Thermopsheric Density Forecasting using Machine Learning

Team Digantara **Date:** May 26th, 2025

Dr. Bhargav Joshi
Data Scientist
Digantara Research and Technologies Pvt. Ltd.
Floor 9, Brigade Senate, 2, Hebbal Kempapura
Bengaluru, Karnataka 560024
bhargav.joshi@digantara.co.in

Archana Trivedi
Jr. Data Scientist
Digantara Research and Technologies Pvt. Ltd.
Floor 9, Brigade Senate, 2, Hebbal Kempapura
Bengaluru, Karnataka 560024
archana.trivedi@digantara.co.in

Sivalinga Raja Shanmugam
Data Scientist
Digantara Research and Technologies Pvt. Ltd.
Floor 9, Brigade Senate, 2, Hebbal Kempapura
Bengaluru, Karnataka 560024
sivalinga.raja@digantara.co.in

Saikat Majumder
Space Weather Analyst
Digantara Research and Technologies Pvt. Ltd.
Floor 9, Brigade Senate, 2, Hebbal Kempapura
Bengaluru, Karnataka 560024
saikat.majumdar@digantara.co.in

Thamim Ansari
Astrodynamics Engineer
Digantara Research and Technologies Pvt. Ltd.
Floor 9, Brigade Senate, 2, Hebbal Kempapura
Bengaluru, Karnataka 560024
thamim.ansari@digantara.co.in

Tanveer Ahmed
Chief Technology Officer
Digantara Research and Technologies Pvt. Ltd.
Floor 9, Brigade Senate, 2, Hebbal Kempapura
Bengaluru, Karnataka 560024
tanveerahmed@digantara.co.in

Abstract

The ”Satellite Tracking and Orbit Resilience Modeling with Artificial Intelligence” challenge addresses the critical need for advanced Space Domain Awareness (SDA) capabilities in an increasingly congested orbital environment. With over 11,000 active satellites orbiting Earth as of today and projections showing continued exponential growth throughout the next decade, traditional tracking and prediction methods face unprecedented demands. This challenge focuses on developing AI-powered solutions that improve satellite tracking accuracy, predict orbital perturbations due to space weather phenomena, and improve resilience modeling for satellite constellations. As solar activity approaches its maximum in 2025, with increased solar storms threatening satellite operations through communication disruptions and atmospheric drag fluctuations, AI solutions that can process vast amounts of complex data offer significant advantages over conventional methods. Preliminary results demonstrate that machine learning models can identify patterns in solar activity and orbital dynamics that traditional approaches often miss, potentially revolutionizing space situational awareness capabilities and improving the protection of critical space infrastructure. [a]

[a] All the code can be found in Github: https://github.com/bmjoshi/MIT-ARCLab25-STORMAI

Table of Contents

Introduction

The 2025 AI challenge hosted by MIT ARCLab presented a unique forecasting problem that required the model to ingest the initial orbital state of satellites along with comprehensive space weather datasets spanning the previous 60 days to generate accurate predictions of thermospheric orbit-averaged neutral mass densities over the subsequent 72-hour period. The input datasets included OMNI2 data providing high-resolution solar wind and interplanetary magnetic field measurements, GOES satellite data containing energetic particle flux measurements and solar X-ray observations, and initial satellite state vectors defining position and velocity components of the orbital configuration. The forecast horizon of 72 hours represents a critical time frame for satellite operations, as thermospheric density variations directly impact atmospheric drag calculations essential for orbital prediction and collision avoidance maneuvers. This report provides a comprehensive description of the approaches used to create an AI model for forecasting space weather-driven thermospheric densities.

Approach Definition

This problem presents a distinctive challenge that transcends conventional approaches to thermospheric density modeling. Unlike previous methodologies that have typically addressed this as either a pure time series forecasting problem or a static prediction task, this work required developing a hybrid approach that simultaneously handles both prediction and forecasting components. Traditional approaches to thermospheric density forecasting have generally fallen into two categories: classical time series forecasting methods that treat density prediction as a univariate or multivariate time series problem[1] incorporating space weather variables as exogenous co-variates, and static prediction models[2] that focus on predicting current or near-term densities based on recent space weather history using machine learning technique. But the problem becomes more difficult having to accurately predict densities along the trajectory of the satellite, given that the thermospheric densities vary exponentially[3][4] with the altitude. Several attempts have also been made to forecast the global space weather using AI[5], but such methods would be computationally expensive for this Challenge.

Given the high-dimensional nature of the input feature space, a physics-based approach was employed for variable selection and feature engineering, drawing upon decades of semi-empirical and physics-based modeling research in thermospheric physics. The feature selection process was anchored by insights from established empirical models, particularly the NRLMSISE-00 model [6], which served as the primary reference framework and is also used to derive the baseline values for scoring in this challenge. NRLMSISE-00 incorporates key space weather drivers including the solar radio flux F10.7 proxy which represents solar EUV variations, geomagnetic activity indices such as Kp or ap, seasonal and diurnal variations, and solar cycle effects. Additional insights were derived from the Jacchia-Bowman 2008 (JB2008) model [7], another empirical model that offers enhanced accuracy during geomagnetically disturbed conditions. JB2008 utilizes solar irradiance proxies in the EUV through the FUV range, including X-ray and Lyman-α wavelengths. The improved density characterization is achieved by using the ring-current distrurbance storm-time index (Dst) along with Kp during periods of enhanced geomagnetic activity. This physics-informed approach to density modeling enabled the identification and construction of derived features that capture the essential physical processes governing thermospheric variability, including multiple representations of solar radiation proxies and geomagnetic activity metrics in addition to F10.7 and ap, such as the Lyman alpha index and Dst which exhibit strong correlation with thermospheric density along with various parameters capturing different aspects of solar wind energy input like proton density, plasma velocity, magnetosonic mach number and the components of interplanetary magnetic field which have weak correlation with thermospheric density. Additionally, temporal lag features accounting for the delayed thermospheric response to space weather drivers, and cumulative effect indicators representing the integrated impact of sustained space weather activity were also considered. This comprehensive feature engineering strategy ensures that the model architecture is grounded in established physical understanding while remaining flexible enough to capture complex nonlinear relationships present in the observational data.

The methodology adopted in this work addresses and decomposes the complex forecasting problem into two interconnected components: a prediction component that establishes the relationship between space weather drivers

and instantaneous thermospheric density, and a time series forecasting component that captures the temporal evolution and propagation of density variations over the 72-hour forecast horizon. This decomposition allows for a physics-informed as well as data-driven way of development rather than a fully data-driven approach.

Data & Pre-processing

The thermospheric densities derived from accelerometer measurements (ACC) [8][9] of the space weather satellites were obtained from ESA's SWARM mission data portal[10]. Orbit-averaged density values were sampled at random timestamps within each satellite's operational period, covering over two decades (2000–2021) and spanning altitudes of 200–550 km in Low Earth Orbit (LEO). The dataset includes observations under varying levels of solar activity, as indicated by the F10.7 index. This data was chosen to cover all the satellites from different time periods as given in the Phase-1 training data (Table 10).

The OMNI2 space weather dataset, maintained by NASA's Goddard Space Flight Center, was acquired from NASA's OMNIWeb portal[11] for a 24-year period (2000–2024). This dataset comprises the 58 space weather variables specified in the challenge documentation[12]. To supplement this with X-ray flux measurements, data from multiple GOES satellite missions were merged, spanning 13 satellites since 1986 with overlapping coverage. For the period 2020–2025, X-ray data were extracted from four distinct GOES missions (Table 1), ensuring no temporal overlaps. Missing values were filled using linear interpolation.

GOES Mission	Period Start	Period End
8	2000/01/01	2003/06/16
10	2003/06/17	2009/12/01
14	2009/12/02	2020/03/04
15	2010/04/07	2020/03/04
16	2020/03/05	2025/01/14

Satellite	Period Start	Period End
CHAMP	2000/07/29	2010/09/05
GRACE-1	2002/04/04	2017/11/01
GRACE-2	2002/04/04	2017/05/23
GRACE-FO	2018/05/29	2024/06/01
SWARM-A	2013/11/28	2024/12/01
SWARM-B	2013/11/28	2024/12/01
SWARM-C	2013/11/28	2024/12/01

Table 1: GOES data sources (left). Time period corresponding to the density data. (right)

To train the decision tree models, all input features were scaled to ensure normalization (Table 5). Additionally, records containing invalid entries—such as infinite values in orbit-averaged density or geodetic coordinates—were removed to ensure data quality.

For the density forecasting model, a comprehensive time series dataset was constructed by generating random initial states based on randomly sampled timestamps for each satellite within the defined time span (Table 1). For each initial state, the corresponding 60-day history of space weather parameters was retrieved to serve as input features. The target outputs were the thermospheric orbit-averaged density values over the following 72 hours, normalized with respect to the initial density value at the starting timestamp. This relative normalization strategy allowed the model to learn changes in density trends rather than absolute magnitudes, thereby minimizing prediction drift and improving robustness.

Methods

Thermospheric density exhibits a strong exponential decrease with increasing satellite altitude, making it challenging for a single model trained on the entire dataset to effectively learn the relationships between density and other relevant features beyond altitude. To address this challenge and enable the utilization of multiple space weather features for more accurate density predictions, a strategy of creating multiple models, each tailored to a specific altitude range, was adopted (further details in Appendix A). Two different approaches of data categorization were tried:

1. Splitting of altitude in 5 bins followed by splitting of F10.7 in three bins of high, medium and low solar activity following similar approaches used in literature [13]. This binning procedure was included in the final highest scoring model, referred to as **xgb_v11**. This model was trained using data from **2020-2021**.

2. Another approach was taken to split the data into bins of altitude and ap index instead. Additionally, this model (submitted by user names archanatrivedi), referred to as **xgb_v12** had only used a few space weather parameters- f10.7, ap and Lyman Alpha. This model was trained using data from **2020-2024**.

Empirical models often incorporate space weather indices with time lags of up to five days. To capture this temporal dependency, the five-day history of each feature was condensed into four proxy variables derived through the application of four aggregation functions to the time series: mean, minimum, maximum, and standard deviation. Initially, a model was developed using only the three primary variables F10.7, ap, and Lyman Alpha that have high degree of correlation with the density. Subsequently, interaction terms were introduced to enable the model to learn non-linear relationships with the target variable. While X-ray measurements from GOES data, along with OMNI2 space weather variables, were explored, they did not demonstrate a significant improvement in the test scores and were therefore excluded from the final model (see Appendix A). Furthermore, certain variables with low feature importance scores were also dropped to enhance model efficiency and performance.

For the implementation of decision tree-based models, the XGBoost and scikit-learn libraries were utilized. The dataset was randomly shuffled and partitioned into training, validation, and testing sets in an 80:10:10 ratio. Model parameters were optimized using the hyperopt library, with the Root Mean Squared Error (RMSE) serving as the metric for selecting the optimal parameter set.

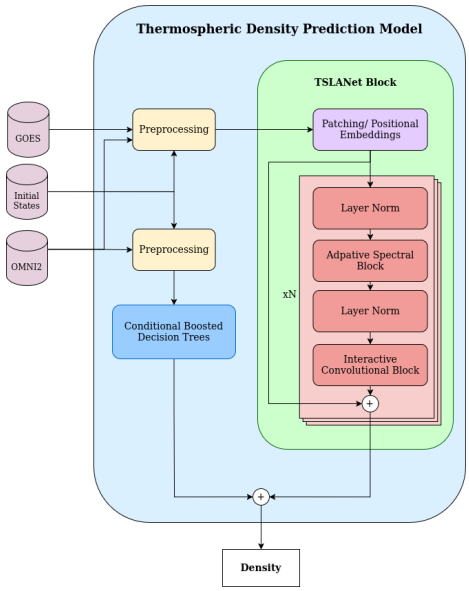

Figure 1: Schematic of the proposed neural forecast model.

The challenge's OD-RMSE metric applied exponentially decaying weights, making accurate initial density predictions crucial for high scores. However, this emphasis limited the model's ability to capture long-term trends over the 72-hour forecast horizon. To address this, a hybrid approach was adopted: initial densities were estimated first, followed by separate temporal trend forecasting. While orbit propagation was initially used to compute satellite positions, inconsistencies in the Phase 1.2 data revealed that the provided orbital elements did not reliably reproduce accurate positions (see Appendix B).

The **Time Series Lightweight Adaptive Network (TSLANet)** [14] offers a notable improvement over traditional Transformer Networks in time series forecasting. While Transformers effectively capture long-range dependencies via self-attention, TSLANet builds on this with adaptive temporal resolution and specialized temporal embeddings, enabling a more nuanced representation of time continuity. Its enhanced capacity allows it to handle a larger number of input features, making it suitable for incorporating additional variables—including those with weaker correlations to density and X-ray data—without compromising performance.

The combination of **xgb_v11** and **TSLANet** to create a hybrid **xgb_v11_tslanet_hybrid** model for the final submission (Figure 1).

Results

The forecasting performance of the models was quantitatively assessed using the OD-RMSE metric on the Phase-1 test set. The hybrid model **xgb_v11_tslanet_hybrid** demonstrated strong capabilities in capturing density variations, especially under nominal space weather conditions and during storm onset phases (Figure 2). It effectively modeled short-term responses by leveraging the strengths of conditional BDTs for accurate initialization and TSLANet for dynamic trend prediction. Notably, the relative error in density remained within acceptable margins for the majority of the forecast horizon. However, beyond 2022, the model's predictive accuracy diminished, likely due exclusion of post-2022 data in the training the predictive model. In contrast, the **xgb_v12** model maintained stable performance across all epochs, benefiting from its minimal reliance on time-sensitive space weather proxies and a simpler, interpretable design grounded in NRLMSISE-00 principles.

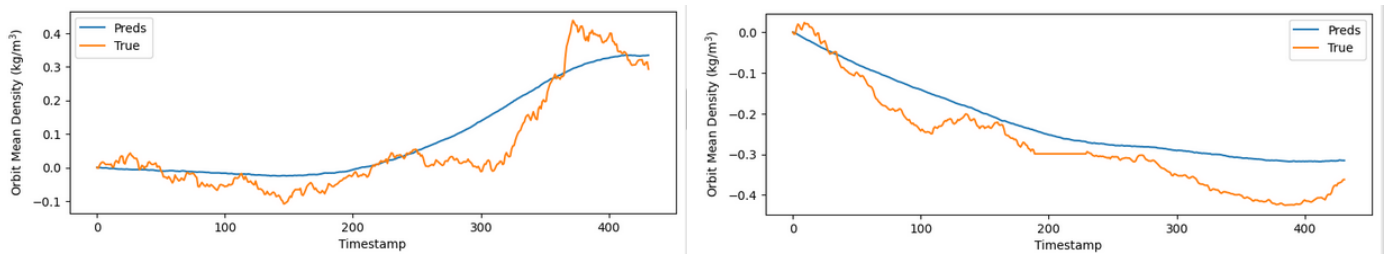

Figure 2: Response of the TSLANet in forecasting density variations after before (Left) and during (Right) a storm event.

Conclusion & Broader Impact

This study demonstrates the viability of hybrid machine learning architectures for thermospheric density forecasting in the context of space weather variability. By integrating physics-informed feature engineering with data-driven learning frameworks, the models not only achieved high accuracy on historical datasets but also offered generalizability across diverse solar conditions. The consistent performance of the **xgb_v12** model, in particular, suggests that even simpler decision-tree models can deliver robust forecasts when well-aligned with empirical models and physical intuition. These advances support improved space situational awareness and operational planning for satellite fleets, especially in low Earth orbit where drag-induced perturbations are significant. The framework developed here has potential applicability beyond this challenge—serving as a foundation for adaptive forecasting systems in real-world space traffic management scenarios.

Discussion & Model Evolution

The modeling strategy evolved significantly through iterative experimentation, beginning with conventional regression pipelines and gradually progressing to hybrid architectures that separated initialization from temporal trend forecasting. Early models struggled with generalizing across altitude ranges and capturing non-linear interactions under storm-time conditions. The inclusion of physics-based binning addressed data heterogeneity and improved model resolution. Additionally, the decision to discard complex propagation steps in favor of observed initial altitudes marked a critical turning point in reducing position-induced density errors. Future improvements could focus on integrating ensemble learning techniques and dynamically adjusting the feature space based on real-time solar indices. Moreover, expanding the architecture to jointly forecast position and density using coupled neural models could further bridge the gap between kinematic and environmental prediction in next-generation SDA systems.

Code availability

The code used to train, test and validate has been made publicly available in a Github https://github.com/bmjoshi/ along with links to data sources and the conda environment (yaml).

References

[1] P. Wang *et al.*, "The prediction of storm-time thermospheric mass density by lstm-based ensemble learning," *Space Weather*, vol. 20, no. 3, e2021SW002950, 2022, e2021SW002950 2021SW002950. DOI: https://doi.org/10.1029/2021SW002950. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2021SW002950. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2021SW002950.

[2] R. J. Licata, P. M. Mehta, W. K. Tobiska, and S. Huzurbazar, "Machine-learned hasdm thermospheric mass density model with uncertainty quantification," *Space Weather*, vol. 20, no. 4, e2021SW002915, 2022, e2021SW002915 2021SW002915. DOI: https://doi.org/10.1029/2021SW002915. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2021SW002915. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2021SW002915.

[3] J. Emmert, "Thermospheric mass density: A review," *Advances in Space Research*, vol. 56, no. 5, pp. 773–824, 2015, ISSN: 0273-1177. DOI: https://doi.org/10.1016/j.asr.2015.05.038. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0273117715003944.

[4] M. Moldwin, *An Introduction to Space Weather*. Cambridge University Press, 2008, p. 156, ISBN: 978-0521711128. [Online]. Available: https://ia801004.us.archive.org/0/items/anintroductiontospaceweather_201907/An%20Introduction%20to%20Space%20Weather.pdf.

[5] G. Acciarini *et al.*, "Improving thermospheric density predictions in low-earth orbit with machine learning," *Space Weather*, vol. 22, no. 2, e2023SW003652, 2024, e2023SW003652 2023SW003652. DOI: https://doi.org/10.1029/2023SW003652. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2023SW003652. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2023SW003652.

[6] J. M. Picone, A. E. Hedin, D. P. Drob, and A. C. Aikin, "NRLMSISE-00 empirical model of the atmosphere: Statistical comparisons and scientific issues," *Journal of Geophysical Research (Space Physics)*, vol. 107, no. A12, 1468, p. 1468, Dec. 2002. DOI: 10.1029/2002JA009430.

[7] B. Bowman, W. K. Tobiska, F. Marcos, C. Huang, C. Lin, and W. Burke, "A new empirical thermospheric density model jb2008 using new solar and geomagnetic indices," DOI: 10.2514/6.2008-6438. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2008-6438. [Online]. Available: https://arc.aiaa.org/doi/abs/10.2514/6.2008-6438.

[8] C. Siemes *et al.*, "New thermosphere neutral mass density and crosswind datasets from CHAMP, GRACE, and GRACE-FO," *Journal of Space Weather and Space Climate*, vol. 13, 16, p. 16, May 2023. DOI: 10.1051/swsc/2023014.

[9] E. Iorfida, I. Daras, R. Haagmans, and A. Strømme, "Swarm a and c accelerometers: Data validation and scientific interpretation," *Earth and Space Science*, vol. 10, no. 2, e2022EA002458, 2023, e2022EA002458 2022EA002458. DOI: https://doi.org/10.1029/2022EA002458. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2022EA002458. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2022EA002458.

[10] ESA, *ESA SWARM Data Portal*. [Online]. Available: https://earth.esa.int/eogateway/missions/swarm/data.

[11] NASA, *OMNI2 Data Portal*. [Online]. Available: https://omniweb.gsfc.nasa.gov/form/dx1.html.

[12] MITARCLab, *2025 AI Challenge Readthedocs*. [Online]. Available: https://2025-ai-challenge.readthedocs.io/en/latest/dataset.html.

[13] J. Briden, P. M. Siew, V. Rodriguez-Fernandez, and R. Linares, "Transformer-based atmospheric density forecasting," *Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, 2023. [Online]. Available: https://arxiv.org/abs/2310.16912.

[14] E. Eldele, M. Ragab, Z. Chen, M. Wu, and X. Li, *Tslanet: Rethinking transformers for time series representation learning*, 2024. arXiv: 2404.08472 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2404.08472.

[15] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," *CoRR*, vol. abs/1603.02754, 2016. arXiv: 1603.02754. [Online]. Available: http://arxiv.org/abs/1603.02754.

[16] MITARCLab, *STORM-AI-devkit-2025*. [Online]. Available: https://github.com/ARCLab-MIT/STORM-AI-devkit-2025.

Model Development

The following section describes model development in more details. For the initial development XGBoost[15] was chosen as the baseline AI solution. Later different models were tried out to see if they offer any significant improvements over the baseline AI solution. The training data was generated independently from the open source databases mentioned in Section 1.2. The training dataset released in the phase-1 of the competition was used only to evaluate the models2 and select the best model accordingly.

Table 2: OD-RMSE scores of the models described in this paper evaluated on test data.

Model	Test Score
xgb_v12	0.7979
xgb_v11	0.9034
xgb_v11_tslanet_hybrid	0.9388

Model xgb_v11_tslanet_hybrid

XGBoost is an optimized, distributed gradient boosting library that uses decision trees as base estimators for solving regression problems. It is specifically designed to enhance computational efficiency and reduce training time. The library leverages vectorized operations, making the algorithm significantly faster at generating inferences. Additionally, the tree-based structure of the estimators enables the evaluation of feature importance by analyzing how frequently each feature is used, weighted by its contribution to reducing the loss function. Thus the *feature importance* can be used to select the most discriminating features while leaving out the features that do not contribute much to improving the accuracy of the model. The selected features can be used further to build more complex model.

A dataset of 893,314 density values was binned by altitude and F10.7 levels (Table 1). Due to the scarcity of data below 340 km during medium and high solar activity, all such entries were grouped into a single bin. Input features—including altitude and space weather variables—were scaled using the ranges provided in Table 5. Model training involved careful hyperparameter tuning to prevent overfitting within individual bins (Figure 3). Data sufficiency across all bins was ensured to support deeper, more expressive decision trees. The feature importance was evaluated across all bins (Figure 4). It showed a reduced dependence on the altitude while enhancing the usage of other space weather features. Quadratic terms of f10.7, ap as well as cross-terms such as f10.7 $\times$ Lyman Alpha, f10.7 $\times$ ap are also included as they seem to boost the overall model performance. X-ray B measurements were added to the OMNI2 feature set, and the model was retrained. However, results showed no significant improvement (Table 4), leading to the exclusion of X-ray features in the final version of the BDT model.

Alt (km)/ F10.7	[63.4, 110.5]	[110.5, 156.7]	>156.7
[200, 270]	1698	0	0
[270, 340]	46142	206	57
[340, 410]	113190	29329	9776
[410, 480]	257625	107029	47125
[480, 550]	192269	59997	24259

Alt (km) F10.7	[63.4, 110.5]	[110.5, 156.7]	>156.7
[200, 270] **[270, 340]**	alt_0_f10_0		
[340, 410]	alt_2_f10_0	alt_2_f10_1	alt_2_f10_2
[410, 480]	alt_3_f10_0	alt_3_f10_1	alt_3_f10_2
[480, 550]	alt_4_f10_0	alt_4_f10_1	alt_4_f10_2

Table 3: 2D altitude-F10.7 binning for training BDT models- data breakdown (left) and bin labels (right).

Alt (km)/ F10.7	[63.4, 110.5]	[110.5, 156.7]	>156.7
[200, 270] **[270, 340]**	-2.44E-02		
[340, 410]	-1.23E-03	-1.71E-04	-8.56E-04
[410, 480]	+3.02E-04	+7.03E-04	+4.28E-04
[480, 550]	-1.63E-04	-8.62E-07	7.50E-05

Table 4: Difference in the test RMSE values after adding X-ray flux-related features.

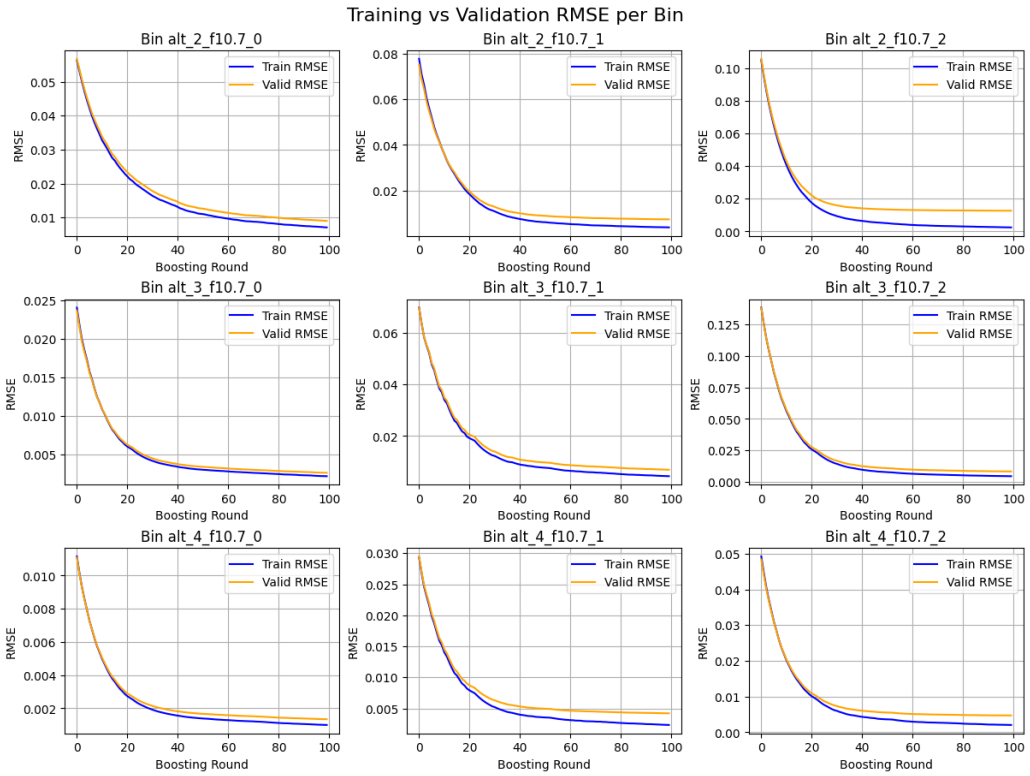

Figure 3: Plot showing the training vs validation scores in each bin.

Table 5: Minimum and Maximum values used for the features scaling in xgb_v11.

Variables	Minimum	Maximum
altitude	0	1000
average_ap_index_nT	0	400
average_f10.7_index	0	63.425
average_Lyman_alpha	0.00588	0.010944
average_Dst_index_nT	-422	71
average_Lyman_alpha2	0	1.00e-4
average_f10.7_index2	3969	62500
average_Lyman_alpha_f10.7	0	2.75
average_ap_index_nT2	0	1.60e+5
average_ap_index_nT_f10.7	0	1.00e+5
average_xrsb_flux	0	4.00e-6
average_xrsb_flux2	0	2.00e-11
average_xrsb_flux_Lyman_alpha	0	3.00e-8
min_ap_index_nT	0	400
min_f10.7_index	0	63.425
min_Lyman_alpha	0.00588	0.010944
min_Dst_index_nT	-422	71
min_Lyman_alpha2	0	1.00e-4
min_f10.7_index2	3969	62500
min_Lyman_alpha_f10.7	0	2.75
min_ap_index_nT2	0	1.60e+5
min_ap_index_nT_f10.7	0	1.00e+5
min_xrsb_flux	0	4.00e-6
min_xrsb_flux2	0	2.00e-11
min_xrsb_flux_Lyman_alpha	0	3.00e-8
max_ap_index_nT	0	400
max_f10.7_index	0	63.425
max_Lyman_alpha	0.00588	0.010944
max_Dst_index_nT	-422	71
max_Lyman_alpha2	0	1.00e-4
max_f10.7_index2	3969	62500
max_Lyman_alpha_f10.7	0	2.75
max_ap_index_nT2	0	1.60e+5
max_ap_index_nT_f10.7	0	1.00e+5
max_xrsb_flux	0	4.00e-6
max_xrsb_flux2	0	2.00e-11
max_xrsb_flux_Lyman_alpha	0	3.00e-8

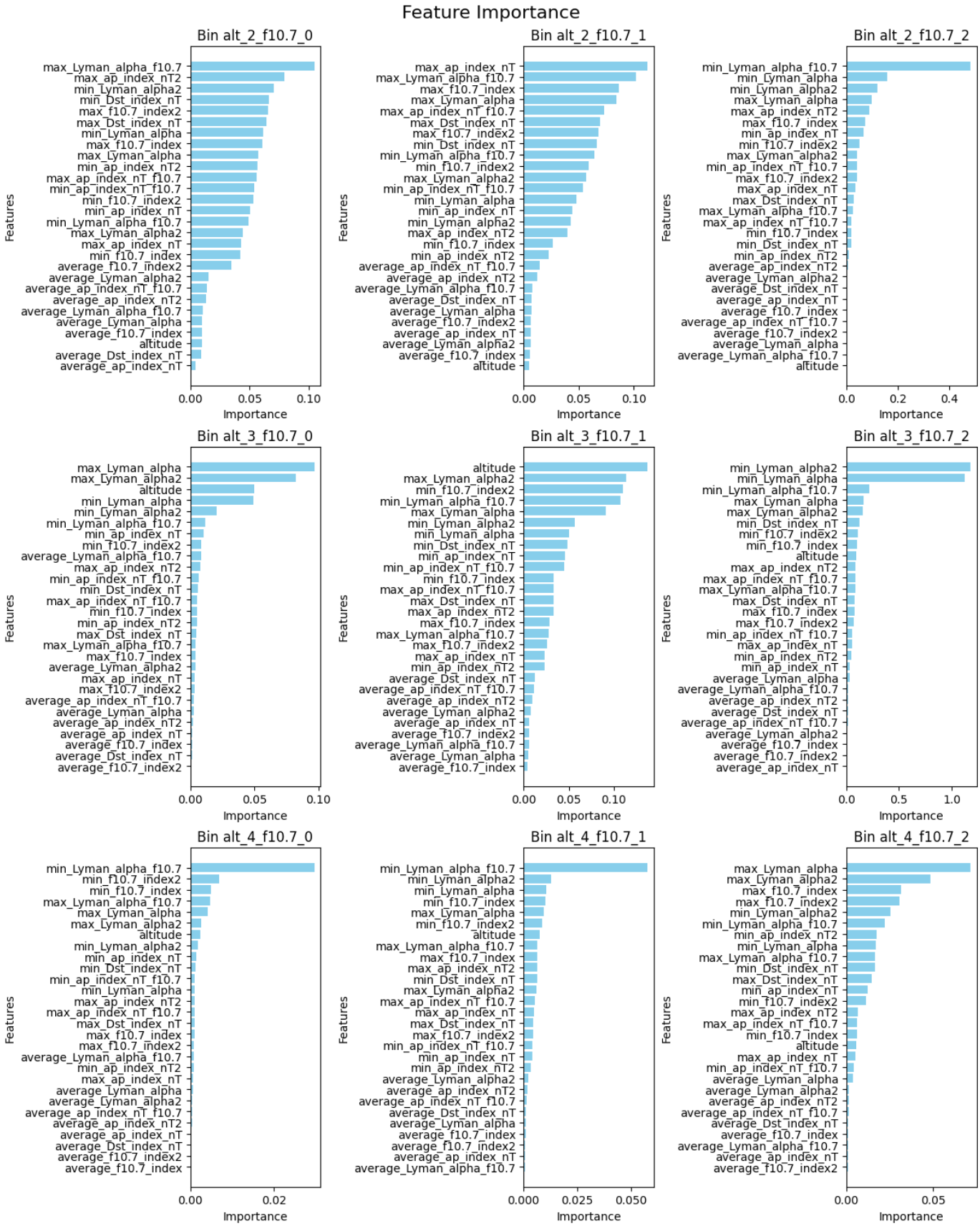

Figure 4: Plot showing the input feature importance in each bin.

Model xgb_v12

This model was developed independently of the xgb_v11_tslanet_hybrid, though it followed a similar approach to xgb_v11. A new training dataset comprising approximately 1.3 million density values was generated and binned into two altitude ranges and five Ap index categories (Tables 7 and 6). Input features—including geodetic coordinates and OMNI2 space weather parameters—were scaled using the ranges shown in Table 8. During pre-processing the invalid entries from the space weather data and the densities were dropped. Training the model with invalid entries would lead to sudden spikes in the density predictions5.

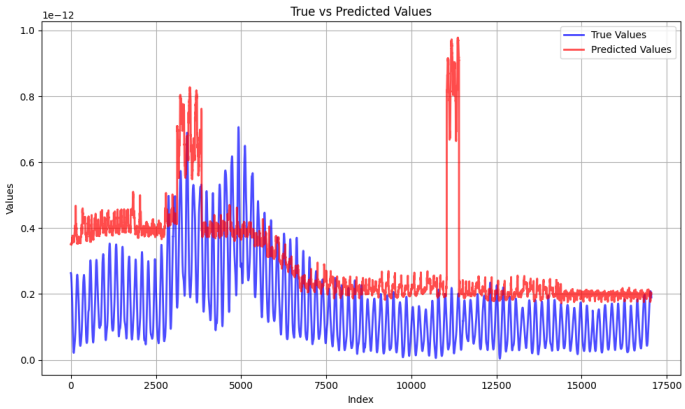

Figure 5: The output of an XGBoost trained with invalid data on a space weather time series.

Alt (km)/ ap	[0, 32]	[33, 65]	[66, 95]	[95, 155]	156-301
[200, 400]	AP_0-32_Alt_201-400	AP_33-65_Alt_201-400	AP_66-95_Alt_201-400	AP_95-155_Alt_201-400	AP_156-301_Alt_201-400
[400, 600]	AP_0-32_Alt_401-600	AP_33-65_Alt_401-600	AP_66-95_Alt_401-600	AP_95-155_Alt_401-600	AP_156-301_Alt_401-600

Table 6: 2D altitude-ap binning for training BDT models.

Alt (km)/ ap	[0, 32]	[33, 65]	[66, 95]	[95, 155]	156-301
[200, 400]	103342	4068	715	315	305
[400, 600]	1151492	46238	8287	2049	1068

Table 7: Training data breakdown.

Table 8: Minimum and Maximum values used for the features scaling in xgb_v12.

Variables	Minimum	Maximum
altitude	0	1000
latitude	-90	90
longitude	-180	180
average_AP_index_nT	0	400
average_f10.7_index	0	200
average_Lyman_alpha	0	0.1
std_AP_index_nT	0	400
std_f10.7_index	0	200
std_Lyman_alpha	0	0.1
min_AP_index_nT	0	400
min_f10.7_index	0	200
min_Lyman_alpha	0	0.1
max_AP_index_nT	0	400
max_f10.7_index	0	200
max_Lyman_alpha	0	0.1

TSLANet

The primary objective of this model was to capture the temporal dynamics of thermospheric density through time series forecasting. Training was conducted using the PyTorch[1] library in Python. A custom time series dataset was

[1] https://pytorch.org/

constructed, consisting of 19 input features sampled at 1-minute intervals over a 24-hour window, resulting in input tensors of shape (19×1440). The target output was a vector of 432 orbit-averaged density values, representing a 72-hour forecast horizon at 10-minute resolution.

All input features were scaled using the normalization scheme described in Table 9, consistent with the preprocessing steps outlined in earlier sections. The model was trained using the Mean Absolute Error (MAE) loss function, as it provided more stable convergence compared to Mean Squared Error (MSE). A custom exponentially weighted MAE loss was also tested to emphasize early-time predictions, but it led to significant degradation in forecast accuracy beyond the 24-hour mark.

Table 9: Minimum and Maximum values used for the scaling the features in TSLANet.

Variables	Minimum	Maximum
altitude	0	1000
ap_index_nT	0	400
f10.7_index	0	63.425
Lyman_alpha	0.00588	0.010944
Dst_index_nT	-422	71
BX_nT_GSE_GSM	-40.8	34.8
BY_nT_GSE	-33.2	63.4
BZ_nT_GSE	-53.7	37.5
SW_Proton_Density_N_cm3	0.1	137.2
SW_Plasma_Speed_km_s	233	1189
Magnetosonic_Mach_number	0.6	14.3
log_Lyman_alpha2	4	8
f10.7_index2	3969	62500
Lyman_alpha_f10.7	0	2.75
ap_index_nT2	0	1.60e+5
ap_index_nT_f10.7	0	1.00e+5
log_xrsb_flux	2.5	5
log_xrsb_flux2	5	10
log_xrsb_flux_Lyman_alpha	5	9

Orbit Propagation

Orekit's numerical and SGP4 propagators represent two fundamentally different approaches to satellite orbit prediction, each suited for specific use cases and accuracy requirements. Orekit's numerical propagator uses advanced integration techniques to solve the equations of motion, accounting for detailed perturbation forces such as atmospheric drag, solar radiation pressure, gravitational harmonics, and third-body effects. While it offers high precision ideal for mission-critical applications and scientific analysis, it requires detailed satellite parameters (e.g., mass, cross-sectional area, drag coefficient) and is computationally intensive.

In contrast, SGP4 (Simplified General Perturbations 4) is an analytical model optimized for Two-Line Element (TLE) data. It employs simplified atmospheric and gravitational models that prioritize speed over precision. Though SGP4 is computationally efficient and useful for space situational awareness (SSA) tasks, its accuracy declines over time, especially in low Earth orbit where drag effects are pronounced.

The STORMAI-devkit [16] includes an implementation of Orekit's numerical propagator. While theoretically capable of high accuracy, its application in the challenge was limited by three key constraints:

1. The initial states were provided as mean orbital elements, which had to be converted to Cartesian coordinates. This conversion, using Orekit's Python wrapper, did not match GPS-derived truth data.

2. Essential satellite-specific parameters such as mass and cross-sectional area were missing from the provided inputs.

3. Even with reduced *degree* and *torder* settings in the gravity model, propagation times were 7 seconds per sample, making it impractical for batch processing within the challenge's computational limits.

A case study was conducted to benchmark a basic SGP4 propagator against the Orekit implementation. The SGP4 predictions were also validated against ground-truth GPS data from ESA's SWARM mission[10] (Figure

6). Results showed that altitude errors remained within ±1 km, while latitude and longitude errors stayed within 0.25° and 5°, respectively. Given that longitudinal errors had minimal impact on orbit-averaged mean density—and considering the 35 km altitude bin size—SGP4's accuracy was deemed sufficient for the modeling goals of this study.

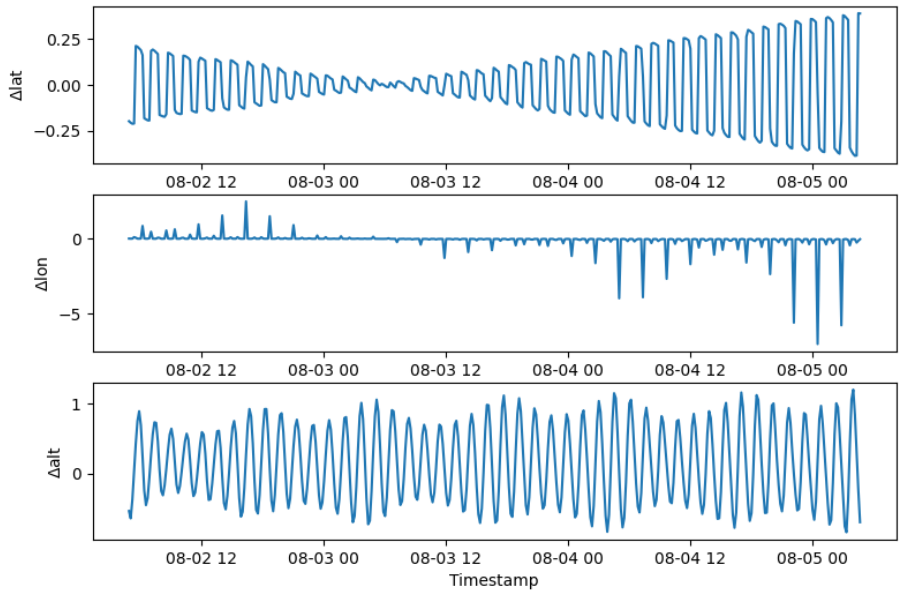

Figure 6: Evolution of errors in geodetic coordinates over 72 hours of propagation using the SGP4 propagator.

Another key limitation in using propagators was the dependence on precise orbital elements. Most initial states from Phase 1 were accurate, except those tied to File IDs 6672–8118. These were verified against TLEs from space-track.org. Initially, the models used SGP4 to compute altitude variations, but performance degraded significantly with the Phase 1.2 dataset. It was suspected that these new initial states had similar inconsistencies. Consequently, the propagation step was removed, and the models instead used the altitudes directly from the initial states, which closely matched the ground-truth GPS data.

Table 10: Phase 1 MIT Training Dataset

	Start_time (UTC)	**End_time** (UTC)	**Files**	**File ID**	**NORAD ID**	**INTL Designation**	**Mass**	**xs (Max)**	**A/m**	**T (min)**
CHAMP	01/08/2000 19:30:00	01/08/2010 19:30:00	2285	0-2284	26405	2000-039B	522	3.51	0.0067	-
GRACE-1	09/09/2002 18:30:00	05/08/2017 19:30:00	117	2285-2357,	27391	2002-012A	475	2.51	0.0053	-
	18:30:00	05/08/2017 19:30:00	117	5571-5614,	27391	2002-012A	475	2.51	0.0053	-
GRACE-2	01/05/2002 19:30:00	01/09/2016 19:30:00	2964	2358-4264,	27392	2002-012B	475	2.51	0.0053	-
				5615- 6671						-
GRACE-FO	31/05/2018 19:30:00	01/01/2021 18:30:00	1306	4265, 5770	43476	2018-047A	600	10.08	0.0168	94
SWARM-A	02/01/2014 18:30:00	02/01/2020 18:30:00	1447	6672, 8118	39452	2013-067B	473	3.85	0.0081	93.6