

# Hybrid Decision Architectures for Space Telemetry: Enhancing Mission Reliability through Ensemble Learning

C. Vinitha<sup>1\*</sup>, Jeedimadla Pranitha<sup>2</sup>, Sapavath Rahul Nayak<sup>2</sup>, Majjiga Saketh Reddy<sup>2</sup>

<sup>1</sup>Assistant Professor, <sup>2</sup>UG Student, <sup>1,2</sup>Department of Computer Science and Engineering

<sup>1,2</sup>Kommuri Pratap Reddy Institute of Technology, Ghanpur, Ghatkesar, 501301, Telangana, India.

\*Correspondence: C. Vinitha ([vnreddy61@gmail.com](mailto:vnreddy61@gmail.com))

## ABSTRACT

The increasing dependence on space-based systems, supported by thousands of active satellites and significant economic value, has made operational reliability a critical concern in aerospace applications. A large proportion of mission failures can be attributed to unnoticed irregularities in telemetry streams, creating a strong need for intelligent systems capable of accurate prediction and early anomaly detection. Telemetry data provides detailed insights into spacecraft subsystems and operational conditions; however, its high dimensionality, presence of noise, missing values, and complex variability make analysis challenging for traditional techniques. To address these issues, this study introduces a scalable framework for telemetry analysis using a locally implemented ensemble learning approach based on OPSSATAD data. The system integrates multiple learning models within a unified structure inspired by classification and regression tree (CART) principles, including Random Forest (RF-CART), Support Vector Machine (SVM-CART), and Gradient Boosting (GB-CART) for both classification and prediction tasks. Furthermore, a hybrid ensemble model combining Extra Trees (ET-CART) and CatBoost (CatBoost-CART) is proposed using a voting-based aggregation mechanism to enhance predictive performance. The ET component captures hierarchical decision patterns, while CatBoost improves learning through efficient handling of feature interactions and categorical data. A structured pipeline involving preprocessing, feature refinement, training, and evaluation is implemented using performance metrics such as accuracy, precision, recall, F1-score, MAE, MSE, RMSE, and R<sup>2</sup>. The results indicate that the ensemble approach consistently achieves higher accuracy and robustness compared to individual models, making it effective for reliable mission prediction and analysis.

**Keywords:** Satellite telemetry, Mission prediction, CART (Classification and Regression Tree), Ensemble learning, OPSSATAD, Aerospace reliability.

## 1. INTRODUCTION

The field of remote-sensing satellite systems is experiencing rapid growth, driven by the deployment of large-scale constellations and heterogeneous sensor networks that significantly expand global observation capabilities. As these networks evolve, satellite ground stations must simultaneously manage continuous telemetry reception and command transmission, increasing the complexity of coordinated operations. Satellite mission scheduling, which involves allocating tasks across multiple satellites and ground infrastructures, has therefore become a critical component in ensuring efficient system performance. Traditional static and semi-static scheduling mechanisms are increasingly inadequate, as they fail to respond to the highly dynamic nature of modern satellite environments. In contrast, dynamic priority-based scheduling frameworks provide greater flexibility by continuously adapting to variations in mission

requirements [1] and fluctuating resource constraints such as communication windows, energy availability, and sensor limitations [2]. These adaptive systems not only improve execution efficiency but also reduce latency and enhance responsiveness to time-sensitive tasks.

However, the growing demand for high-resolution data, real-time monitoring, and global coverage introduces several challenges. Satellite resources remain limited and expensive, requiring intelligent allocation strategies that can balance competing mission priorities while maximizing utilization. Inflexible scheduling can lead to resource underutilization, increased delays, and degraded mission outcomes. Additionally, ensuring accurate execution of high-priority tasks under strict temporal and operational constraints requires robust and scalable optimization algorithms. The presence of uncertainties, such as environmental variations, orbital dynamics, and communication disruptions, further complicates scheduling decisions.

To address these issues, recent research has emphasized the development of dynamic mission scheduling frameworks capable of handling time-varying conditions and complex system interactions [3]. These approaches often incorporate adaptive decision-making techniques to respond to changes in satellite availability and mission demands in real time. Furthermore, advanced resource allocation strategies have been proposed to optimize the distribution and utilization of satellite network resources [4], enabling more efficient, resilient, and intelligent mission planning for next-generation remote-sensing applications.

## **2. Related Work**

Anomaly detection in satellite telemetry has evolved significantly from traditional monitoring techniques to advanced machine learning and graph-based approaches. Early methods primarily relied on threshold-based and statistical techniques, which were simple to implement but limited in handling complex, nonlinear, and high-dimensional telemetry data. Recent advancements focus on hybrid, adaptive, and data-driven models that improve detection accuracy while addressing real-time operational challenges.

### **2.1 Threshold-Based and Statistical Methods**

Threshold-based approaches form the foundation of early telemetry anomaly detection systems. Heras and Donati et al. [6] implemented an automated monitoring system deployed in real satellite missions under the European Space Agency, where dynamic thresholds were used to monitor thousands of telemetry parameters during missions such as XMM-Newton. This approach reduced manual supervision and improved operational efficiency.

To overcome the limitations of static thresholds, more advanced statistical techniques were introduced. Zeng et al. [7] proposed a method combining parametric causality with Double-Criteria Drift Streaming Peaks Over Threshold (DCDSPOT), enabling adaptive threshold updates based on data drift and peak analysis. While this improves sensitivity to subtle anomalies and reduces false negatives, it requires careful parameter tuning and can be sensitive to noise in dynamic environments.

### **2.2 Machine Learning and Hybrid Approaches**

With the growing complexity of telemetry data, machine learning-based methods have been widely adopted. Vos et al. [5] introduced a hybrid framework combining Long Short-Term Memory (LSTM)

networks with One-Class Support Vector Machines (OCSVM). The LSTM captures temporal dependencies and generates prediction errors, which are then used by OCSVM for anomaly detection. This two-stage model enhances detection of subtle and temporal anomalies but increases computational complexity, limiting its suitability for real-time applications.

Similarly, Ramachandran et al. [11] proposed the ADTM framework, integrating Artificial Neural Networks with an Extra Tree Classifier to extract topological features from telemetry data. This approach effectively detects both known and unknown anomalies; however, the hybrid architecture results in increased computational overhead.

### **2.3 Clustering-Based and Unsupervised Techniques**

Unsupervised learning methods have gained importance due to the scarcity of labeled anomaly data. Gao et al. [8] proposed a clustering-based approach that models normal system behavior using grouping patterns derived from telemetry data, allowing anomalies to be identified as deviations from these clusters. Jin et al. [9] further enhanced this concept using dominant sets clustering, a graph-based technique that identifies cohesive groups for improved anomaly separation.

Although these methods eliminate the need for labeled datasets, their performance may degrade in the presence of overlapping clusters or rapidly changing system conditions. Additionally, graph-based clustering methods may face scalability issues for large datasets.

### **2.4 Adaptive and Real-Time Detection Methods**

To address real-time requirements, adaptive and lightweight models have been developed. Bernard et al. [10] introduced the ELMER method, which uses neural networks to dynamically adjust threshold boundaries, reducing false alarms and improving detection speed.

Baireddy et al. [12] proposed Extreme Learning Machines (ELMs) for real-time anomaly detection, offering fast training and low computational cost, making them suitable for streaming telemetry data. However, these methods can be sensitive to noise and require careful parameter initialization for stable performance.

### **2.5 Graph-Based and Multivariate Models**

Recent research focuses on capturing complex relationships among multiple telemetry variables. Song et al. [13] proposed a Graph Convolutional Network (GCN)-based approach that models telemetry variables as interconnected nodes, incorporating attention mechanisms to emphasize important relationships. This enables effective detection of correlation-based anomalies that traditional methods often fail to identify. Despite their improved accuracy, graph-based approaches introduce challenges such as high computational overhead and the need for careful graph construction, particularly in large-scale or real-time systems.

## **3. PROPOSED METHODOLOGY**

The system architecture is designed as a scalable and unified ML framework that integrates user management, data processing, model training, and prediction within a Flask-based environment. To address limitations in traditional standalone models, this system introduces a structured pipeline for telemetry-style

data analysis using a locally implemented ensemble learning approach. The workflow begins with secure user authentication and controlled access, followed by data management operations including loading, cleaning, feature selection, and dataset splitting. EDA and preprocessing steps ensure high-quality input by analyzing distributions, correlations, and feature relevance before model training. The architecture incorporates multiple models inspired by CART principles, including RF-CART, SVM-CART, and GB-CART, enabling both classification and prediction tasks within a unified framework.

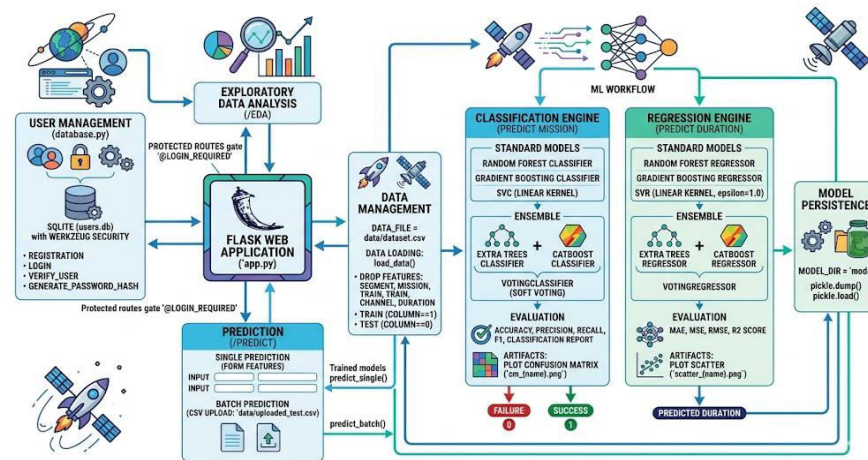


Fig. 1: System architecture of the proposed satellite mission prediction framework.

In addition, a hybrid ensemble model combining ET-CART and CatBoost-CART is implemented using a voting-based mechanism to enhance predictive performance, where ET captures hierarchical decision structures and CatBoost improves learning through efficient handling of complex feature interactions. The models are evaluated and compared using metrics such as accuracy, precision, recall, F1-score for classification, and MAE, MSE, RMSE, and  $R^2$  for regression to identify optimal performance. The trained models are persisted using pickle for reuse, ensuring efficient deployment. In the prediction phase, users interact through the Flask interface to perform single or batch predictions, where input data is processed through the same pipeline and passed to the trained ensemble models as illustrated in Fig. 1. Finally, the system generates classification outputs and predicted values, delivering a robust, secure, and end-to-end intelligent prediction framework.

**1. Data Collection and Input Handling:** The system begins by collecting satellite telemetry data from the OPSSATAD dataset, which contains various operational parameters related to mission performance. This data is stored in a structured CSV format and serves as the primary input for both classification and regression tasks.

**2. Feature Extraction and Selection:** Relevant features are selected from the telemetry data to represent satellite behavior effectively. These features are used as inputs for machine learning models, ensuring that only significant attributes contribute to prediction performance.

**3. Implementation of Classification Models (Including CART):** The system applies multiple classification algorithms such as Random Forest, Gradient Boosting, and Support Vector Machine (SVM). Tree-based models like Random Forest and Gradient Boosting are based on CART, which helps in learning decision rules for mission success classification.

**4. Implementation of Regression Models (Including CART):** For predicting mission duration, regression models such as Random Forest Regressor, Gradient Boosting Regressor, and Support Vector Regressor (SVR) are used. CART-based regression techniques are utilized in tree-based models to capture nonlinear relationships in telemetry data.

**5. Proposed Ensemble Model Construction:** In addition to individual models, a proposed ensemble model is developed using Extra Trees and CatBoost algorithms. These models are combined using a Voting mechanism to enhance prediction accuracy and reduce variance compared to single models.

**6. Model Training and Storage:** All classification and regression models, including CART-based and ensemble models, are trained using the training dataset. The trained models are then saved using serialization techniques to enable reuse without retraining.

**7. Model Evaluation and Performance Analysis:** The system evaluates model performance using metrics such as accuracy, precision, recall, and F1-score for classification, and MAE, MSE, RMSE, and  $R^2$  score for regression. Visualization techniques like confusion matrices and scatter plots are used for analysis.

**8. Prediction Module (Single and Batch Processing):** The system provides prediction functionality where users can input individual telemetry values or upload batch CSV files. The trained models generate outputs for mission success classification and mission duration prediction.

**9. Web-Based Deployment and User Interaction:** The complete system is deployed using a Flask web application that includes user authentication, data visualization, model comparison, and prediction interfaces. This ensures easy accessibility and interaction for users.

### Proposed ensemble model

The proposed system implements a locally stored ensemble learning model for satellite mission success classification and duration prediction by integrating CART-based ET and CatBoost models using a Voting mechanism. The Extra Trees model, built on the principles of CART, constructs multiple randomized decision trees to capture hierarchical decision patterns and nonlinear relationships in telemetry data. Alongside this, CatBoost enhances prediction capability through gradient boosting, effectively modeling complex feature interactions. The ensemble combines these models using soft voting for classification and average aggregation for regression, resulting in improved accuracy, robustness, and stability. Additionally, all trained models are stored locally, enabling efficient reuse and real-time prediction within the Flask-based system as illustrated in fig 2.

**1. Telemetry Data Acquisition and Segregation:** The system begins by loading satellite telemetry data from a CSV file, where records are pre-labeled with a train flag to distinguish training and testing datasets. The dataset includes mission status and duration along with multiple telemetry features. This structured segregation eliminates the need for dynamic splitting and ensures consistency across experiments.

**2. Feature Selection and CART-Based Structuring:** Irrelevant attributes such as segment, channel, and identifiers are removed to retain only meaningful telemetry inputs. The Extra Trees model (CART-based) internally constructs multiple randomized decision trees using selected features and splitting rules based on impurity measures. This enables the system to capture nonlinear relationships and hierarchical decision boundaries in mission success prediction.

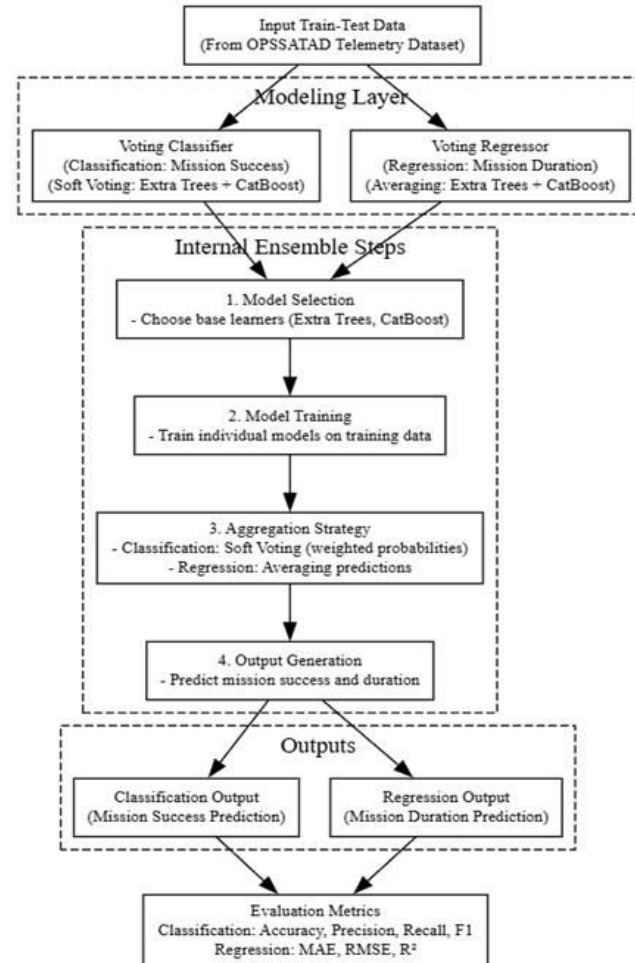


Fig 2: Internal workflow of proposed ensemble model

**3. Training of CART-Based Extra Trees Model:** The Extra Trees classifier and regressor are trained on the processed telemetry dataset using multiple randomized trees. Unlike traditional decision trees, it introduces randomness in feature selection and split thresholds, reducing variance and preventing overfitting. This forms the CART backbone of the proposed ensemble system, providing fast and diverse predictions.

**4. Training of CatBoost Model for Gradient Optimization:** Parallel to Extra Trees, the CatBoost model is trained using gradient boosting over decision trees, handling feature interactions effectively. It improves predictive performance by sequentially minimizing errors and capturing complex dependencies in telemetry signals. This complements the CART-based model by adding boosting-based learning capability.

**5. Construction of Voting-Based Ensemble Model:** The trained Extra Trees and CatBoost models are combined using a Voting Classifier (for classification) and Voting Regressor (for regression). In classification, soft voting is applied where probability outputs from both models are averaged to determine final mission success. In regression, outputs are averaged to estimate mission duration.

**6. Model Persistence and Reusability:** Once trained, the ensemble models are saved locally using pickle files. During subsequent executions, the system first checks for existing models and loads them instead of retraining. This significantly reduces computational overhead and ensures faster response time for real-time predictions in the deployed Flask application.

**7. Prediction and Decision Generation:** For new telemetry input, features are processed and passed through the ensemble model. The CART-based Extra Trees contributes structural decision logic, while CatBoost refines predictions through boosting. The final output is a robust aggregated prediction, indicating mission success/failure and estimated duration with improved reliability.

#### 4. Results Description

Figure 3 shows the confusion matrix of the RF (CART) model for mission success prediction shows strong performance in identifying failure cases, with 415 instances correctly classified as failure and only 1 misclassified as success, indicating very high accuracy for the failure class. However, for mission success cases, the model correctly predicts 54 instances while misclassifying 59 as failure, suggesting a bias toward the dominant failure class. This imbalance indicates that while the model is highly reliable in detecting failures, it struggles to equally capture success patterns in telemetry data. The RF (CART) model demonstrates high precision for failure detection but requires improvement in sensitivity toward successful mission classification.

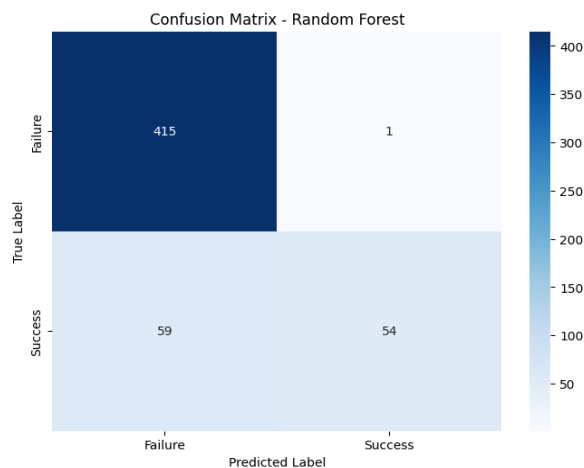


Figure 3: Confusion matrix obtained using RFC for Mission status target

Figure 4 confusion matrix of the GB (CA) model shows that it predicts all instances as failure, correctly classifying 416 failure cases with no errors in that class, but completely failing to identify any success cases, as all 113 success instances are wrongly predicted as failure. This indicates a strong bias toward the majority class and an inability to capture patterns related to mission success in the telemetry data. While the model performs perfectly for failure detection, its inability to classify success makes it ineffective as a standalone model for balanced prediction.

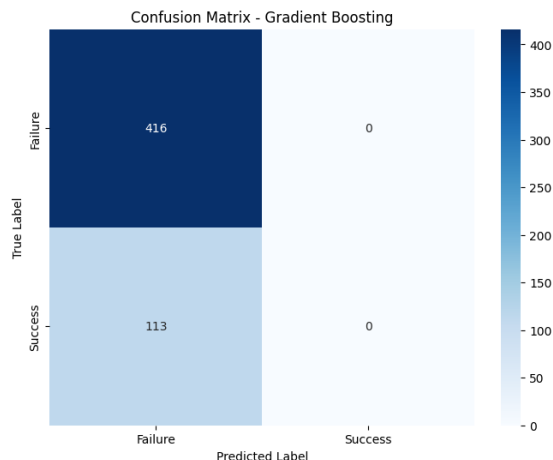


Figure 4: Illustration of confusion matrix using GB (CART) for mission status target

Figure 5 presents confusion matrix of the SVM (CART) model shows that it performs well in identifying failure cases, correctly classifying 412 instances while misclassifying only 4 as success. For mission success cases, it correctly predicts 60 instances and misclassifies 53 as failure, indicating a more balanced performance compared to other models. This reflects the model's ability to distinguish between both classes more effectively, though it still shows a slight bias toward failure prediction. The model achieves a reasonable trade-off between detecting failures and identifying successful missions.

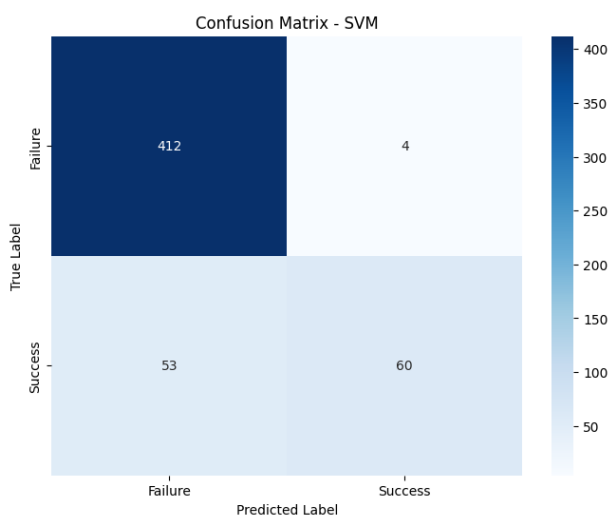


Figure 5: Illustration of confusion matrix using SVM (CART) for mission status target

Figure 9.6 depicts the confusion matrix of the Ensemble (CART) model shows highly balanced and improved performance, correctly classifying 413 failure cases with only 3 misclassified as success, and accurately identifying 103 success cases with just 10 misclassified as failure. This demonstrates a significant improvement over individual models, especially in capturing mission success patterns while maintaining strong failure detection. The combination of CART-based tree learning and boosting

effectively reduces bias and variance, resulting in more reliable and generalized predictions for satellite mission outcomes.

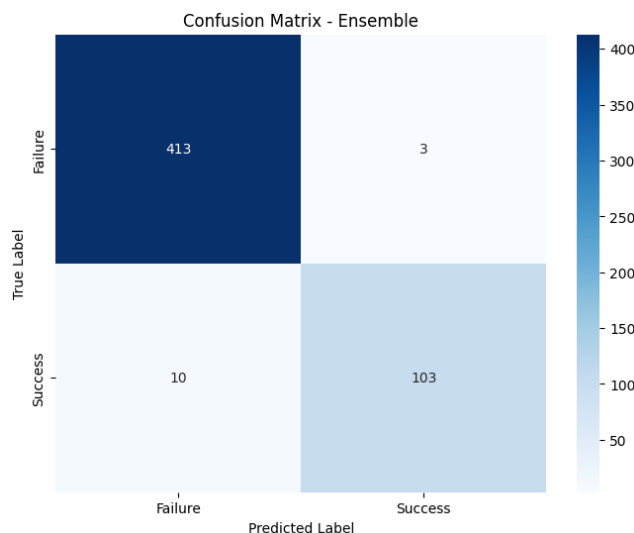


Figure 6: Illustration of confusion matrix using proposed ensemble model for mission status target

Figure 7 shows scatter plot of the RF (CART) model for duration prediction shows a moderate alignment between actual and predicted values, with several data points following the diagonal reference line, indicating reasonable prediction accuracy. However, noticeable clustering and horizontal band patterns suggest that the model tends to produce similar predictions for ranges of input values, reflecting limited sensitivity to finer variations in telemetry data. At higher durations, predictions begin to deviate from the ideal line, indicating underestimation and reduced performance for extreme values. The model captures general trends but lacks precision in accurately predicting continuous duration values across the full range.

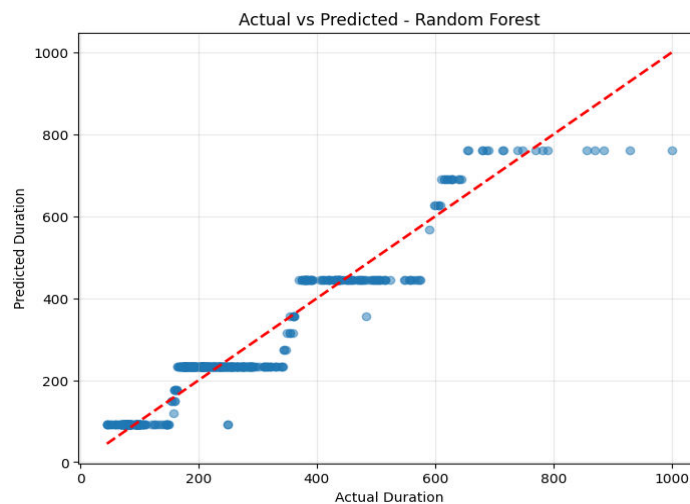


Figure 7: Scatter plot obtained using RF model for duration analysis target

Figure 8 shows scatter plot of the GB (CART) model for duration prediction shows a clear lack of alignment with the ideal diagonal line, indicating poor predictive performance. Most predicted values are concentrated within a narrow range, forming a horizontal pattern regardless of the actual duration, which suggests that the model fails to capture variability in the data. As the actual duration increases, the predictions do not scale accordingly, leading to significant underestimation for higher values. The model is unable to learn meaningful relationships from the telemetry data for accurate duration estimation.

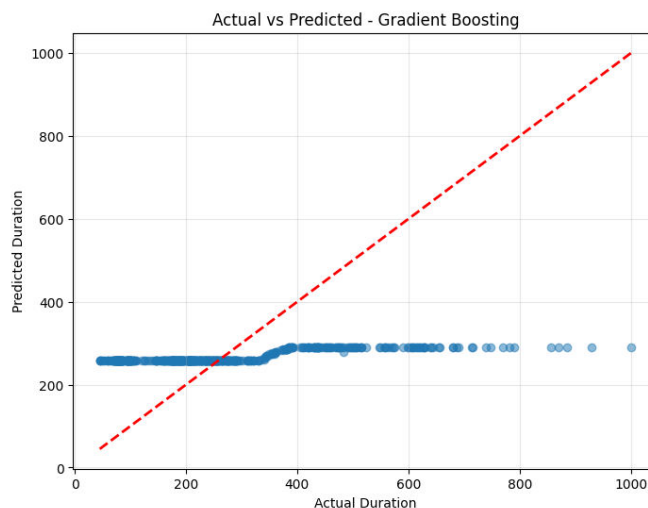


Figure 8: Scatter plot obtained using GB model for duration analysis target

Figure 9 presents the scatter plot of the SVM (CART) model for duration prediction shows a strong linear trend where most data points closely follow a straight pattern, indicating that the model captures the general relationship between actual and predicted values effectively. However, many predictions fall below the ideal diagonal line, suggesting consistent underestimation, especially at higher duration values. Additionally, the presence of a few extreme outliers indicates instability in certain regions of the data. The model demonstrates good trend learning but lacks accuracy in scaling predictions to match actual higher values precisely.

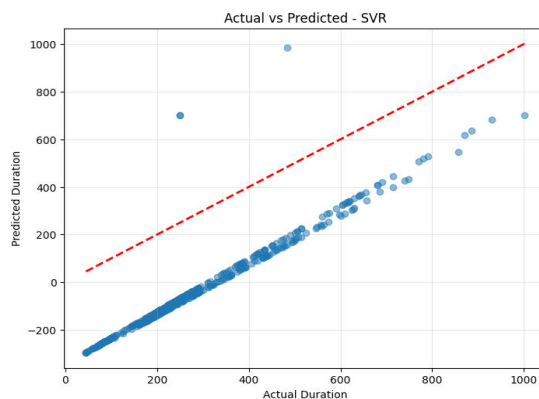


Figure 9: Scatter plot obtained using SVR model for duration analysis target

Figure 10 scatter plot of the Ensemble (CART) model for duration prediction shows an almost perfect alignment of data points along the diagonal reference line, indicating very high prediction accuracy. The predicted values closely match the actual durations across the entire range, demonstrating the model’s strong ability to capture both linear and nonlinear relationships in the telemetry data. Unlike individual models, there is minimal deviation and very few outliers, reflecting improved stability and consistency. The ensemble model provides highly precise and reliable duration predictions by effectively combining multiple learning approaches.

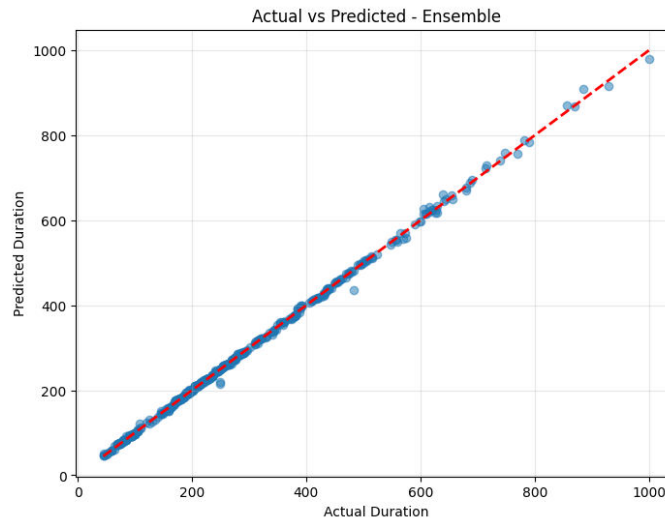


Figure 10: Illustration of scatter plot using proposed ensemble model for duration analysis target

Table 1: Performance comparison of all the classification models for mission status target

Model	Accuracy	Precision	Recall	F1-Score
Random Forest	0.8866	0.8982	0.8866	0.8707
Gradient Boosting	0.7864	0.6184	0.7864	0.6924
SVM	0.8922	0.8970	0.8922	0.8803
Ensemble	0.9754	0.9754	0.9754	0.9751

Table 1 presents the performance comparison of different classification models for mission status prediction, highlighting clear differences in their effectiveness. The RF (CART) model achieves strong performance with high accuracy and precision, indicating its ability to correctly identify mission outcomes, though its F1-score suggests slight limitations in balancing both classes. The GB (CART) model shows comparatively lower performance, with reduced precision and F1-score, reflecting its bias toward the dominant class and weaker capability in distinguishing successful missions. The SVM (CART) model demonstrates balanced and improved results, achieving higher accuracy and better F1-score than the previous models, indicating more reliable classification across both classes. However, the Ensemble (CART + Boosting) model significantly outperforms all individual models, achieving the highest accuracy, precision, recall, and F1-score, which indicates excellent balance and consistency in prediction. This

superior performance highlights the effectiveness of combining CART-based and boosting approaches to enhance overall classification reliability and robustness in satellite mission analysis.

Table 2: Performance comparison of all the regression models for duration analysis target

Model	MAE	MSE	RMSE	R <sup>2</sup> Score
Random Forest	35.8970	2179.1271	46.6811	0.9237
Gradient Boosting	118.7667	25034.0234	158.2214	0.1232
SVR	324.9601	105979.5512	325.5450	-2.7117
Ensemble	2.7594	24.4241	4.9421	0.9991

Table 2 presents the performance comparison of different regression models for duration prediction, clearly showing variations in prediction accuracy and error levels. The RF (CART) model demonstrates strong performance with relatively low error values and a high R<sup>2</sup> score, indicating that it can effectively capture the relationship between telemetry features and mission duration. In contrast, the GB (CART) model shows significantly higher error values and a very low R<sup>2</sup> score, reflecting poor learning capability and inability to model the data accurately. The SVR (CART) model performs the worst among all, with extremely high error metrics and a negative R<sup>2</sup> score, indicating that its predictions are highly unreliable and worse than a simple baseline model. However, the Ensemble (CART) model outperforms all other models by a large margin, achieving extremely low error values and an R<sup>2</sup> score close to 1, which indicates near-perfect prediction accuracy. This highlights the effectiveness of combining multiple learning approaches to achieve highly precise and stable duration predictions in satellite telemetry analysis.

## 5. Conclusion

The study presents a powerful and scalable solution for satellite mission evaluation by utilizing a locally implemented ensemble learning architecture on OPSSATAD telemetry data. The framework unifies both classification and prediction tasks to determine mission outcomes and estimate execution duration within a single analytical pipeline. By employing tree-based CART models combined with boosting strategies, the proposed Ensemble Voting (CART) approach effectively learns complex patterns, feature dependencies, and nonlinear relationships in telemetry signals. Experimental findings indicate that standalone models such as RF (CART), GB (CART), and SVM (CART) show inconsistent performance, whereas the ensemble model delivers consistently superior results across all metrics. It achieves highly balanced classification performance along with extremely accurate regression outputs, reflected by minimal error values and a near-perfect R<sup>2</sup> score. The use of model serialization improves computational efficiency by enabling rapid inference without retraining. Additionally, the Flask-based interface supports real-time interaction, batch processing, and visual analysis. The approach provides a reliable and high-performance solution for satellite mission prediction and decision support systems.

## REFERENCES

- [1]. Wang, Q.; Tang, C. Deep reinforcement learning for transportation network combinatorial optimization: A survey. *Knowl.-Based Syst.* **2021**, *233*, 107526]

- [2]. Dai, C.Q.; Li, C.; Fu, S.; Zhao, J.; Chen, Q. Dynamic scheduling for emergency tasks in space data relay network. *IEEE Trans. Veh. Technol.* **2020**, *70*, 795–807.
- [3]. Zhou, D.; Sheng, M.; Wang, Y.; Li, J.; Han, Z. Machine learning-based resource allocation in satellite networks supporting internet of remote things. *IEEE Trans. Wirel. Commun.* **2021**, *20*, 6606–6621.
- [4]. He, Y.; Xing, L.; Chen, Y.; Pedrycz, W.; Wang, L.; Wu, G. A generic Markov decision process model and reinforcement learning method for scheduling agile earth observation satellites. *IEEE Trans. Syst. Man Cybern. Syst.* **2020**, *52*, 1463–1474.
- [5]. Vos, K.; Peng, Z.; Jenkins, C.; Shahriar, M.R.; Borghesani, P.; Wang, W. Vibration-based anomaly detection using LSTM/SVM approaches. *Mech. Syst. Signal Process.* **2022**, *169*, 108752.
- [6]. Martínez-Heras, J.; Donati, A. Enhanced Telemetry Monitoring with Novelty Detection. *AI Mag.* **2014**, *35*, 37–46
- [7]. Zeng, Z.; Jin, G.; Xu, C.; Chen, S.; Zhang, L. Spacecraft Telemetry Anomaly Detection Based on Parametric Causality and Double-Criteria Drift Streaming Peaks over Threshold. *Appl. Sci.* **2022**, *12*, 1803.
- [8]. Gao, Y.; Yang, T.; Xu, M.; Xing, N. An Unsupervised Anomaly Detection Approach for Spacecraft Based on Normal Behavior Clustering. In Proceedings of the 2012 Fifth International Conference on Intelligent Computation Technology and Automation, Zhangjiajie, China, 12–14 January 2012; pp. 478–481.
- [9]. Jin, X.; Wang, H.; Jin, Z. Anomaly detection of satellite telemetry data based on extended dominant sets clustering. *J. Phys. Conf. Ser.* **2023**, *2489*, 012036
- [10]. Bernard, D.; Doyle, R.; Riedel, E.; Rouquette, N.; Wyatt, J.; Lowry, M.; Nayak, P. Autonomy and software technology on NASA's Deep Space One. *IEEE Intell. Syst. Their Appl.* **1999**, *14*, 10–15.
- [11]. Ramachandran, S.; Rosengarten, M.; Belardi, C. Semi-Supervised Machine Learning for Spacecraft Anomaly Detection & Diagnosis. In Proceedings of the 2020 IEEE Aerospace Conference, Big Sky, MT, USA, 7–14 March 2020; pp. 1–10.
- [12]. Baireddy, S.; Chan, M.W.; Desai, S.R.; Foster, R.H.; Comer, M.L.; Delp, E.J. Spacecraft Time-Series Online Anomaly Detection Using Extreme Learning Machines. In Proceedings of the 2022 IEEE Aerospace Conference (AERO), Big Sky, MT, USA, 5–12 March 2022; pp. 1–9.
- [13]. Song, Y.; Yu, J.; Tang, D.; Yang, J.; Kong, L.; Li, X. Anomaly Detection in Spacecraft Telemetry Data using Graph Convolution Networks. In Proceedings of the 2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Ottawa, ON, Canada, 16–19 May 2022; pp. 1–6.