Automated Data Accountability for Missions in Mars Rover Data

Ryan Alimo
Jet Propulsion Laboratory,
California Institute of Technology
sralimo@jpl.nasa.gov

Dylan Sam*
Department of Computer Science
Brown University
dylan_sam@brown.edu

Dounia Lakhmiri*
GERAD
Polytechnique Montreal
dounia.lakhmiri@gerad.ca

Brian Kahovec
Jet Propulsion Laboratory,
California Institute of Technology
bkahovec@jpl.nasa.gov

Dariush Divsalar
Jet Propulsion Laboratory,
California Institute of Technology
dariush.divsalar@jpl.nasa.gov

This paper proposes an automated solution system to assist with Real-Time Operations and automatically identify and report on issues with data transfer, archive, and manipulation throughout the Ground Data System (GDS) process. As the Mars Curiosity Rover transmits data to the JPL Ground Data System (GDS), it frequently observes data loss and corruption, requiring re-transmits from the rover and Ground Data System Analysts (GDSA) to monitor the downlink process. As new missions are launched, the GDSA team redistributes analysts to these new missions, causing shortages in previous missions. The prior state of GDS issue detection and resolution was largely manual. GDSAs receive email alerts when something goes wrong, but it's not always clear what the exact problem is or how to fix it. This paper presents machine learning and deep learning based approaches to automate and optimize the detection of data loss. We first created a pipeline to automatically accumulate data from the telemetry databases (MAROS, Telemetry Data Storage, and GDS Elastic Search Database) in the downlink process. With our newly created datasets, we perform feature selection to supplement the GDSA understanding of the downlink process and provide supplemental analysis on the importance of different features. We implemented various supervised machine learning-based models and evaluate their accuracies to identify a downlink process is complete or incomplete. We utilize fast hyperparameter optimization methods that allow our models to quickly be re-trained, allowing them to quickly be tuned and optimized on daily incoming data in near real time.

TABLE OF CONTENTS

1 INTRODUCTION

The Automated Data Accountability for Missions (ADAM) diagnoses data fidelity issues accurately and automatically using data-driven algorithms. During the operation time of a Mars Rover mission, the Rover sends data from the surface of the Mars to Earth. This data travels at the speed of light but can experience several types of distortions as it travels through space such as cosmic events like solar flares or exotic particles can damage information on any of the platforms this data is relayed through, and any issues with the hardware or software onboard can cause more errors [2]. Any interruption in the data stream, whether due to rotations of planets or objects passing through the transmission on its way back to Earth, will often corrupt the critical science and operational information the mission depends on, see Figure 1.

ADAM is an autonomous system [1] that improves reliability and productivity while reducing risk and operational costs and potentially lowering development costs. ADAM will eventually function with minimal human intervention and supervision through Augmented Intelligence (AI). ADAM will allow real-time operations to run smoothly by monitoring Telemetry Data Storage and Ground Data Systems (GDS) machines to predict, recognize, and resolve issues and alert Ground Data Systems Analysts (GDSA).

The Mars Science Laboratory (MSL) Real-Time Operations team at NASA's Jet Propulsion Laboratory monitors the downlink process of telemetry data from the Mars Rover [4] back down to Earth. The current MSL downlink process includes the Mars Orbiters, the Deep Space Network (DSN) [5], JPL Data Control, and MSL's GDS [6]. During the transmission of data through this process, time stamps, data volumes, and other metadata are recorded to make sure the data is being transmitted successfully. However, there are frequent losses in data as it is sent through these multiple locations, which sometimes require re-transmissions by the rover. There is a need for a better understanding of the cause of data loss and re-transmission, which can help the GDSA team better determine the root cause of the issues in the GDS.

The GDSA team is looking for a set of schemes to help automate and optimize the detection of data loss in the downlink process of telemetry data from space and particular Mars missions. As new missions are launched, the GDSA team redistributes analysts to these new missions, causing a shortage in previous missions. As a result, they are looking for technologies like augmented intelligence [11, 12] (i.e. machine learning extends human capabilities) to increase the performance of their monitoring tasks and to reduce their staffing needs. In this regard, reasoning-based machine learning approaches are particularly useful, as they can learn abstract and relational features in large datasets such as data transmission metadata. Although the GDSA team has an expert understanding of the feature space, machine learning algorithms allow the model to possibly learn other important features that are obscure even to the analysts. Supervised machine learning models [8, 31] can learn patterns from the historical data and generalize their predictions on new data.

*This project was performed while the author was a research intern at JPL.

These models can be applied to determine whether future Mars Rover data transmissions are corrupted and require re-transmission based on the recorded historical data. In this paper, we experiment with various supervised learning approaches to detect missing data in the MSL downlink process.

Real-time operations can run more smoothly by monitoring GDS machines and leveraging historical data from the MSL downlink process. In this setting, machine learning models can predict, recognize, and alert the GDSA of any incorrect and irregular behavior [4, 14, 47]. Therefore, the use of machine learning models has become prevalent in the analysis of data from projects in space [7, 13, 16, 17]. However, the frequent inability of scientists to deeply understand these models is problematic [18–22] and potentially a bottleneck to their applications. Scientists and analysts dislike machine learning methods that serve as black box modules, as they do not give further intuition or justification for the classification task. Thus, there is a need for opening this black box and understanding why such machine learning methods work. In fact, in many situations explainable but slightly more accurate models are preferable to more accurate but unexplainable models. In supervised learning, several works suggest that explainability is related to trust [25], where trust refers to model's performance, robustness, or ability to detect a causality between features and labels.

In this work, we compare the results of different supervised learning models to the existing state of the art of GDSA software and analysts' performance.

In this context, finding the best setting for each supervised learning scheme is a necessary preliminary step since the final performance of each method is highly sensitive to the choice of a certain number of hyperparameters (HPs). Methods such as SVMs need to tune only a small number of HPs compared to deep neural networks where different aspects are taken into consideration from the structure or architecture of the network, from how the DNN learns from the training data, etc. These hyperparameter optimization processes often rely either on exploiting prior knowledge on the problem or on using sophisticated algorithms to quickly and efficiently find an optimum configuration.

The paper is organized as follows; following the introduction, Section 2 provides background for the data transmission issue and an overview of the problem statement. Section 3 describes the proposed approaches to the data accountability problem. Then, the adopted algorithms are outlined in more detail in Section 4, and a discussion of how to make the methods more explainable is provided. Section 5 presents an analysis and comparison of the performance of the proposed machine learning strategies. Finally in Section 6, we highlight our important performance and explainability results and state the larger-scale impacts of our methods.

2 MSL'S DATA ACCOUNTABILITY PROBLEM

Mars Rovers, like Curiosity and Perseverance, enable in-depth exploration of the Martian surface and send its data to Earth. The Mars Science Lab Real-Time Operations team encounters a variety of challenges in extracting data and maintaining the operations of the Curiosity rover. The downlink process to receive the rover's data transmissions is complicated; as seen in Figures 1. A small error in data transmissions when operating on Mars or another planet can lead to critical failure. The operational data, information about rover itself, and scientific data, data captured from Mars surface, follows a multi-stage transmission path: from the rover to satellites orbiting Mars, to Deep Space Network stations back on Earth, and finally to Mission Control. Data can be interrupted or corrupted at any of these stages, limiting the data's usefulness. The ground analysts must scour this deluge of downlinked data to find these interruptions, determine their root causes, and correct them before sending any new commands, like moving or using an instrument [2].

Figure 2 shows the process of sending telemetry data recorded by the rover back to the MSL team. First, the Curiosity Rover sends data to one of the Mars Orbiters. Then, the orbiter sends the data to one of the Deep Space Network stations. Afterward, the data is sent to the Jet Propulsion Laboratory (JPL), where it is received by Data Control and stored as transfer frames. Finally, the MSL team receives the data and converts the frames into packets and data products. GDSAs report the total data volume of the downlink in megabits [49].

We see in Figure 2 that the high-level architecture for the Mars Science Lab downlink process contains many different checkpoints and locations. Thus, there are many places where data can be corrupted or transmission can be interrupted. When problems occur, it is often difficult to determine the root cause, leading to the infamous question "Where is my data?". When a downlink is unsuccessful, it can take several hours for the GDSA team to diagnose the cause of an incomplete pass and request a re-transmission. By monitoring the MSL Ground Data Systems and using classification, we can determine when data is lost but not what happened and why it was missed.

In the downlink process, we record the metadata about each transmission of data at each location in Figure 2. These locations that store metadata are the Mars Relay Operations Service (MAROS) [15, 24], Telemetry Data Storage (TDS), and GDS Elastic Search Database. The GDSAs view these metadata to determine whether the data is being successfully transferred through the downlink process (complete passes) or not (incomplete passes). In this work, similar to the GDSAs, we curate a dataset of metadata from the MSL data sources to train supervised learning algorithms to detect when a downlink process is a complete or incomplete pass.

For our dataset creation, we built a data collector that gathers raw data from each of the three data sources and a signal processor that computes the important features from the raw data, which we use to train machine learning models.

The GDSA Dashboard currently automates the identification of complete and incomplete downlink passes [26]. Table 1 shows the performance of the existing GDSA labeler and the corresponding confusion matrix. In this paper, we aim to improve the accuracy and F1 score of the GDSA Dashboard platform so that it can be used more reliably in future mission operations.

Table 1 shows the performance of an experienced GDSA performance on the MSL's downlink passes with 9547 data points. The performance of the GDS labeler for incomplete are measured using precision, recall, f1-score defined as follows:

$$\text{precision} = \frac{|\text{downlink passes correctly labeled as incomplete}|}{|\text{downlink passes}|}, \tag{1a}$$

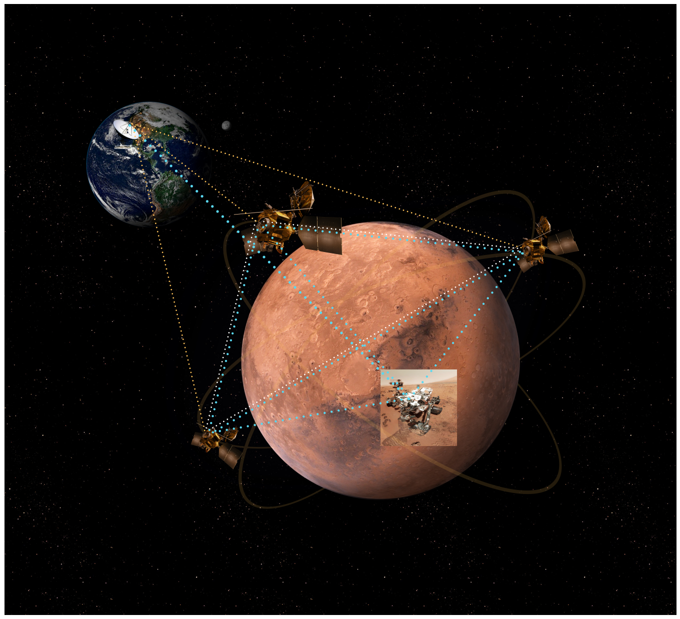

Figure 1. **A schematic illustration of Rover's telemetry data transfer pipeline.**

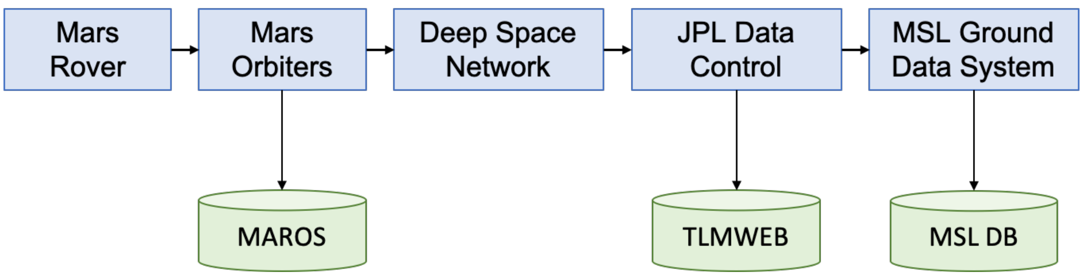

Figure 2. **High-level architecture of the Mars Science Lab Downlink Process.**

Table 1. Accuracy of the Ground Data System Analyst (GDSA) Dashboard Labeler [49].

Approach	GDSA		
	Complete (+1)	Incomplete (-1)	Total/Avg.
Precision	0.95	0.75	0.91
Recall	0.97	0.55	0.92
F1-score	0.95	0.64	0.91
no. pts	8287	1260	9547

$$\text{recall} = \frac{|\text{incomplete passes are labelled as incomplete}|}{|\text{downlink passes}|}, \tag{1b}$$

$$\text{f1} \; = 2\,\frac{\text{precision} \times \text{recall}}{\text{precision+recall}}. \tag{1c}$$

For example, when the GDSA gets the precision score of 0.75 on incomplete passes that implies that 75% of the GDSA passes that were predicted as incomplete were correctly labled. Also, when the GDSA gets the recall score of 0.56 for incomplete passes that implies that 56% of incomplete passes are labelled as incomplete.

GDS labeler's objective is to identify the situations that the downlink pass has not transferred successfully; as a result, the

In the following section, we compared the accuracy of the proposed approaches with the GDSA dashboard labeler's performance.

3 Data and Feature Engineering

In this work, we used a dataset consisting of downlink data from Sol [2] 337 to Sol 2450 from the Curiosity Rover (i.e., MSL downlinks in the year 2019-20). We utilized this dataset to evaluate different models' performance and determine for a given downlink pass if a significant portion of the data transmission was missed.

The overview of the data pipeline is briefly described in Section 2. We collected metadata [3] from the three different data sources in the downlink process using internal JPL APIs. As shown in Figure 2, MAROS, Telemetry Data Storage (TDS), and the GDS Elastic Search Database are the three different sources that store the metadata in the downlink process. In our dataset, data is transferred from Mars to Earth using six orbiters. In each data transfer, MAROS receives metadata from the Mars orbiters, TDS from JPL's ground data systems, and GDS from the GDSA Team's data processing. These API's return data volumes, start and end time of data

[2] a Sol refers to the duration of a solar day on Mars [43].

[3] Metadata is data that provides information about MSL data that is sent back to the Earth and is captured by the sensors for the scientific discoveries. Metadata is data about MSL data.

transmission, information about orbiter height and location, and other important features. For this study, we created a data collection and processing pipeline that queries each API and for all the data from a given Sol. In this pipeline, we combine the three sources of metadata (MARSO, TDS, and GDS) into one dataset by matching the downlinks pulled from each of the three sources. This process resulted in 9547 data points, which is the total amount of data that was available during the preparation of this paper. Although, the dataset may look small, particularly for deep neural network models, this automated pipeline gathers more data everyday, which helps to increase the performance of our machine learning models as more metadata is collected.

After collecting and merging our data, we use a sensitivity analysis scheme in the signal processing stage to compute the most relevant features of the metadata. In the validation process, we consulted with subject matter experts (i.e. Ground Data System Analysts) to gain prior knowledge about feature importance. Also, we calculated the disparities in data volumes between the different locations and between the actual and predicted volumes. Then we include both the differences in times and volumes as well as the raw amounts in our metadata. We construct one-hot vectors to represent from which Mars Orbiter the metadata was transmitted and at which Deep Station Network (DSN) station the metadata is received.

4 METHODOLOGY

This section presents the supervised machine learning algorithms that are investigated to identify complete and incomplete downlink passes.

Supervised learning models

A Supervised learning method is an algorithm that learns to map an input vector to a quantitative output for a regression problem or a qualitiative output for a classification problem, by training on a set of labeled data. The most commonly used models for classification are Logistic Regression (LR), Support Vector Machine (SVM) [39, 40] and Deep Neural Network (DNN). For the description of these approaches, we encourage the reader to review [23, 31]. We provide a brief explaination of the DNN approach.

Deep Neural Network (DNN) [4] models [31] use multiple fully-connected layers with nonlinear activation functions to learn the function mapping data points to their correct labels. With sufficient training and a sufficient number of parameters, DNNs are more robust to noise and perturbations in a dataset, compared to other supervised learning methods. DNN approaches are particularly attractive in this application, as the large number of parameters help the networks learn abstract and relational features from a complex dataset. They can augment the GDSA team's expert understanding of the feature space, by possibly learning other important features that are obscure. DNNs can utilize these learned relations to make accurate predictions on new data. Therefore, they can determine whether future Curiosity rover transmissions are corrupted and require re-transmission based on patterns learned from recorded historical data.

The drawback of DNNs is that there are many hyperparameters that significantly impact the performance of the network. These hyperparameters are problem specific and dataset specific and are computationally expensive to optimize. Since DNNs are significantly larger than other methods like SVMs, they require a much larger amount of data and much more training time. In many situations, large quantities of labeled training data can be expensive to obtain. In addition, these models are less explainable and in many applications are used as blackbox models, which makes them less attractive to scientists. In recent years, the machine learning community has attempted to make deep models more explainable with works such as the XAI (eXplainable Artificial Intelligence) program by DARPA [44].

A simple feed-forward DNN model (FF-DNN) to perform binary classification on our dataset can be cast as

$$\hat{y} = f_\theta(x)$$

where x is the input features, y is the label, θ are hyperparameters of the model f_θ representing the result of passing the input x through the layers of the DNN. Each layer l of the network contains neurons connected to all the neurons of the next layer through weighted arcs. The input of the next layer is the result of the linear combination of the current layer and said weights. This combination goes through an activation function $\phi(.)$ to introduce a non linearity. For more detailed discussions on DNNs, see [31].

We use a weighted binary cross-entropy loss function

$$L(y,\hat{y}) = -\sum_i \beta_0\,(1+\hat{y}_i)\,\log(1+y_i) + \beta_1\,(1-\hat{y}_i)\,\log(1-y_i), \tag{2}$$

where β_0 and β_1 are the weights propulsion to the number of passes in complete pass , considered as 1, or incomplete pass that is considered as -1. This loss function helps us to reduce the effect of the class imbalance.

However, the performance of a FF-DNN can be effected by the its architecture in a specific problem and is dependent upon the choice of a set of hyperparameters (HPs), e.g., the architecture of the deep learning network, forms of activation functions, regularization coefficients, and the choice of the optimizer. In practice, these hyperparameters are usually selected with grid search or from prior experiments on similar problems. The selection and initial setting of these hyperparameters critically impacts the performance of deep learning networks in terms of quality of solution and training time required [32]. To compute the network hyperparameters for a FF-DNN with five hidden layers, we can solve a blackbox optimization problem that minimizes a possibly nonconvex function, defined as with the feasible domain in a hyperparameter space that is a simple convex region bounded by linear inequality constraints [33–35]; In this study, this task is carried out by the Δ-MADS algorithm described in the following section.

Brief overview of Δ-MADS method

The HPO problem can be framed as a blackbox one. The blackbox represents the code that takes the hyperparameters as input, builds, trains and tests the corresponding network and returns as measure of performance as the objective value to be minimized. By doing so, derivative-free optimization methods can be applied. The Δ-MADS algorithm [38] is a combination of two DFO schemes: the local refinement of the MADS algorithm [37] and the global search of Δ-DOGS [36]. At each iteration k, MADS defines a mesh

[4] Neural Networks models with more than five hidden layers are considered as deep models.

M_k around the current incumbent x_k and evaluates a set of candidates $p_i^k \in M_k$ chosen along search directions that defines a positive basis. MADS allows to plug in a global search strategy without losing its convergence properties as long as 1) every sampled point is projected on the mesh and and 2) each iteration is guaranteed to evaluate a finite number of candidates. Delaunay-based derivative-free optimization via global surrogates [36], dubbed Δ-DOGS, is a generalized family of surrogate based optimization algorithms that decouples the task uncertainty model of surrogate from the surrogate model unlike most of the surrogate based optimization schemes such as Bayesian optimization. Δ-DOGS is globally provably convergent for the optimization problems where the Lipschitz bound of the objective function is bounded [33]. It was shown that for the even the problems where these assumptions are not true that Δ-DOGS can find an optimal solution. In the paper [38], the authors show that Δ-MADS finds efficient DNNs quicker than other optimization schemes, including its two components individually. It is therefore the algorithm selected to carry out the HPO of the DNN considered in this study.

5 RESULTS

In this section, we evaluate some supervised learning schemes' performance and the DNN approach of Section 4 on the dataset described in Section 3. A naive measure of a classifier's performance is accuracy, which is the number of times the classifier correctly predicted the label of a pass divided by the total number of passes. Since our complete and incomplete passes are not balanced, to better understand classifiers' performance, the overall performance is reported using precision, recall, F1-score, and accuracy for both incomplete and complete passes.

Understanding the behavior of DNN

To understand the behavior of the DNN, we visualized a pattern of data in one of the hidden layers with 16 dimension. We used t-SNE [48] to project this into two dimensions to visualize, as shown in Figure 3. t-SNE model with a perplexity of 30, a learning rate of 200, and 1000 iterations was used.

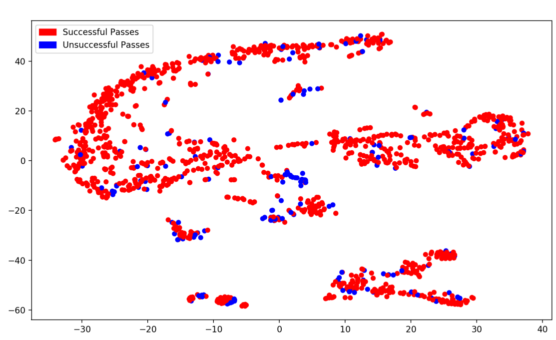

Figure 3. Latent space projection of middle hidden layer in DNN on a sample data with t-SNE. The red points represent complete passes and the blue points represent incomplete passes [49].

Figure 3 illustrates that the complete passes can be characterized into three different classes. We found that these three classes are associated with different data resources such as transmission of data from different orbiters (TGO, ODY, MRO, MVN) to DSN and other types of transfer in the GDS data described in [49].

Using this observation, the metadata available for this study was split into three separate classes of TGO, ODY, and MRO and evaluated separately. The FF-DNN in this situation achieved higher performance compared to using all classes. The recall on the incomplete passes in this situation was 0.78 versa the recall of 0.52 in the one class model, the recall on the complete passes achieved 0.99, the precision score on incomplete passes 0.85, the precision score on complete passes 0.99, the F1-score score on incomplete passes 0.90, the F1-score score on complete passes 0.99. On the test dataset, the model achieved 0.96 accuracy score. When splitting the test results by orbiter, we saw that there was an accuracy of 0.96 on MROs, 0.98 on TGO, and 0.96 on ODY. The overall accuracy of the previous FF-DNN was 0.91 accuracy similar to the GDSA. But, the t-SNE visualization helped us to split the data into different groups and evaluate each model separately. The performance of FF-DNN could improve as GDS collects more data.

Machine learning-based labeler

We evaluate several supervised machine learning methods for improving the performance in ADAM. Each datapoint includes categorical one-hot vectors that consists of 42 features, where 24 of those are categorical features and 18 are continuous features representing the differences in data loss in each communication segment. All the features in the dataset are normalized with z-score normalization [49]. A detailed discussion of input features for each methodology can be found in [49].

Table 2 shows the performance of the machine learning approaches that perform better compared to the existing GDSA Dashboard labeler shown in Table 1. To study the performance of the machine learning based classifier, we first split the data to 90% of it as a training set (8592 passes) and 10% of it as a test set (955 passes). We used a stratified split to take into account the class imbalance. The training data is further split into training data and validation data, using 10-fold cross validation, to find the best hyperparameters for each classifier using a grid search and derivative-free optimization methods [33, 35]. Then the classifiers are evaluated using the 10% test data that is withheld.

The FF-DNN architecture consists of five dense layers with a dropout rate of 0.1 between each of the first four layers. Each hidden layers used ReLU [27] as its activation function, and sigmoid was applied on the output layer to compute the probability of success. In order to find the architecture of the network we used the hyperparameter optimization technique described in [35]. We change the number of nodes in each layer and then we tuned the dropout rate and learning rate. Our observation was the network in this application was more sensitive to the dropout rate and learning rate compared with the number of nodes in each layer. The network was trained for 400 epochs with a batch size of 20, with data randomly shuffled between each epoch. A learning rate of 0.0042 and SGD were used to update the model's weights.

The performance of different methods is compared in Table 2 in order to identify complete and incomplete downlink passes for MSL data from Sol 337 to Sol 2747. The training data includes 9547 datapoints, which 1260 are incomplete, and 8287 are complete passes. We used 118 incomplete and

Table 2. Comparison of different approaches' accuracy on the MSL dataset. 1: Complete pass and -1: Incomplete pass. Avg.: Average performance, and W/Avg.: Weighted Average performance based on the number of labels in each class. Algorithms are LR: Logistic Regression classifier, SVM: Support Vector Machine, GNB: Gaussian Naive Bayes, DNN: Deep Neural Network, HPO-DNN: optimizing the hyperparameters of DNN with [38],

Metric	Precision				Recall				F1-score				Accuracy
Alg.	1	-1	Avg.	W/Avg.	1	-1	Avg.	W/Avg.	1	-1	Avg.	W/Avg.	Score
LR	0.92	**0.91**	**0.92**	**0.92**	**0.99**	0.38	0.69	**0.92**	**0.95**	0.53	0.74	**0.90**	**0.92**
SVM	0.95	0.43	0.69	0.88	0.87	0.68	0.77	0.85	0.91	0.53	0.72	0.86	0.85
GNB	0.90	0.47	0.69	0.61	0.96	0.27	0.61	0.87	0.93	0.34	0.64	0.85	0.87
FF-DNN	0.93	0.69	0.81	0.74	0.97	0.52	0.74	0.91	**0.95**	0.59	0.77	**0.90**	0.91
HPO-DNN [38]	**0.97**	0.78	0.87	0.91	0.90	**0.82**	**0.86**	0.89	0.93	**0.80**	**0.87**	**0.90**	**0.92**

837 complete passes to validate our results, which in the test set size is 955. Each method's results are averaged to be comparable with the GDSA analyst reports summarized in Table 1.

Discussions

In all cases, the supervised machine learning labelers outperform the existing GDSA dashboard labeler. The lack of explainability for machine learning based methods is the main concern of GDSA in using these approaches. We can see in the Table 2 approaches that are explainable and relatively simple such as Logistic regression (LR) and Gaussian Naive Bayes (GNB) [45] have even lower recall score for the incomplete downlink passes compared to the GDSAs. However, such a schemes report high accuracy score which can be misleading.

As we saw in Table 1, the HPO-DNN has 92% accuracy and in general outperforms the other approaches except the Logistic Regression (LR). The recall of the HPO-DNN classifier on incomplete classes is 82%, which dramatically improves upon the 56% recall of the GDSA Dashboard Labeler on incomplete passes. We used the implementation available in [45] to validate our results.

We can see in Table 2 that LR has higher recall performance on complete passes compared to other approaches, but it is less reliable for the incomplete passes. Although approaches such as LR, SVB, and GNB labelers are more explainable compared to DNNs, the overall performance of these methods are lower than DNN.

Overall, when comparing these values to the precision, recall, and F1-score in Table 1; in most cases, we can see a significant improvement on the current GDSA dashboard labeler using supervised machine learning labelers.

The results presented in this work and our previous related studies [38,49] convinced GDS for MSL to leverage machine learning-based methods in the operation of MSL. Our framework can provide explanations on "why the data is missing" to the GDSA by sensitivity analysis about which features contributed the most to making a downlink unsuccessful [38,49].

6 Conclusion and Future Work

This work demonstrated various supervised machine learning algorithms' performance to classify a downlink as complete or incomplete that can be used in ADAM for Mars Rovers. This work introduced a framework for machine learning-based labelers that can assist the GDSA analysts in their work. We implemented this work in the GDS MSL software to investigate data volume reduction in MSL daily operations. Our work simplified and partially automated the labor-intensive and manual task of a GDSA analyst; thus, a new GDSA member can respond to the Ground Data System's issues with shorter training time.

In future works, we will include identifying more data sources in the downlink process and identifying the presence of an issue and the location where the issue occurs. In addition, we will experiment with generative neural network models to increase the robustness our data volume detectors for the operation of MSL and other flight missions.

Acknowledgments

The authors also gratefully acknowledge partial funding from Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (NASA) in support of this work. The authors would like to thank Miguel Pena, Ray Han, Ashton Vaughs, and Adans Ko, the MSL Real-Time Operations Team Chief. Also authors thank P. Shokouhi, A. Barrett, D. Hanks, R. Castano, and F. Y. Hadaegh.

References

[1] Jet Propulsion Laboratory. 2019. Strategic Technologies 2019. Jet Propulsion Laboratory, California Institute of Technology (Pasadena, CA)

[2] Jet Propulsion Laboratory. 2020. Technology Highlights. Jet Propulsion Laboratory, California Institute of Technology (Pasadena, CA)

[3] Fong, T. W., Frank, J. D., Badger, J. M., Nesnas, I. A., & Feary, M. S. (2018). Autonomous systems taxonomy.

[4] Dehghani, N., & Tankenson, M. (2006). A multi-mission event-driven component-based system for support of flight software development, ATLO, and operations first used by the Mars Science Laboratory (MSL) project.

[5] Johnston, M., & Lad, J. (2018). Integrated Planning and Scheduling for NASA's Deep Space Networkâ from Forecasting to Real-time. In 2018 SpaceOps Conference (p. 2728).

[6] Guzewich, S. D., Lemmon, M., Smith, C. L., MartÃnez, G., de Vicente Retortillo, Ã ., Newman, C. E., ... & Harri, A. M. (2019). Mars science laboratory observations of the 2018/mars year 34 global dust storm. Geophysical Research Letters, 46(1), 71-79.

[7] Kerner, H. R., Wellington, D. F., Wagstaff, K. L., Bell, J. F., Kwan, C., & Amor, H. B. (2019, July). Novelty detection for multispectral images with application to planetary exploration. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 33, pp. 9484-9491).

[8] Deng, L., & Yu, D. (2014). Deep learning: methods and applications. Foundations and TrendsÂ® in Signal Processing, 7(34), 197-387.

[9] Fazlyab, M., Morari, M., & Pappas, G. J. (2019). Safety verification and robustness analysis of neural networks via quadratic constraints and semidefinite programming. arXiv preprint arXiv:1903.01287.

[10] Arabshahi, F., Singh, S., & Anandkumar, A. (2018). Combining Symbolic Expressions and Black-box Function Evaluations in Neural Programs. arXiv preprint arXiv:1801.04342.

[11] Jordan, M. I. (2019). Artificial intelligence The revolution hasn't happened yet. Harvard Data Science Review.

[12] Noury, K., Yang, B. (2019). Class of Stabilizing Parallel Feedforward Compensators for Nonminimum-Phase Systems. In 2019 Proceedings of the ASME Conference on DSCC. ASME.

[13] Bue, B. D., Wagstaff, K. L., Rebbapragada, U. D., Thompson, D. R., & Tang, B. (2014). Astronomical data triage for rapid science return. In Proc. 2014 Conf. on âBig Data from Space (BiDSâ14) (Vol. 206).

[14] Bayer, T. J. (2009, March). Mars Reconnaissance Orbiter in-flight anomalies and lessons learned: An update. In 2009 IEEE Aerospace Conference (pp. 1-11). IEEE.

[15] Chamberlain, N., Gladden, R., Barela, P., Epp, L., & Bruvold, K. (2015, March). MAVEN relay operations. In 2015 IEEE Aerospace Conference (pp. 1-12). IEEE.

[16] James, M. L., Shapiro, A. A., Springer, P. L., & Zima, H. P. (2009). Adaptive fault tolerance for scalable cluster computing in space. The International Journal of High Performance Computing Applications, 23(3), 227-241.

[17] Kolcio, K., Fesq, L., & Mackey, R. (2017, March). Model-based approach to rover health assessment for increased productivity. In 2017 IEEE Aerospace Conference (pp. 1-13). IEEE.

[18] Kerber, L., Dickson, J. L., Head, J. W., & Grosfils, E. B. (2017). Polygonal ridge networks on Mars: Diversity of morphologies and the special case of the Eastern Medusae Fossae Formation. Icarus, 281, 200-219.

[19] Thompson, D. R., Babu, K. N., Braverman, A. J., Eastwood, M. L., Green, R. O., Hobbs, J. M., ... & Mathur, A. (2019). Optimal estimation of spectral surface reflectance in challenging atmospheres. Remote Sensing of Environment, 232, 111258.

[20] Lee, H., Kalashnikova, O. V., Suzuki, K., Braverman, A., Garay, M. J., & Kahn, R. A. (2016). Climatology of the aerosol optical depth by components from the Multi-angle Imaging SpectroRadiometer (MISR) and chemistry transport models. Atmospheric Chemistry and Physics, 16(10), 6627-6640.

[21] Mandrake, L., Wagstaff, K. L., Gleeson, D., Rebbapragada, U., Tran, D., CastaÃ±o, R., ... & Pappalardo, R. T. (2009, March). Onboard detection of natural sulfur on a glacier via a SVM and hyperion data. In 2009 IEEE Aerospace conference (pp. 1-15). IEEE.

[22] Noury, K., Yang, B. (2019). Tip Position Control of Single Flexible Links via Parallel Feedforward Compensation. In 2019 Proceedings of the ASME Conference on IDETC/CIE. ASME.

[23] Friedman, J., Hastie, T., & Tibshirani, R. (2001). The elements of statistical learning (Vol. 1, No. 10). New York: Springer series in statistics.

[24] Gladden, R. E. (2014). Mars Relay Operations Service (MaROS): a present service preparing for the future. In SpaceOps 2014 Conference (p. 1916).

[25] Lipton, Z. C. (2016). The mythos of model interpretability. arXiv preprint arXiv:1606.03490.

[26] Ko, A., Maldague, P., Lam, D., Bui, T., & McKinney, J. (2010, April). The Evolvable Advanced Multi-Mission Operations System (AMMOS): Making Systems Interoperable. In SpaceOps 2010 Conference Delivering on the Dream Hosted by NASA Marshall Space Flight Center and Organized by AIAA (p. 2303).

[27] Davidson, G., & Mozer, M. C. (2019). Sequential mastery of multiple tasks: Networks naturally learn to learn. arXiv preprint arXiv:1905.10837.

[28] Bach, S., Binder, A., Montavon, G., Klauschen, F., MÃŒller, K. R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one, 10(7), e0130140.

[29] Chien, S., Doubleday, J., Thompson, D. R., Wagstaff, K. L., Bellardo, J., Francis, C., ... & Piug-Suari, J. (2016). Onboard autonomy on the intelligent payload experiment cubesat mission. Journal of Aerospace Information Systems, 307-315.

[30] Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

[31] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT press.

[32] Ash, J. T., & Adams, R. P. (2019). On the Difficulty of Warm-Starting Neural Network Training. arXiv preprint arXiv:1910.08475.

[33] Alimo, S. R., Beyhaghi, P., & Bewley, T. R. (2017, December). Optimization combining derivative-free global exploration with derivative-based local refinement. In 2017 IEEE 56th Annual Conference on Decision and Control (CDC) (pp. 2531-2538). IEEE.

[34] Beyhaghi, P., Alimo, R., & Bewley, T. (2019). A derivative-free optimization algorithm for the efficient minimization of functions obtained via statistical averaging. arXiv preprint arXiv:1910.12393.

[35] Lakhmiri, D., Digabel, S. L., & Tribes, C. (2019). HyperNOMAD: Hyperparameter optimization of deep neural networks using mesh adaptive direct search. arXiv preprint arXiv:1907.01698.

[36] Alimo, R., Beyhaghi, P., Bewley, T. R. (2020). Delaunay-based derivative-free optimization via global surrogates. Part III: nonconvex constraints. Journal of Global Optimization, 1-34.

[37] Audet, C., Dennis Jr, J. E. (2006). Mesh adaptive direct search algorithms for constrained optimization. SIAM Journal on optimization, 17(1), 188-217.

[38] Lakhmiri, D., Alimo, R., Digabel, S. L. (2020). Tuning a variational autoencoder for data accountability problem in the Mars Science Laboratory ground data system. arXiv preprint arXiv:2006.03962.

[39] SchÃ¶lkopf, B., Williamson, R. C., Smola, A. J., Shawe-Taylor, J., & Platt, J. C. (2000). Support vector method for novelty detection. In Advances in neural information processing systems (pp. 582-588).

[40] Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine learning, 20(3), 273-297.

[41] Cherkassky, V., & Ma, Y. (2004). Practical selection of SVM parameters and noise estimation for SVM regression. Neural networks, 17(1), 113-126.

[42] Alimo, S. R., Beyhaghi, P., & Bewley, T. R. (2019). Delaunay-Based Global Optimization in Nonconvex Domains Defined by Hidden Constraints. In Evolutionary and Deterministic Methods for Design Optimization and Control With Applications to Industrial and Societal Problems (pp. 261-271). Springer, Cham.

[43] Weir, A. (2014). The martian. Random House.

[44] Gunning, D., & Aha, D. W. (2019). DARPA's Explainable Artificial Intelligence Program. AI Magazine, 40(2), 44-58.

[45] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. Journal of machine learning research, 12(Oct), 2825-2830.

[46] Dorogush, A. V., Ershov, V., & Gulin, A. (2018). CatBoost: gradient boosting with categorical features support. arXiv preprint arXiv:1810.11363.

[47] Sarkarati, M., Spada, M., Moulin, S., & Fischer, D. (2010). Leveraging advanced software technologies for implementing the european space situational awareness ground data systems. In SpaceOps 2010 Conference Delivering on the Dream Hosted by NASA Marshall Space Flight Center and Organized by AIAA (p. 1922).

[48] Maaten, L. V. D., & Hinton, G. (2008). Visualizing data using t-SNE. Journal of machine learning research, 9(Nov), 2579-2605.

[49] Kahovec, B. et al. (2019). Deep Learning and Anomaly Detection in Mars Rover Data. NTR-51336. JPL's Internal Report.

BIOGRAPHY

Ryan Alimo *is a data scientist working in machine learning at the Jet Propulsion Laboratory, California Institute of Technology. Dr. Alimo is advancing the field of human-AI and multi-agent systems using integration of the optimization algorithms with deep learning and deploying them to spacecraft formation flying, deep-space communication, and data accountability for the Mars data.*

Dylan Sam *is a senior at Brown University studying Mathematics and Computer Science. His research interests are in weakly supervised learning, semi-supervised learning, and statistical/theoretical machine learning. He interned at NASA JPL during the summer of 2019, working on deep learning and anomaly detection.*

Dounia Lakhmiri *is a PhD candidate at Polytechnique Montreal from the departement of applied Mathematics. Her research is focused on applying derivative-free optimization methods for the hyperparameter optimization problem of deep neural networks.*

Brian Kahovec *is the Mars Science Laboratory Real-Time Operations Deputy Team Chief. At NASA Jet Propulsion Laboratory, he has applied machine learning technology to a variety of problems including cybersecurity and data accountability.*

Dariush Divsalar *received the Ph.D. degree in EE from UCLA. He is a JPL Fellow. He has been involved with developing state-of-the-art technology for advanced deep-space communications systems and future NASA space exploration. He has published more than 275 papers, coauthored a book, and contributed to three other books, and holds 25 U.S. patents. Dr. Divsalar is a co-recipient of several IEEE best paper awards. He has received a NASA Exceptional Engineering Achievement Medal. He is IEEE fellow and was an Editor for the IEEE Transactions on communications.*