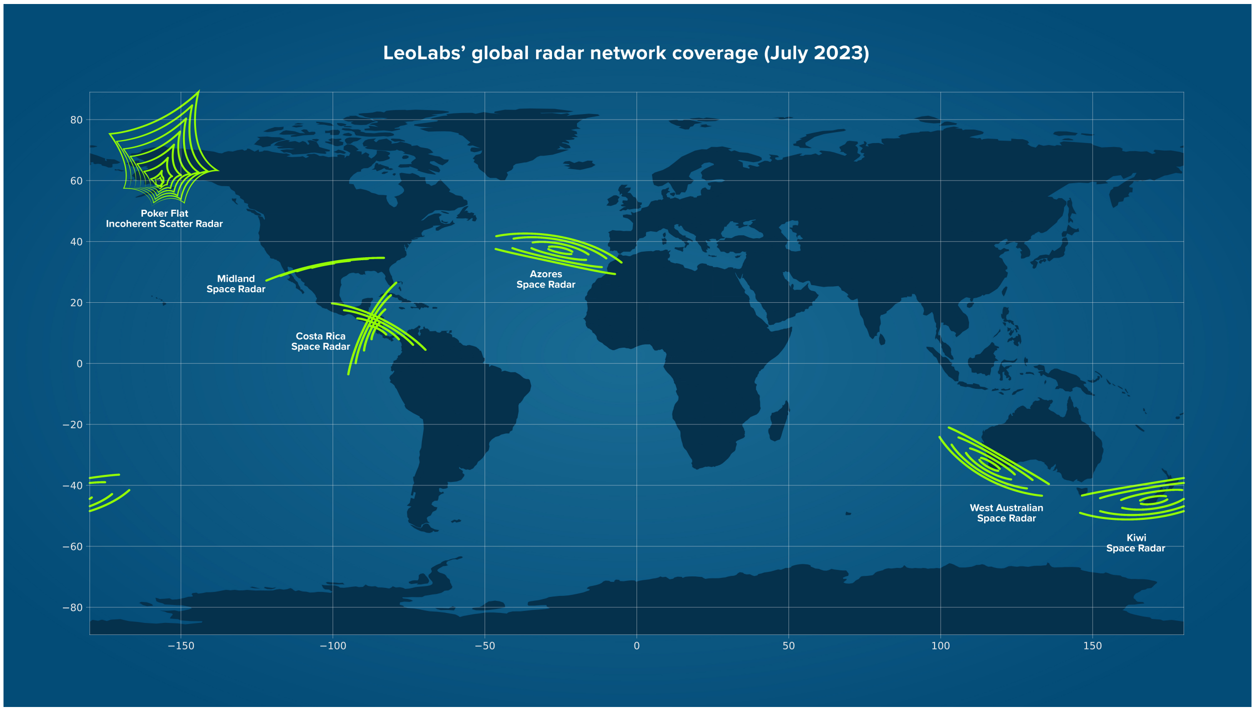

Fig. 2: LeoLabs' global radar network as of July 2023.

Currently, our Object Characterization models are primarily fed by LeoLabs' global network of radars. This network includes eight S-Band radars and two ultra high frequency (UHF) radars, which are deployed at various strategically located sites around the world, as illustrated in Figure 2 [8]:

Inputs:

- LeoLabs Measurement SNR

- LeoLabs Radar Cross Section (RCS)
- LeoLabs Two Line Element sets (TLEs) and LeoLabs States
- SpaceTrack TLEs

Outputs:

- SpaceTrack Object Type Labels
- LeoLabs maneuverable payload labels
- LeoLabs Hard Body Radius (HBR)

In addition to LeoLabs data, we use SpaceTrack TLEs to help augment our behavioral maneuver analysis, as well as SpaceTrack object type labels for basic object type ground truth (Payload, Debris, Rocket Body).

A more realistic estimate of HBR compared to what is available in the public catalog is evaluated and can be used to train an ML model to perform HBR (size) regression.

Table 1: Comparison of Default HBR and LeoLabs Approach

Object Type	**Default HBR in Public Catalog**	**LeoLabs Approach**
Operational Payloads	Use operator provided value	Use operator provided value
Non-Operational Payloads	5 m	Use two independent open-source citations to calculate actual HBR value
Rocket Bodies	3 m	Use two independent open-source citations to calculate actual HBR value
Debris Fragments	1 m	RCS-derived if enough high-quality measurements exist, otherwise default to 0.5m

The HBR for debris fragments is determined using the median of the high-quality RCS values collected from individual radar passes using Equation 1 which follows:

$$\mathrm{HBR} = \sqrt{\left(\frac{\mathrm{RCS}}{\pi}\right)} \tag{1}$$

A high-quality tracklet (measurements in a single radar pass, for a single object), for LeoLabs S-band radars, has a signal-to-noise ratio (SNR) greater than 15dB and at least three measurements within the tracklet. A minimum of 50 tracklets and Gaussian distribution of the RCS in dBsm are needed for confidence in median RCS value. As the dataset of RCS measurements grows, the requirements for a high-quality tracklet and minimum number of tracklets will be re-examined and may be adjusted. These values are updated on a bi-annual basis.

It is likely over ~90% of the fragments on orbit have an HBR at least an order of magnitude less than the default value of 1 m used in the public catalog. Previous LeoLabs analysis used the RCS values provided by the 18th SDS of over 10,000 fragments from June 2022 and ~97% of the debris fragments had an HBR less than 0.2 m (small), ~3% of the fragments had an HBR as large as 0.56 m (medium), and less than 0.1% had an HBR greater than 0.56 m (large).

The 18th SDS radar cross section (RCS) families are categorized based on their sizes as follows:

- **Small** RCS: $< 0.1\,\mathrm{m}^2$ with corresponding HBR of $< 0.18\,\mathrm{m}$.
- **Medium** RCS: $0.1\,\mathrm{m}^2 < \mathrm{RCS} < 1\,\mathrm{m}^2$, with a maximum HBR of $0.56\,\mathrm{m}$.

- **Large** RCS: $> 1\,\mathrm{m}^2$, with a minimum HBR of $0.56\,\mathrm{m}$.

For any given characterization of an object, the past year of data is queried from LeoLabs' data lake infrastructure, and fed into the model pipeline. How each input and output data type is relevant to our approach is discussed in detail in Section 2.2.1 which follows.

2.2 Implementation

2.2.1 Feature engineering and pre-processing

As we are adopting ensemble machine learning methods, one of the most crucial parts of the process was feature engineering; for each model, we had to carefully devise and generate specific transformations of raw radar data that were predictive of a specific object's physical characteristics, morphology, and also critically, its behavior.

To enable the models to efficiently capture and represent the distribution of various predictive features of objects in the public SATCAT, we applied several pre-processing techniques that allowed the models to learn this compressed information effectively.

To do this, one of the key transformations that proved pivotal in our analysis was grouping all the data collected by the object in question, for the past year of data, and calculating a number of indicative percentiles for different metrics. These included the minimum, 1st percentile, 5th percentile, 25th percentile, median, mean, 99th percentile, and maximum, which were generated for most features.

To help predict basic object type (payload, debris, rocket body), features that primarily correspond to the different sizes and shapes of the satellite were selected. These include RCS, SNR, and tracklet length (number of measurements for a given radar pass). We define RCS (σ) below in Equation 2 as:

$$\sigma = \frac{\mathrm{SNR} \cdot (4\pi)^3 \cdot R^4 \cdot kT_s \cdot L_s}{P_t \cdot G_t \cdot G_r \cdot \lambda^2 \cdot \tau} \tag{2}$$

where:

- σ (Radar cross section): Measurement of the equivalent area seen by a radar.
- **SNR** (Signal-to-Noise Ratio): Measurement of signal strength relative to noise.
- **R** (Range): Distance between the radar and the target.
- T_s (System Temperature): Default system temperature in Kelvin.
- **k** (Boltzmann Constant): $k_B \approx 1.38 \times 10^{-23}\,\mathrm{J/K}$
- $\mathbf{L}_s$ (System Losses): Represents losses in the radar system.
- λ (Wavelength): The wavelength of the radar signal.
- τ (Integration Time): Time over which the measurement is integrated.
- $\mathbf{P}_t$ (Transmit Power): Default transmit power of the radar system.
- $\mathbf{G}_t$ (Transmit Gain): Gain of the transmitting antenna.
- $\mathbf{G}_r$ (Receive Gain): Gain of the receiving antenna.
- **d** (Duty Factor): Factor accounting for the radar's operational duty cycle, representing the fraction of time the radar is actively transmitting.

The tracklet RCS metric is a smoothed, fitted RCS value across all measurements in a given tracklet; it uses the changing SNR values as the object passes across the beam to get a best fit for the gain of this effect. Correcting SNR for this gain on each measurement ideally produces a constant value across the beam - making it a high-confidence estimate. The SNR defined as signal-to-noise ratio gives similar information but on a per-measurement basis, rather than based on a full tracklet.

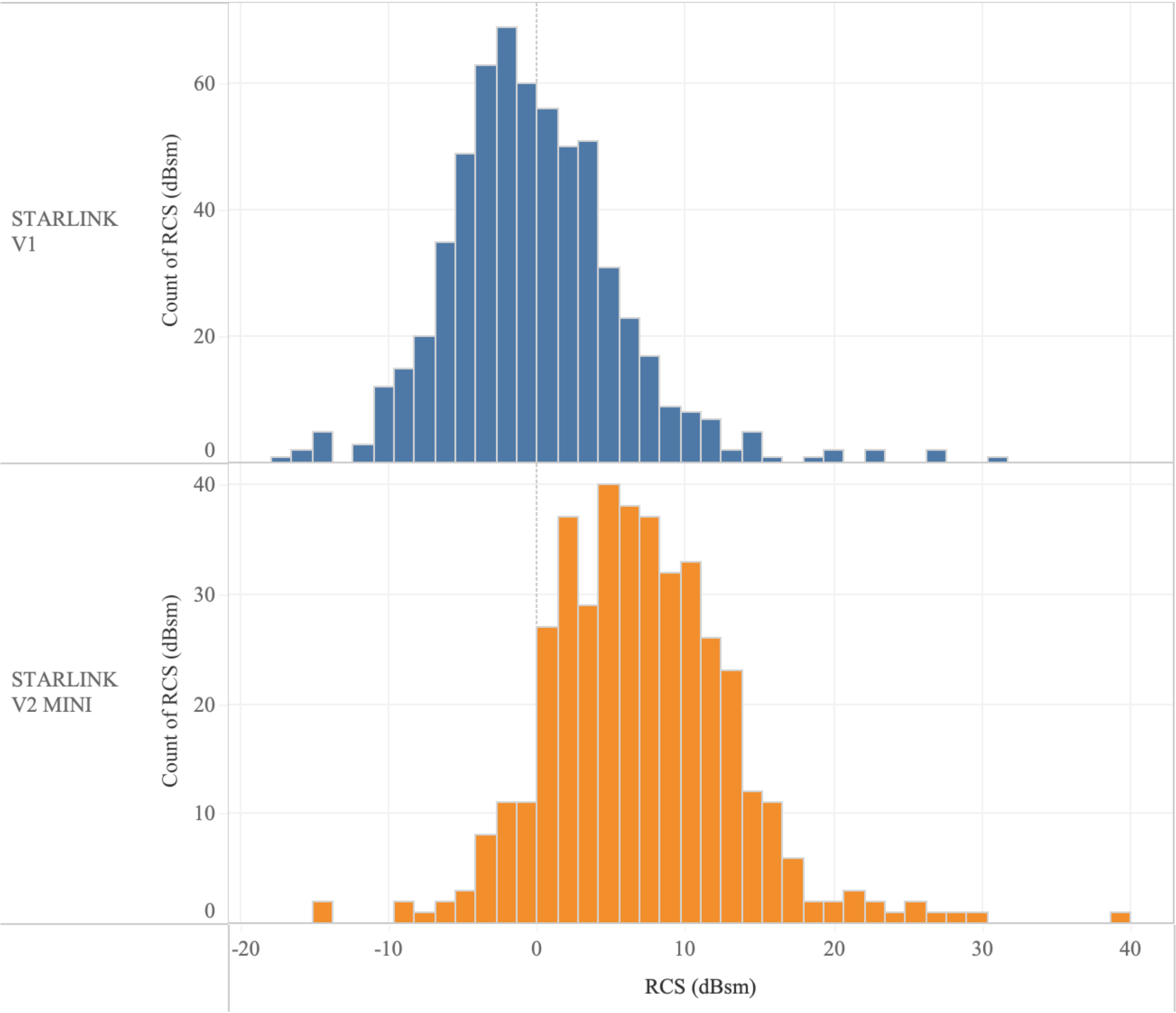

Fig. 3: RCS Distribution for Starlink V1 vs V2 - Note for the larger V2 Mini, the median RCS is also larger

A higher average RCS can typically be indicative of a larger satellite and vice versa. Multiple peaks in the RCS distribution (as seen in higher percentiles) may also indicate the presence of large structural features, such as solar panels or antennas, which can affect the RCS significantly when oriented toward the radar.

For example, we can see this by looking at the RCS distribution for different generations of SpaceX's Starlink satellite in Figure 3, the V2 Mini shows a clearly right shifted median RCS in dBsm, which can probably be attributed to the Starlink V2 Mini's larger 30m solar arrays [11].

The tracklet length describes the number of individual measurements that comprise a given tracklet at a given site, for a particular object. If a satellite is larger in size, it will likely be tracked over longer periods as it travels through the radar's field of view.

During the LeoLabs Orbit Determination process, we perform force modeling, which includes forming an estimate of atmospheric drag when we generate each state. This drag coefficient can contain information about the degree to which a satellite may be maneuvering, especially if the force from drag appears to be greater than expected, or changes rapidly.

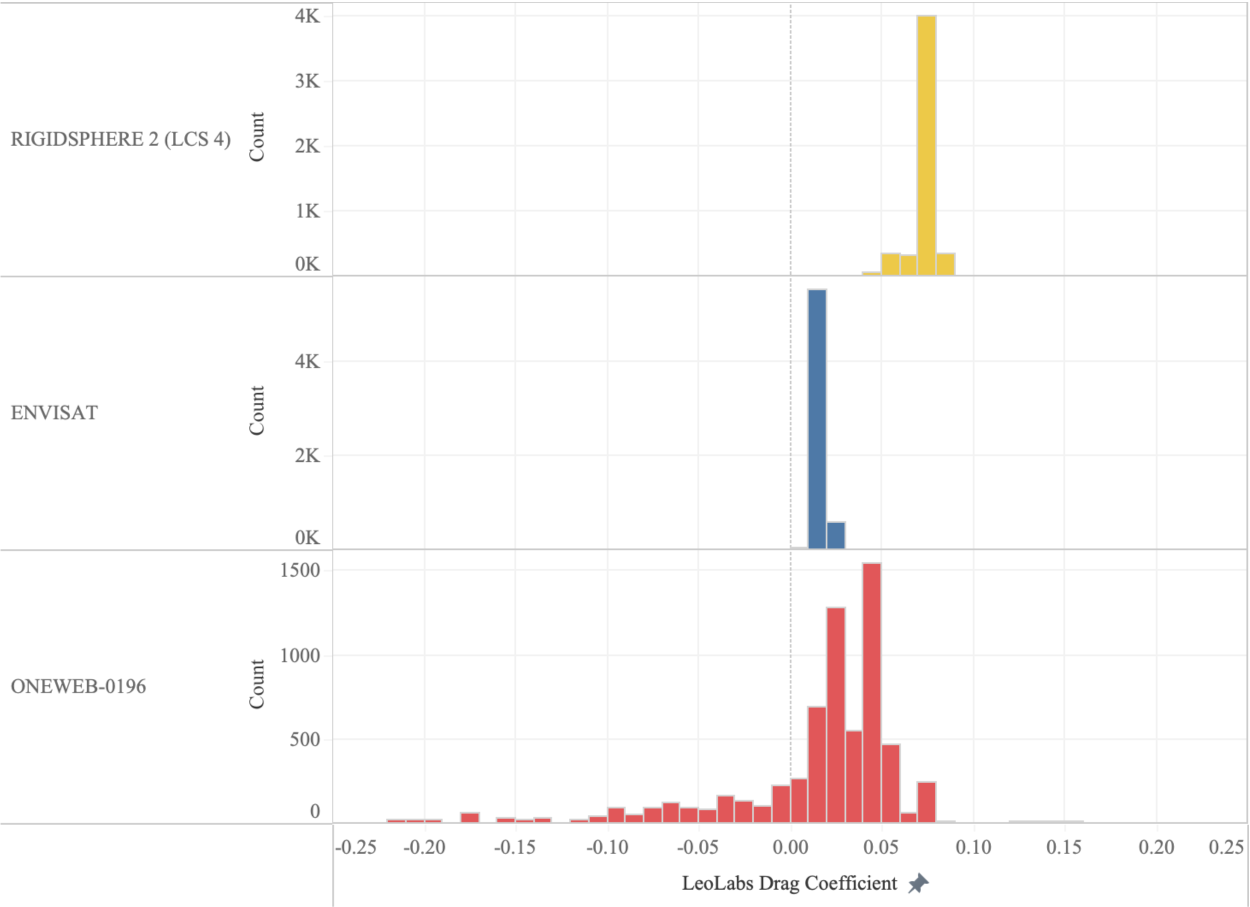

Fig. 4: LeoLabs drag coefficient distribution for non-maneuvering satellites: Rigidsphere and Envisat vs a maneuvering OneWeb constellation satellite.

For example, seeing many negative drag coefficient values may indicate copious maneuvering activity by a satellite for the observed period. This can be observed in Figure 4 which shows non-maneuvering satellites having only positive drag coefficients close to zero, while the maneuvering OneWeb satellite's distribution skews negative.

Since a large component of detecting maneuver behavior is time-based, it is essential that we observe a broad set of states over an extended period for each object and attempt to characterize the frequency and magnitude of such maneuvers.

To do this, a year of both LeoLabs and SpaceTrack generated TLEs detailing the altitude history (trivially derived from the semi-major axis for each object in the catalog) are pre-processed, and maneuver detection is run on them.

Additional features predictive of maneuverability are produced by other LeoLabs ML models (decay prediction) and create a number of maneuvering metrics.

For all states in a given object's timeline, these are the features generated:

- **Maneuver rate** – How many post-maneuver states are detected to have been maneuvering states divided by the total number of the states in the timeline for that object.
- **Grand altitude change** – From the start of the timeline to the end, what is the total altitude change, normalized

by time.

- **Altitude delta per hour** – From state to state, how much does the altitude vary, normalized by time.
- **Integrated maneuver score** – Maneuver rate multiplied by altitude delta per hour, it gives a representation of the cumulative maneuvering ‘area’ by taking into account both the frequency of maneuvering and their magnitude.
- **Absolute Delta V** – The absolute change in velocity between states.
- **Maneuver distance score** – A calculated score based on deviation from natural decay expectations.
- **LeoLabs drag coefficient** - The LeoLabs orbit determination derived drag coefficient.

Adding another level of specificity to reduce noise and to distill more highly predictive features for object maneuverability, the states were also further filtered to include only those where a maneuver was detected, and the following features (whose calculation is also defined above) computed:

- **Altitude delta per hour**
- **LeoLabs drag coefficient**
- **Absolute Delta V**

We reduced the model complexity by trimming features that produced the lowest feature importances, and also by using backward stepwise regression.

2.2.2 Architecture

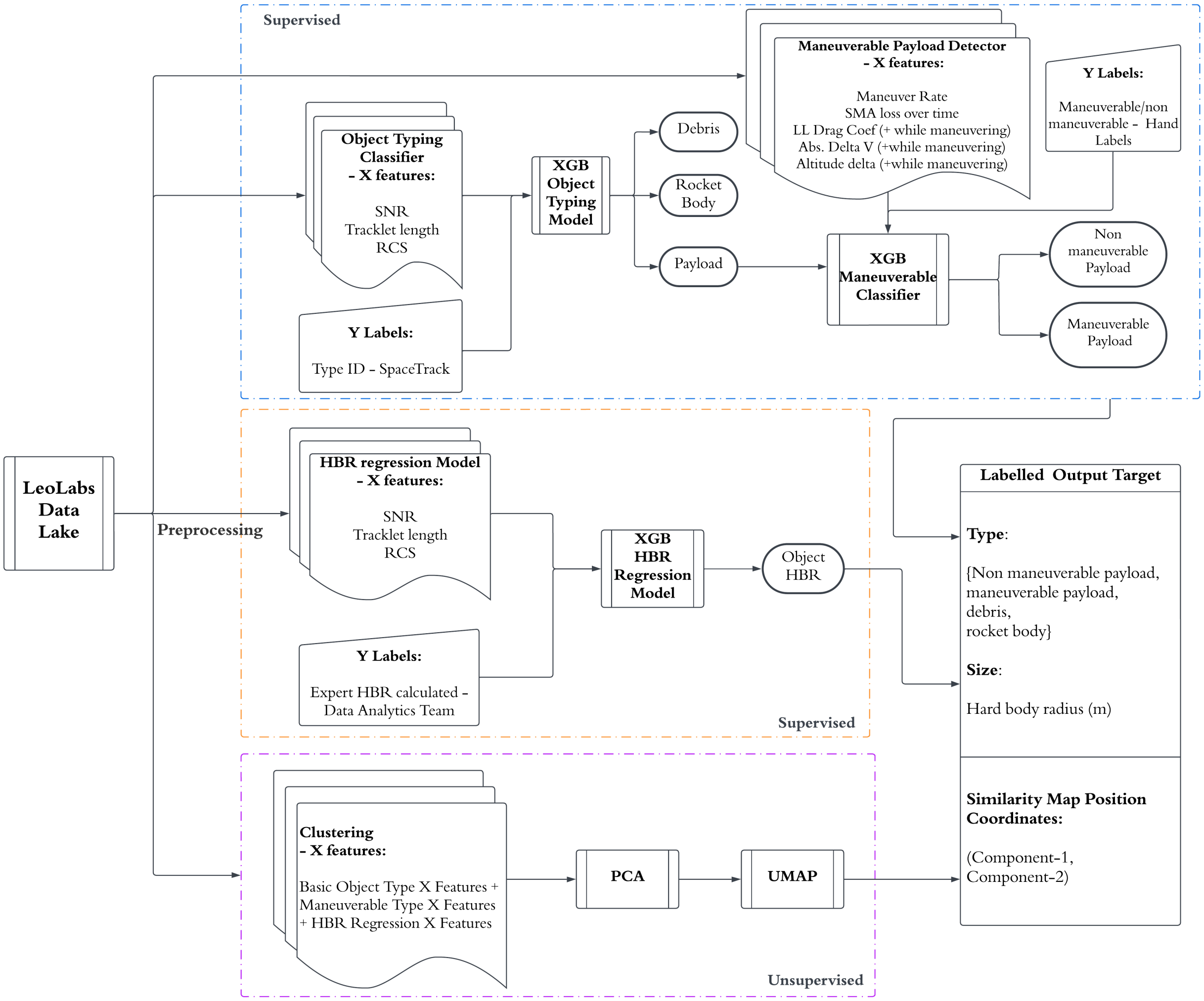
Supervised
LeoLabs Data Lake
Preprocessing
Object Typing Classifier - X features:
SNR
Tracklet length
RCS
Y Labels:
Type ID - SpaceTrack
XGB Object Typing Model
Debris
Rocket Body
Payload
Maneuverable Payload Detector - X features:
Maneuver Rate
SMA loss over time
LL Drag Coef (+ while maneuvering)
Abs. Delta V (+while maneuvering)
Altitude delta (+while maneuvering)
Y Labels:
Maneuverable/non maneuverable - Hand Labels
XGB Maneuverable Classifier
Non maneuverable Payload
Maneuverable Payload
HBR regression Model - X features:
SNR
Tracklet length
RCS
Y Labels:
Expert HBR calculated - Data Analytics Team
XGB HBR Regression Model
Object HBR
Supervised
Labelled Output Target
Type:
{Non maneuverable payload, maneuverable payload, debris, rocket body}
Size:
Hard body radius (m)
Similarity Map Position Coordinates:
(Component-1, Component-2)
Clustering - X features:
Basic Object Type X Features + Maneuverable Type X Features + HBR Regression X Features
PCA
UMAP
Unsupervised

Fig. 5: Multi-layered ML Model Architecture

The full architecture of our system detailed in this section is attached and can be referred to in Figure 2.2.2 above.

For all supervised machine learning models we decided to use XGBoost, which stands for Extreme Gradient Boosting. This is a scalable, distributed gradient-boosted decision tree (GBDT) machine learning library [7].

The reason for this is that it has been shown that XGBoost trains rapidly and also outperforms many other methods of ML on tabular data when using structured data with a relatively small feature set [9]. The first two layers of the Object Characterization are based on supervised XGBoost models.

Layer 1 – Basic Object Type and Size: The purpose of the first layer of the ML model is to characterize the general shape and size of an unknown object, with no regard for its behavior at this stage. As such the model is trained on the transformed features: tracklet length, RCS, and SNR.

The outcome types used to train this layer are the publicly provided SpaceTrack SATCAT basic object types. The sizes used are hand-labelled hard body radius data provided by our data analytics team from operator and other third-party information that is available publicly.

This produces an XGBoost classifier, the output of which is a basic object type: Payload, Debris, or Rocket Body, and an XGBoost regressor, whose output is a hard body radius (size). If the object already has a size as given by our data analytics team, then we use that value, if no size is provided it will be predicted using our XGBoost HBR regressor.

Any payload identified by SpaceTrack, or any unknown object that is classified by the first layer as a payload is then passed to the second layer of our ML solution.

Layer 2 - Maneuverability characterization: The second layer allows any previously identified payloads to be further classified into whether they have maneuvered or not during the period in question.

This consists of a XGBoost decay prediction model which looks at a fused LeoLabs and SpaceTrack TLE timeline of a particular object (de-noised and pre-processed to account for anomalies); it identifies maneuvers by estimating natural decay due to atmospheric drag, and then flags orbital state changes that exceed those natural thresholds.

These identified maneuvering states are used to generate the relevant maneuverability features mentioned in Section 2.2.1.

The XGBoost binary classification model is then trained on these features, and whose ground truth outcomes (maneuverable or not) are based on a dataset created by LeoLabs expert analysts looking at any given object's orbital element timeline, and judging whether or not the object maneuvered during its orbital timeline.

The output of this model therefore further classifies payloads as either non-maneuverable or maneuverable payloads.

Layer 3 – Clustering: The final layer in our multi-tiered machine learning approach uses an unsupervised method to project the feature space of all the previous modeling layers onto a simpler 2-dimensional space to help reason about the similarity of all objects; generating clusters whose distance represents both physical and behavioral propinquity.

To do this we fetch the set of all generated features from the basic object type, maneuverable type, and size regression models, and use principal component analysis (PCA) to initialize our clustering, a dimensionality reduction technique that transforms a large set of features into a smaller set while retaining the most important features that explain the majority of the variability in the data. Using PCA allows the succeeding projection and clustering process to be more robust as it denoises the features by discarding unimportant variations, reduces the likelihood of overfitting, and allows the manifold transformation to focus on the principal components that represent the major and more stable trends in the underlying relationships [4].

Uniform Manifold Approximation and Projection (UMAP) is a technique used to simplify complex, high-dimensional data (the many transforms and aggregations of radar data) into 2D data, to facilitate easier visualization.

UMAP begins by modeling the local structure of the data using a probabilistic approach. For a given data point x_i, it defines a neighborhood based on the local distances. This is represented by a conditional probability p_{ij} that quantifies the similarity between points x_i and x_j in the high-dimensional space. The probability is given by [6]:

$$p_{ij} = \frac{\exp(-\|x_i - x_j\|^2/\sigma_i^2)}{\sum_{k\neq i}\exp(-\|x_i - x_k\|^2/\sigma_i^2)}, \tag{3}$$

where σ_i is the scale parameter that adjusts the neighborhood size for point x_i.

UMAP then constructs a low-dimensional embedding by minimizing the cross-entropy between the high-dimensional and low-dimensional probability distributions. Let q_{ij} be the probability in the low-dimensional space, given by:

$$q_{ij} = \frac{(1+\|y_i - y_j\|^2)^{-1}}{\sum_{k\neq i}(1+\|y_i - y_k\|^2)^{-1}}, \tag{4}$$

where y_i and y_j are the corresponding low-dimensional representations of x_i and x_j.

The objective function UMAP aims to minimize is:

$$\mathscr{L} = \sum_{i,j} p_{ij} \log \frac{p_{ij}}{q_{ij}}. \tag{5}$$

Essentially, the method starts by understanding the neighborhood of each data point in the high-dimensional space, identifying which points are close to each other (similar) and which are not. UMAP then tries to project these points onto a 2D plane in such a way that those relationships are preserved as much as possible. Simply put, two data points that are close in the original high-dimensional space, should also appear close on the 2D graph.

UMAP balances local and global data structures. Locally, it makes sure that points that are neighbors in the high-dimensional space remain neighbors in 2D. Globally, it arranges these local groups in the 2D space so that the overall structure of the data is maintained. This process results in a 2D representation that can reveal patterns, clusters, and relationships that might be hard to observe directly in the original high-dimensional dataset [6].

A main alternative technique that was explored for this final layer of clustering was t-distributed Stochastic Neighbor Embedding (t-SNE). However, through experimentation with both projection methods, we found that UMAP tended to produce more stable and informative clusters for our use cases. In studies, it has also been argued to preserve more global structure than t-SNE while also being more computationally efficient [6].

After training the model on all our features, we then plot the two components produced by the UMAP projection into a 2D visualization, with each data point representing a specific satellite in the SATCAT and its aggregated metadata, to view and reason about what clusters have organically formed. Coloring these clusters with independent information that the model is not trained on can reveal what underlying structure the UMAP technique may be uncovering. This is explored in detail in Section 2.4.3.

2.2.3 Productionization

To provide relevant and useful insights to analysts, operators, and other users, the object characterizations generated need to be provided in a timely, continuous, reliable, and easily accessible manner.

Consequently, we decided to implement our full multi-layered ML pipeline using a Databricks Lakehouse architecture. This model lends itself well to our use case since Databricks workflows allow decomposition of the larger overall task into smaller, manageable components, each responsible for each specific part of the process, for example, pre-processing, training, validation, prediction, and visualization. In particular, it allows us to keep extract-transform-load (ETL) and ML pipelines in the same platform for seamless connectivity.

Databricks Lakehouse solution is also highly scalable, and we can rapidly provision specific distributed compute resources to process large amounts of training data on a regular basis. Furthermore, it also affords better maintainability and extensibility for the future since individual components or layers can be modified, added, or subtracted seamlessly as new models mature and are added to the multi-layered system. Lastly, Databricks also allows us to schedule automated runs of the full pipeline so that the resulting data automatically stays up to date and accessible for all users without manual intervention [3].

All required data for the multi-layered process are loaded into delta lake tables in Databricks, the processing performed on their compute clusters, and results are saved into a delta lake table, which can be easily queried directly, or connected to our visualization tool of choice.

After some experimentation, we chose to use the Tableau framework over a static plot for the visualization of the LeoLabs Similarity map; this allows for more interactive and dynamic exploration of the data, where analysts and operators can drill down into specific regions of interest in the map, focusing on closely clustered families or even individual satellites. Myriad metadata can be overlayed and clearly displayed through the use of a number of user-interface elements such as coloring, marker shape, size, and others. Tableau allows effortless sharing of dashboards between multiple users, enabling collaborative analysis and discussion about the data.

2.3 Validation and Performance

2.3.1 Basic object type and size

Our truth labels for the basic object types (payload, debris, rocket body) come exclusively from types given by the 18th SDS through Spacetrack. We used repeated stratified 5-fold cross-validation to measure model performance (using 80% of all Spacetrack object type labels as the training set, and 20% as the test set, repeated ten times). The resulting mean cross-validated accuracy across all repeats was 0.907 ± 0.005. A sample confusion matrix for one arbitrary fold is shown in Table 2.

True label / Predicted label	Payload	Debris	Rocket Body
Payload	0.90	0.05	0.05
Debris	0.04	0.95	0.01
Rocket Body	0.33	0.03	0.64

Table 2: Basic object type confusion matrix

As we can observe, the basic type model performs extremely well on payload and debris recall, but not as well on rocket bodies. Diving deeper into such cases, we observe that some of these rocket bodies seem in most features to be very similar to non-maneuverable payloads. Regardless, for our use case we are mainly concerned with maximizing payload prediction accuracy, as these are the most relevant dynamic group of objects that are passed down our ML system chain to be further processed for maneuver capability.

For HBR size regression, we also used 5-fold cross-validation. The resulting mean R^2 across all folds was 0.893 ± 0.0158.

2.3.2 Maneuverable Type

Our truth labels for the maneuverable payload types (non-maneuverable payload, maneuverable payload) are populated by operator maneuverability information, combined with LeoLabs hand labels by expert analysts. As Starlinks comprise the vast majority of maneuverable payloads on orbit (over 6000), we downsample this constellation to 500 random Starlinks to allow for more representative training; however, it is important to note that this also negatively impacts the reported accuracy that follows. Again, we used stratified 5-fold cross-validation, with ten repeats, to measure model performance. The resulting mean cross-validated accuracy after downsampling was 0.917 ± 0.008. An arbitrary confusion matrix for one particular fold is shown below in Table 3.

True label / Predicted label	Maneueverable Payload	Non-Maneueverable Payload
Maneueverable Payload	0.91	0.09
Non-Maneueverable Payload	0.07	0.93

Table 3: Maneuverability Confusion matrix

As we can see, this binary classifier performs well on maneuverability status, which is a crucial component of determining behavioral patterns.

2.3.3 Clustering and Similarity Map

Validating clustering accuracy and performance is a more subjective matter, since ground truth is less readily available. Unsupervised methods do not use labels to train, implying no one clearly "correct" interpretation. In addition, because we are reducing an extremely complex feature space with multiple tens of dimensions down to just two, as a result, some information may be lost, making it difficult at times to judge the quality of the projections.

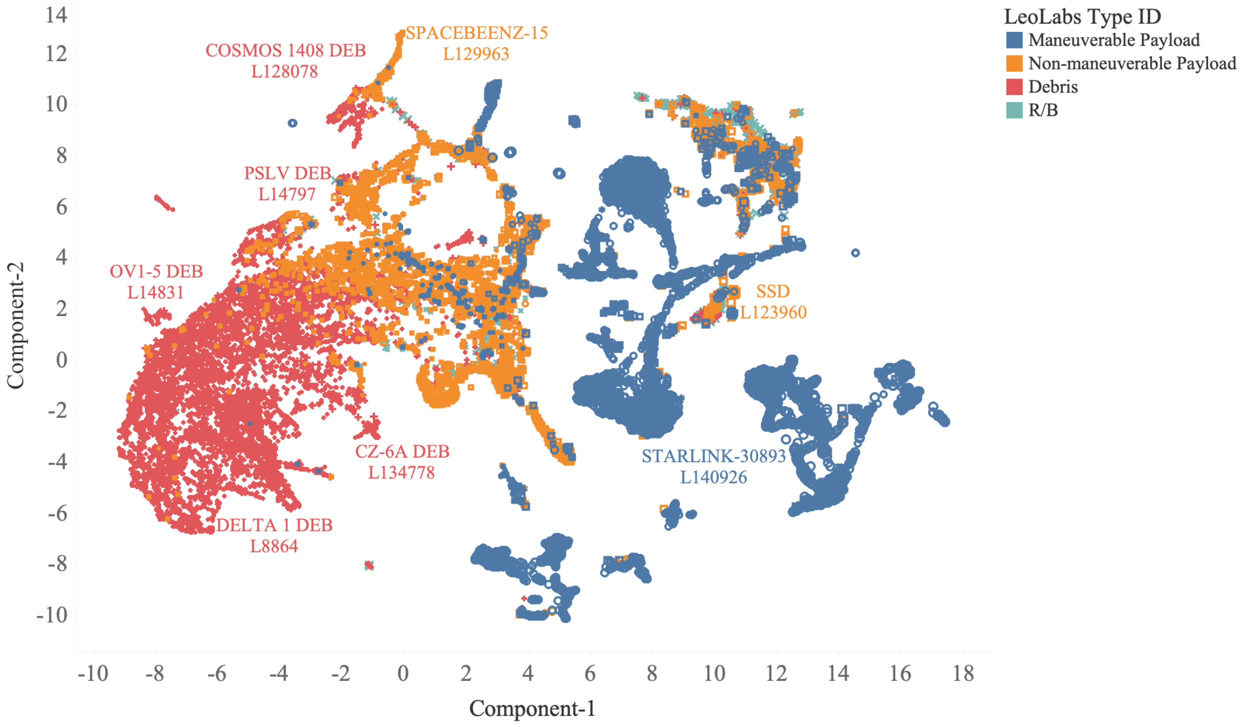

Fig. 6: LeoLabs Similarity Map projected using UMAP - colored by object type

However, a few techniques that we explored that proved helpful when assessing the quality and underlying structure of the clustering.

As we can see in Figure 6, when we overlay the output label of our known and predicted object types: non-maneuverable payload, maneuverable payload, rocket body, and debris as colors on our UMAP projection, the different types tend to cluster well together. This is strong evidence that the UMAP embedding is learning about the differences between object types without even providing any labels for them.

In addition, we can also highlight different families of satellite, for example, the Starlink constellation in Figure 7. When we do this, we can clearly see they are distinctly separated into their own clusters, indicating the similarity in physical and behavioral characteristics picked up by our UMAP processing. The V2 Starlink group nicely together, with V2 minis, being grouped at the rightmost side.

Coloring the similarity map with median RCS also reveals loosely how the primary projected components are ordered, with higher RCS satellites (such as rocket bodies, Starlink, and other bright objects) being grouped more towards the right extreme of the map, and smaller debris objects being grouped towards the leftmost side.

3. USE CASES AND IMPACT

Our multi-layered ML object characterization system offers a wide range of applications and use cases for space situational and domain awareness, operational planning, and post-launch analysis.

One of the most compelling use cases for our system is rapid object characterization directly after launches. Traditionally, for multi-payload rideshare missions, it can take up to two weeks after a launch for objects to appear in the public SATCAT, and perhaps several days or weeks for these unknown objects to be typed in basic object types. In addition, there is no public information pertaining to the maneuverability status of many satellites (i.e. any non-cooperative launches), so promptly providing this essential information alongside similarity and clustering characterization with other satellite subfamilies can prove to be a boon to many operators and agencies.

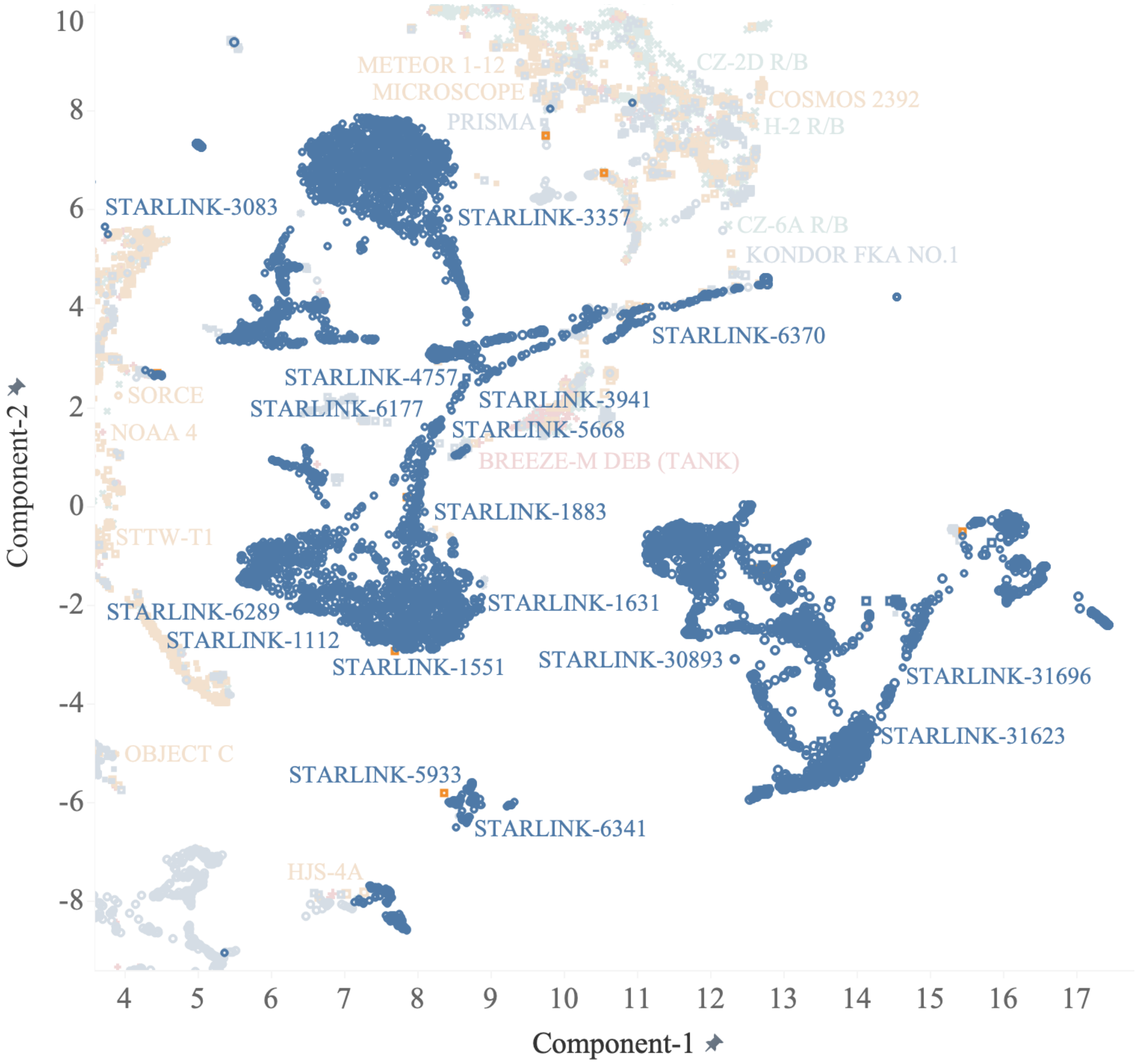

Fig. 7: Highlighting Starlink on the LeoLabs similarity map - they form neat clusters, with the V1 generation being clearly separated from V2 (rightmost cluster).

The cleanest example of our Object Characterization dashboard providing rapid assessment of post-launch payloads can be found during Starlink launches. Starlink launches almost always place tens of satellites with an identical form factor, in the same orbit, all at once. The transition from early on-orbit behavior to operational behavior can be clearly observed in our Object Characterization dashboards. Clustering of the newly launched Starlinks begins almost immediately, with a loose structure forming quickly post-launch. Over the course of a few days, as the satellites begin transitioning into their operational phase, the clustering tightens significantly. Eventually, we see these newly launched Starlink satellites falling into the broader cluster that captures their given Starlink generation (Figure 9. Given this characterization is done without any prior knowledge of the identity of payloads, only radar data, the applicability of this method for non-cooperative launches is significant.

For non-cooperative multi-payload missions, where information is limited regarding mission intent, payload count, and onboard capability of the launched satellites, our system can provide rapid assessment of the unexpected.

An example of this is China's Qianfan constellation, whose goal is to provide similar capabilities to SpaceX's own

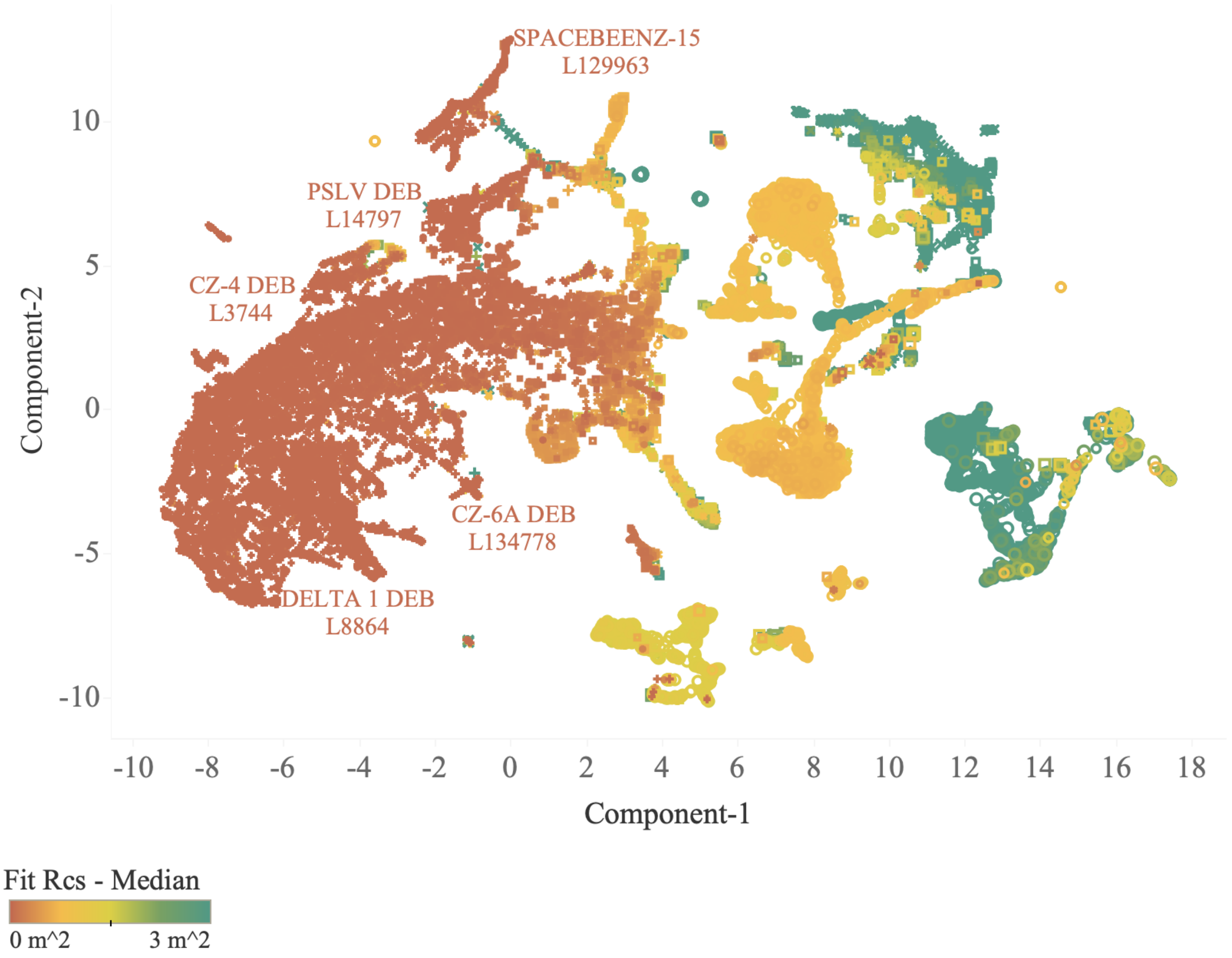

Fig. 8: Coloring the similarity map by median RCS reveals a striking increasing trend along the primary component of our UMAP.

Starlink satellites. The expectation is these satellites will each exhibit a similar form factor and physical characteristics, making them relatively easy to classify into a cluster, as has been done for Starlink in the LeoLabs Similarity Map. Variations to this form factor or mission can be difficult to determine when lost in the scale of an expected 14,000 satellite constellation [1].

As launch cadence increases and the payloads on board these launches increase in quantity, the problem of identifying abnormalities in the payloads launched worsens. Automated object characterization can identify abnormalities in launched objects without requiring labor-intensive manual work from an analyst. Rapidly identifying changes in form factor post-launch may provide indications of differences in generations of technology (e.g. indicating a progression from a V1 to V2), anomalies on board that prohibit the spacecraft from functioning properly, or even an object not expected on the manifest being launched alongside those that are reported publicly, and that we therefore expect to observe.

Multi-payload rideshare missions provide a unique opportunity to deliver potentially threatening assets on orbit while obfuscating the capability in the noise of an ever-growing commercial constellation. Object characterization can address this concern by identifying which asset "bucks the trend" of nominal behavior, and allows analysts to commit more resources to in-depth analysis, synthesizing characterization information across multiple data points.

Lastly, accurately giving operators the best estimate of object type, size, and general similarity of unknowns to other known objects can prove beneficial by complementing the public catalog with additional characterization information that can be used to derive relevant operational insights such as mission intent, and future behavior prediction.

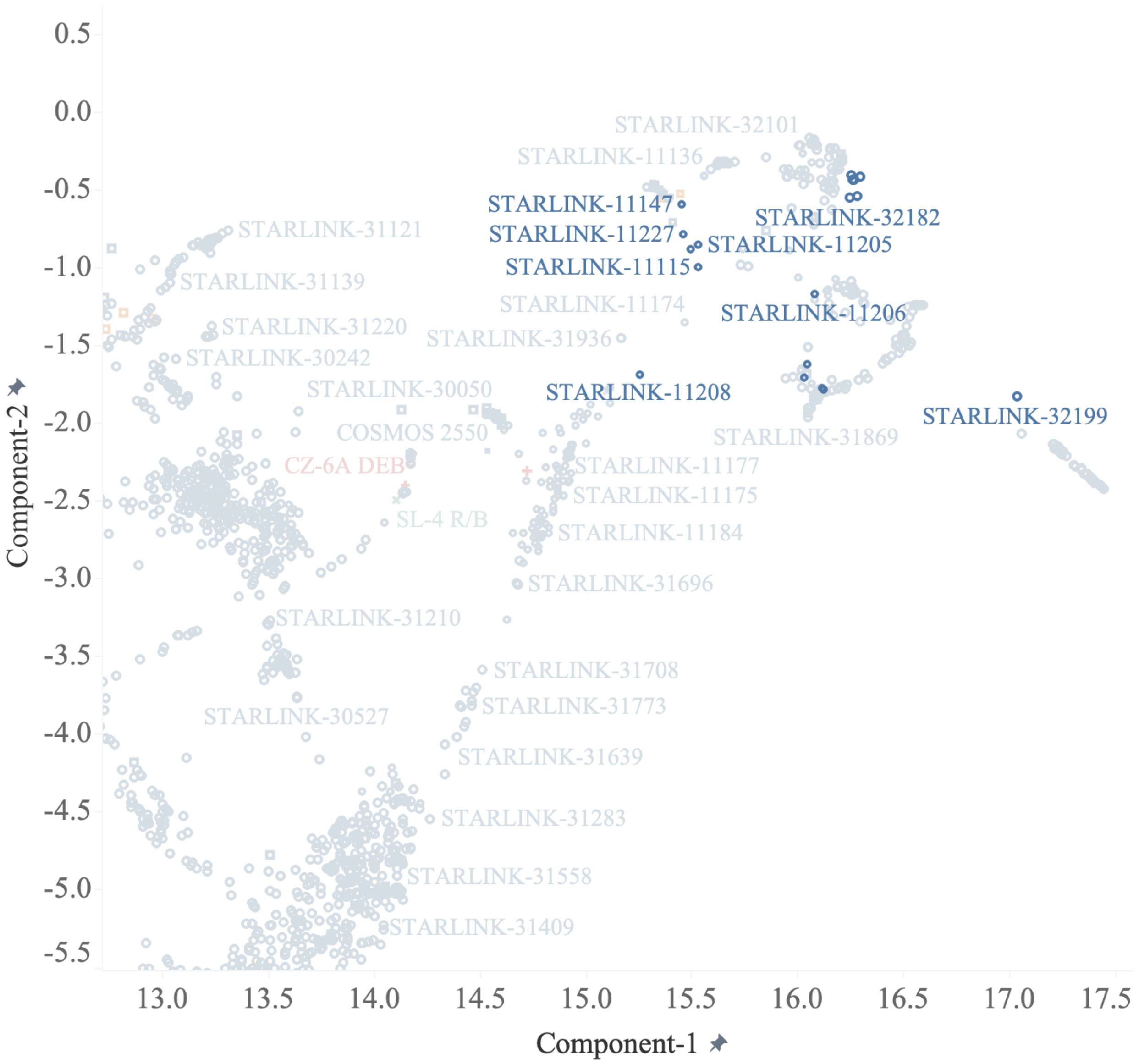

Fig. 9: Payloads from a given Starlink launch cluster closely together

4. FUTURE PLANS

We can continue to extend the robustness of object characterization through the use of LeoLabs' curated catalog and Space Domain Awareness specific attribute tags compiled from expertise coupled with open-source research; training future iterations and enhancing object profiling. These attributes can include additional data such as mission type, launch information, and on-board capabilities. Plans include more precisely assessing the most likely range of delta V based on the propulsion system and a more comprehensive maneuvering history.

There is also ongoing research at LeoLabs to explore Patterns of Life analysis for objects in LEO. This not only involves classifying in a binary manner (whether or not a satellite maneuvered) but also actively predicting the specific type of maneuver (a prototype of this tool is shown in Figure 10). For example, these can include but are not limited to, station keeping, circularizing, orbit raising, lowering, or de-orbiting. These additional insights can afford operators a higher-resolution picture of what a typical behavior profile for a specific family of satellites could look like over the

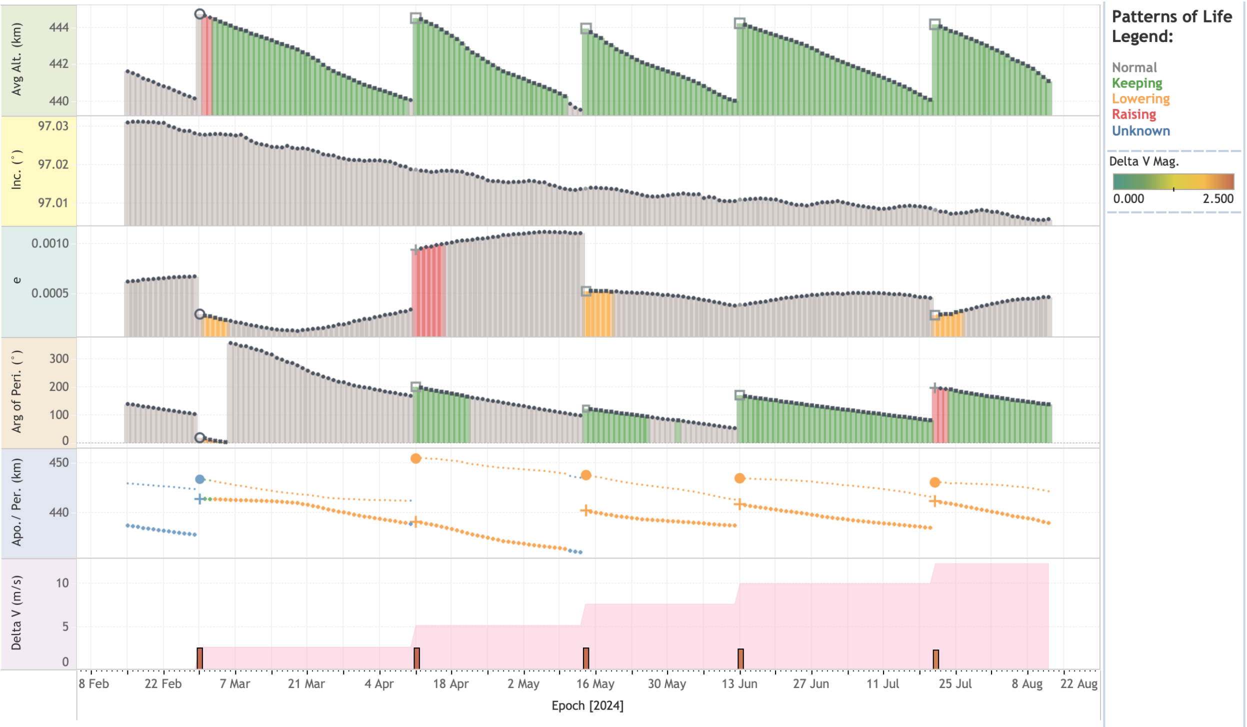

Fig. 10: LeoLabs Patterns of Life prototype - highlighting the detection of different subclasses of maneuvering

course of its lifetime.

In this first iteration, we've limited the tool to predicting whether an object can maneuver or not. In the future we see a large opportunity to be specific about the kinds of maneuvers we've seen from an object, including whether the object appears to maneuver in response to conjunctions, whether it is station keeping, the largest delta-v we've observed, and what this might say about its propulsion type. In general, we see an opportunity to create a tool that summarizes both the static physical characteristics of an object, but also the behavioral patterns so they can be cross-referenced against one another to find interesting useful anomalies quickly.

By incorporating even more on-orbit information LeoLabs can expand Object Characterization to define mission type and patterns of life with higher precision and granularity. Certain mission sets, such as Earth observation and imagery, maintain orbital element regularity over extended periods, and therefore these stereotypic behavior profiles may be detected and characterized autonomously. This will expand the utility of our system beyond simply the physical characteristics and maneuvering ability of a spacecraft, and begin identifying more complex behavior and higher-level mission intent. This will not only provide more clarity to an operator regarding a satellite's similarity to other targets on orbit, but also provide more useful clustering in the LeoLabs Similarity map - clusters indicating specific mission types or lifecycles.

Our Object Characterization dashboard can also be upgraded to give more detailed information about surrounding satellites, for example, a list of the ten 'closest' objects in the LeoLabs similarity map. This list of closest neighbors may allow inferences to be made about the object's mission type or family.

Further down the roadmap for our tool, we could feasibly train a Large Language Model (LLM) to automatically integrate and synthesize data from all the current and future layers of our full Object Characterization system. This would allow an intuitive interface to users that could provide a holistic summary of key metadata and characteristics of particular satellites or families of satellites.

An LLM could also be trained to better interpret the results of unsupervised clustering using UMAP, relaying natural language summaries and clustering rationale to operational users. For example, a generated possible explanation could include something similar to: "This clustering of debris generated from this breakup event appears consistent with a high relative velocity collision". LeoLabs already performs post-breakup analysis, evaluating likely causes of debris-generating events by leveraging data collected from our global network of radars following breakups and expert analysis. This analysis has allowed LeoLabs to assess the likelihood of debris-generating event types such as; propulsion or battery ruptures, low relative velocity collisions, high relative velocity collisions, and even anti-satellite weapons. In the future, through the use of our Object Characterization dashboard, these assessments can be made autonomously, further augmenting operational oversight and the origins of debris-generating events.

An LLM could also help explain anomalous satellite behavior and keep operators abreast of how object clustering in the LeoLabs similarity map is evolving over time. For example, an object may previously be typed as debris, but suddenly maneuver, causing it to change object type and shift clusters to another area on the map. Tracking many such events may become increasingly difficult for an analyst as mission sizes increase, whereas an LLM may facilitate more effective operations, by automatically honing in on areas of interest to focus on.

In addition, by integrating similar thresholds as used for the RCS-derived HBR for fragments for determining high-quality RCS into the model (tracklets with SNR $>$15dB and 3+ measurements, resulting in a Gaussian distribution), we can potentially improve the performance of the ML models by improving the information contained in the RCS estimates. Longer term, these evaluations could also include a confidence or uncertainty metric associated with the approximation.

5. CONCLUDING REMARKS

LeoLabs is addressing a growing challenge in LEO caused by the ever-growing participation of many nations in satellite launches. The rapid rise in these satellite deployments necessitates advanced, automated methods for accurately characterizing and classifying objects in space. Our global network of radars produces the extensive raw data needed to accurately characterize unknown objects in LEO. Our cutting-edge multi-layered machine learning solution integrates both supervised and unsupervised methods, offering significant improvements over current conventional approaches. By enhancing these capabilities, we aim to better manage the congested LEO environment, support the development of effective space traffic management systems, and contribute to the sustainability and security of global space operations.

REFERENCES

[1] CNN. China's new satellite threatens spacex's starlink system. *CNN*, 2024. Accessed: 2024-08-09.

[2] Daniel R. D. *Global Space Governance: An Overview of International Space Law and Policy*. Cambridge University Press, 2023.

[3] Databricks. Databricks lakehouse: Unified data platform, 2024. Accessed: 2024-08-12.

[4] Ian T. Jolliffe. *Principal Component Analysis*. Springer, New York, 2002.

[5] O. Marshall, D. Ottesen, and D. Baker. Degradation of launch and early orbit. In *10th Annual Space Traffic Management Conference: Inflection Points of Change: Civil, Commercial, and Security*, Austin, Texas, Feb 2024.

[6] John McInnes, Lukas Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018. Version 1, submitted on 6 Feb 2018.

[7] NVIDIA. Xgboost, 2024. Accessed: 2024-08-05.

[8] B. Reihs, J. Rowland, O. Marshall, P.T. Williams, M. Stevenson, D. McKnight, H. She, and M. Shouppe. Increasing capabilities in a growing radar network. In *Proceedings of the 73rd International Astronautical Congress (IAC)*. International Astronautical Federation, 2022. Paper presented at the 73rd International Astronautical Congress (IAC), S04.

[9] Ravid Shwartz-Ziv and Amitai Armon. Tabular data: Deep learning is not all you need. *arXiv preprint arXiv:2106.03253*, 2021. Version 2, submitted on 23 Nov 2021.

[10] Space-Track.org. Space-track.org. `https://www.space-track.org`, 2023.

[11] Spaceflight Now. Spacex unveils first batch of larger, upgraded starlink satellites. `https://spaceflightnow.com/2023/02/26/spacex-unveils-first-batch-of-larger-upgraded-starlink-satellites/`, 2023. Accessed: 2024-08-12.

[12] U.S. Government Accountability Office. Highlights of gao-22-105166, a report to congressional addressees. `https://www.gao.gov/assets/gao-22-105166-highlights.pdf`, 2022. Accessed: 2024-08-05.