Karman - a Machine Learning Software Package for Benchmarking Thermospheric Density Models

Giacomo Acciarini
Surrey Space Center, University of Surrey, Guildford, UK
Edward Brown
Computer Science Department, University of Cambridge, Cambridge, UK

Christopher Bridges
Surrey Space Center, University of Surrey, Guildford, UK

Atılım Güneş Baydin
Department of Engineering Science & Department of Computer Science, University of Oxford, Oxford, UK

Thomas E. Berger
Space Weather Center, CU Boulder, Boulder, USA

Madhulika Guhathakurta
NASA Headquarters, Washington DC, USA

ABSTRACT

Recent events, such as the loss of 38 satellites by SpaceX due to a geomagnetic storm have highlighted the importance of having more accurate estimation and prediction of thermospheric density. Solar and geomagnetic activities wield significant influence over the behavior of the thermospheric density, exerting an important impact on spacecraft motion in low-Earth orbit (LEO). The impending Solar Cycle 25's peak arrives at a time when the number of operational satellites in LEO is surging, driven by the proliferation of mega-constellations. This escalating satellite presence, spanning sectors from defense to commercial applications, increases the intricacy of the operational environment. The accuracy of thermospheric neutral density models, which underpin crucial safety-oriented tasks like satellite collision avoidance and space traffic management, is therefore pivotal. While the importance of solar events on thermospheric density is apparent, currently, the influence of the Sun in thermospheric density models is only included in the form of solar proxies (such as F10.7). This can be underwhelming, leading to mispredictions of thermospheric density values. A shared framework that supports the ingestion of inputs from various sources to devise thermospheric density models, and where thermospheric density models can be compared, is currently lacking. Furthermore, the recent advancements in machine learning (ML) offer a unique opportunity to construct thermospheric density models that use these models to describe the relationship between the Sun and the Earth's thermosphere. For this reason, this study introduces an open-source software package, called Karman, to help solve this problem. Essential for this, are three steps: first, the preparation and ingestion of input data from several sources in an ML-readiness fashion. Then, the construction of ML models that can be trained on these datasets. Finally, the creation of a benchmarking platform to compare ML models against state-of-the-art empirical models, evaluating their performances under varying conditions, such as geomagnetic storm strength, altitude, and solar irradiance levels. The utility of this framework is demonstrated through various experiments, showcasing its effectiveness in both benchmarking density models and discerning factors driving thermospheric density variations. The study compares the performance of traditional empirical models (NRLMSISE-00 and JB-08) with machine learning models trained on identical inputs. The results reveal a consistent 20-40% improvement in accuracy, highlighting the potential of machine learning techniques. One particularly significant area addressed by this research involves the incorporation of additional inputs to refine density estimations.

Current approaches rely on solar proxies for estimating the Sun's impact on the thermosphere. However, it is suggested that direct Extreme Ultraviolet (EUV) irradiance data could enhance accuracy. The framework outlined in this paper enables the integration of such inputs, facilitating the validation of hypotheses and supporting the evolution of thermospheric density models. In conclusion, this study presents a comprehensive framework for advancing thermospheric density modeling in the context of LEO satellites. Through the development of neural network models, an extensive dataset, and a benchmarking platform, the paper contributes significantly to the improvement of satellite trajectory predictions. As the space environment becomes increasingly intricate, tools such as the presented framework are crucial for maintaining the safety and effectiveness of satellite operations in LEO.

1. INTRODUCTION

Recently SpaceX lost 38 satellites due to a geomagnetic storm that unexpectedly increased atmospheric drag, preventing satellites from leaving safe mode and forcing them to deorbit [8, 12]. This unfortunate event came at an estimated cost of $25 million. Such episodes underscore the potential hazards that geomagnetic storms and solar events, in general, can pose to the space environment and other valuable space assets.

To mitigate these risks and ensure the integrity of space operations, an enhanced understanding of thermospheric density and a more precise model detailing the Sun's intricate influence on Earth's atmosphere have become indispensable requisites in mission planning, execution, and overall space environment preservation.

Solar activity dominates the behavior of the thermospheric density and significantly influences the estimation and prediction of spacecraft motion in low-Earth orbit (LEO) [7, 20, 2]. As we approach the imminent solar maximum, the population of operational satellites in LEO is experiencing an exponential surge, with the planned launch of mega-constellations.

For crucial safety-related tasks within LEO, encompassing satellite collision avoidance, space traffic management, and re-entry projections, entities such as NASA heavily depend on empirical density models. Yet, the existing models exhibit suboptimal performance in assessing and forecasting the impacts of solar activity on the thermosphere [10]. This discrepancy necessitates a paradigm shift towards more refined models that can accurately encapsulate the solar influence, thus ushering in a new era of safer space operations.

State-of-the-art solutions for estimating thermospheric density predominantly rely on solar proxies, such as F10.7, to gauge the Sun's impact. However, it has been suggested that direct observations based on extreme ultraviolet (EUV) measurements themselves might better capture the complexity of the Sun, rather than indirect proxies [26]. The drawback is that EUV measurements have only been collected in the last 20/30 years. Instruments like TIMED/SEE, SOHO, SDO/EVE, and GOES-R/EXIS have been instrumental in acquiring these measurements, yet challenges persist. Disparities in wavelength coverage and data cadence, along with recurring issues of instrument degradation, introduce uncertainty and noise, thereby hindering the utilization of EUV measurements in operational contexts [1].

Numerous endeavors have sought to refine inputs for thermospheric density models, as well as to ascertain the significance of current inputs across diverse solar activity periods. However, an important void remains: the absence of a collaborative open-source framework that facilitates the comparison and evaluation of various models. This research follows up on a previous work from the authors [3]. It addresses this deficiency by introducing Karman (Fig. 1), a machine learning (ML) software package designed for both thermospheric density modeling and benchmarking. Written in Python, this package encompasses three distinct components.

Fig. 1: Karman is an ML open-source library for data-driven thermospheric density modeling, released at `https://github.com/spaceml-org/karman`.

The first component focuses on dataset management, enabling the assimilation of inputs from diverse sources. These inputs are temporally aligned and prepared to be readily utilized within ML frameworks. The second component is dedicated to the modeling aspect, offering different neural network models. These models can be employed for training purposes or directly employed as pre-trained models. Importantly, these ML models encompass inputs spanning solar, geomagnetic, and location data, enabling predictions concerning thermospheric density behavior.

The third component extends to a benchmarking framework, allowing for a comparative analysis of ML models against widely utilized empirical density models, such as NRLMSISE-00 and JB-08 [17, 4]. This capability empowers researchers to evaluate their self-developed or pre-trained solutions across various altitude ranges, temporal contexts, and diverse geomagnetic and solar scenarios. In essence, Karman introduces a collaborative platform that advances the understanding of thermospheric density modeling while enhancing the capacity to perform reliable predictions for space operations in an increasingly intricate environment.

The paper is structured as follows: in Sec. 2, we discuss the related work and background information relevant for the tackled problem. Then, in Sec. 3, we introduce the methodology: this encompasses both the dataset description, as well as the proposed models and benchmarking framework. Moreover, in Sec. 4, we discuss some experiments and test cases to show the software usage. Finally, in Sec. 5, we present the conclusions of the work.

2. RELATED WORK

Various physics-based thermospheric neutral density models have been created over the years. Earlier examples include the widely used NRLMSISE-00 model [17] and the JB08 model [4], which are semi-empirical models using a mix of proxies for solar activity such as the F10.7 index, as well as geomagnetic information to make their estimates. The NCAR/TIE-GCM model is a simulation model of the upper atmosphere that has been used to provide density estimates (amongst other outputs), although being a large general circulation model means access to its outputs is limited [21].

More recently machine learning models have been developed to leverage large datasets, in a data-based approach as opposed to a physics-based approach. Pérez et al. (2014) [19] train artificial neural networks to forecast thermospheric density estimates for the CHAMP satellite. They follow this work up by using density estimates of three different empirically-based models to obtain an improved estimate [18]. George et al. (2020) [9] use long short-term memory networks (LSTMs) to forecast thermospheric density prediction for CHAMP and the Gravity Recovery and Climate Experiment (GRACE) missions. The precursor to this work used LSTMs applied to data from 6 satellites to calculate the density as well as include a measure of uncertainty [3]. Furthermore, the META-HASDM model has applied machine learning to the HASDM thermospheric density database to produce density estimates [14, 13]. Other works have used reduced order modeling techniques to model thermospheric density behaviors or machine learning models to predict the long-term behavior of thermospheric density trends [15, 27, 25].

3. METHODOLOGY

3.1 Dataset Preparation

For the thermospheric density prediction task, the availability of a software pipeline that facilitates users to combine inputs from different sources and prepares them for ML readiness is paramount. Therefore, it is important to support solar inputs, geomagnetic inputs, and thermospheric density inputs (both ground-truth data that can be used as targets for ML models, as well as empirical density outputs that can be used as a baseline). We use the TU Delft thermospheric density data as ground truth data for the thermospheric density[1] [6, 22]. This encompasses a dataset of high-frequency (e.g. 30 or 10 seconds intervals) thermospheric density values, spanning 20 years, at different altitudes and geographic locations. The data is derived via a precise orbit determination procedure from five satellites: CHAMP, GOCE, GRACE, SWARM-A, SWARM-B, and, more recently, SWARM-C. In terms of empirical thermospheric density models, we use NRLMSISE-00 and JB-08 models as baseline: these are two of the most used models in operational contexts. Alongside their output values, the dataset also supports their inputs: in this way, ML models can be trained on exactly the same inputs and a fair comparison between ML and empirical models can be performed, which we will discuss more in detail in Sec. 4. Hence, we support solar proxies (e.g., F10.7, S10.7, M10.7, Y10.7) and geomagnetic indices (e.g., DST, Ap) that are used by these models[2],[3] [24]. Additionally, extra information concerning

[1] `http://thermosphere.tudelft.nl/`, date of access: August 2023.

[2] `https://celestrak.org/`, date of access: August 2023.

[3] `https://spacewx.com/`, date of access: August 2023

solar irradiance and geomagnetic inputs is also supported. In terms of geomagnetic inputs, there is support for all the NASA's OMNIWeb high-resolution data. This encompasses the magnitude of the interplanetary magnetic field, the magnetic field vector's X,Y,Z-components both in GSM and GSE coordinate systems, SYM/D, SYM/H, ASY/D, ASY/H indices, and the X,Y,Z components of the bow shock nose, among others[4]. As for the solar irradiance, besides the aforementioned solar proxies, the data pipeline also supports ingestion of FISM2 "stan bands" from the Laboratory of Atmospheric and Space Physics (LASP) at CU Boulder[5] [5]: this is a collection of 23 bands derived using wavelength binning techniques, spanning the ultraviolet range from 0.01 to 121nm. These bands have been developed to be used as inputs in models of the Earth's ionosphere and thermosphere [23].

All these data, coming from different sources, are aligned temporally (at the same times as the thermospheric ground truth data points) and prepared for ML readiness (both in terms of data cleaning and normalization): the structure of a queried date is shown in Fig. 3. Then, the user can query a specific date (or set of dates) and obtain all these data points (or a subset of it) for the requested times. The supported data cover a period of more than 20 years (e.g. about 2 solar cycles), with a high cadence (e.g. thermospheric density data is available at a few seconds cadence), which makes the data particularly interesting for thermospheric density modeling. In order to train ML models that can ingest these data and compare experiments and models over the same data, it is essential to devise a strategy to divide training, validation, and test sets. The general approach is to train the ML models on the training set, evaluate them on the validation set, and finally report the performances on an unseen test set. To do this, we divide each year into nine months of training, one month of validation, and one month of test, making sure to cyclically permute the test and validation months, for each different year. This is explained in Fig. 3.

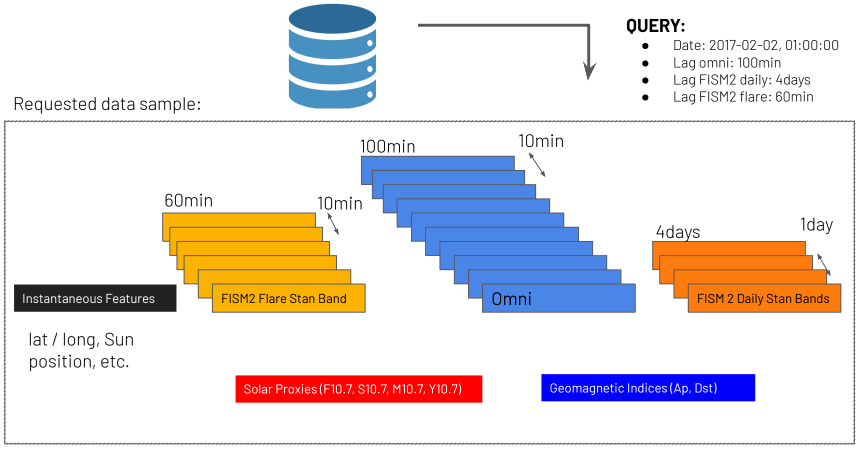

Fig. 2: A sample for the queried data at a given date, assuming that all the supported inputs are requested.

3.2 Machine Learning Models

To be able to perform thermospheric density prediction it is important to have ML architectures that can ingest and process the supported data. For the high temporal cadence of the data, and the possibility to request a time history of the inputs, it is important to have dimensionality reduction techniques (e.g. encoder-like architectures) that can handle such complexity and reduce the input size. The data can be prepared for thermospheric density prediction in two ways. One possibility is that it is used as a direct input for thermospheric density prediction: for this, a feed-forward neural network is used. Another option is to use ML models (e.g. feed-forward neural networks, LSTMs, or CNNs) to first reduce the dimensionality of some of the data, which is then concatenated and fed into a final neural network for thermospheric density prediction. This more complex architecture supported by Karman is highlighted in Fig. 4. This more complex architecture is important to compress the size of the inputs and condense their information to a limited number of features, which can be more easily handled by the ML models.

In terms of ML training, we use the mean squared error (MSE) during training, and we also use the mean absolute percentage error (MAPE) to assess the performance of the prediction. The latter is more informative for downstream

[4] `https://omniweb.gsfc.nasa.gov/`, date of access: August 2023.
[5] `https://lasp.colorado.edu/`, date of access: August 2023.

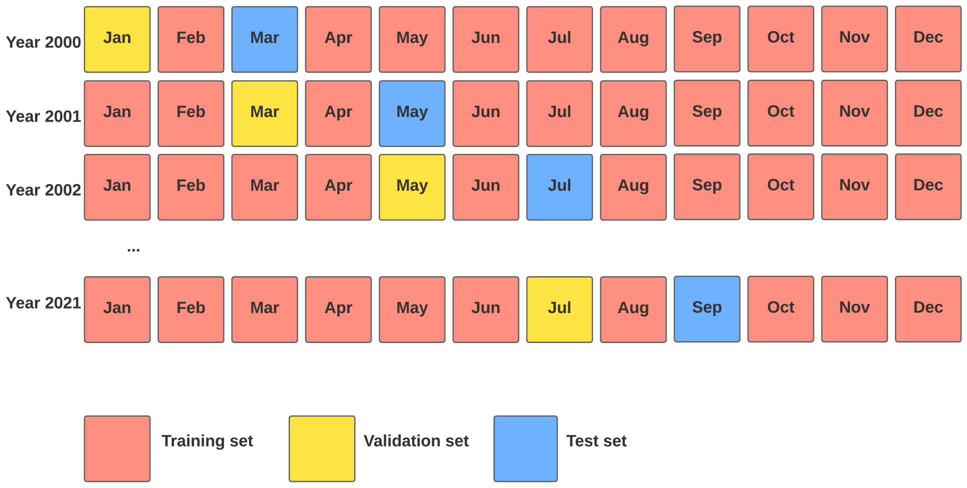
Year 2000
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
Year 2001
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
Year 2002
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
...
Year 2021
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
Training set
Validation set
Test set

Fig. 3: Dataset partitioning scheme with orange, yellow, and blue months added to the training, validation, and test sets respectively.

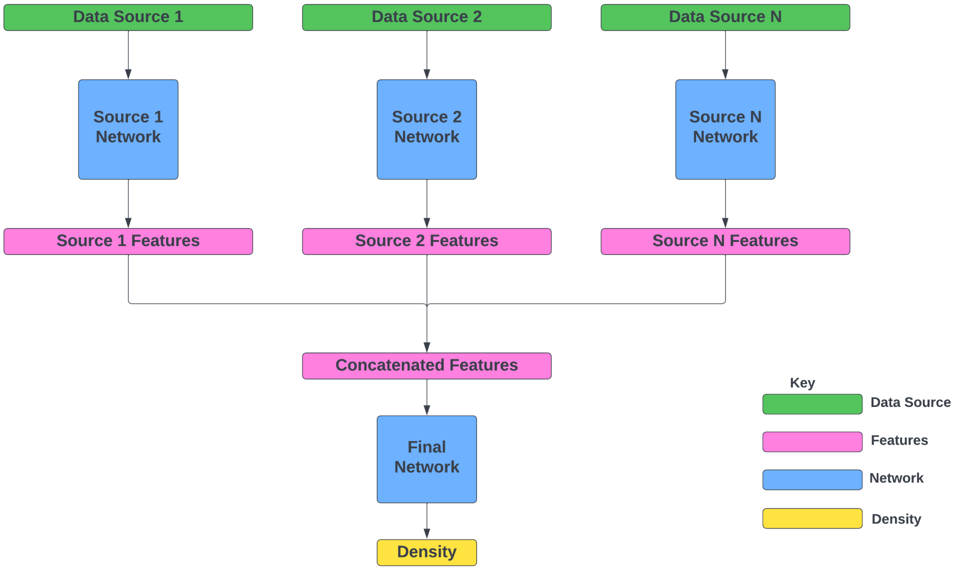

Fig. 4: General architecture for reducing the dimensionality of the data and predicting thermospheric density values.

tasks because it directly gives an insight into the percentage error in the force on the spacecraft due to drag calculations. We use the default implementation of the Pytorch Adam optimization algorithm to minimize the neural network loss [11, 16]. The weight decay is set at 1.37×10^{-8} is used, and the learning rate at 5×10^{-5}. Finally, due to the randomness inherent in stochastic gradient descent, each experiment is run five times, with different seeds and the reported results are a mean over five runs of the reported metrics.

3.3 Benchmarking Framework

The thermospheric density ground truth data ranges from 200-550 km, with various orders of magnitude differences. Similarly, depending on the geomagnetic storm conditions, there can be considerable thermospheric density enhancements. It is therefore important to assess the performance of the developed ML models against the state-of-the-art empirical models not only overall, but also at different altitude ranges, and geomagnetic storm conditions. For this reason, we report the results of our experiments at different altitude bins: from 200 km to 550 km altitudes, with a

50 km binning scheme. As for the geomagnetic storm, we classify the geomagnetic storms as quiet (for an Ap value of 0-15), mild (for an Ap value of 15-30), minor (for Ap of 30-50), and major (for Ap above 50). In Sec. 4, we will perform experiments and report their results at with the above binning scheme.

4. EXPERIMENTS

4.1 Empirical Density Models Comparison

In this case, we train feed-forward neural networks using the same inputs as JB-08 and NRLMSISE-00: the objective is to investigate whether an ML-based approach can bring advantages in terms of prediction accuracy, w.r.t. empirical models, using exactly the same inputs. We report in Tab. 1 the experiments' results. In the table, we use ML(JB08) to indicate the ML model that uses the same inputs as JB-08, and ML(NRLMSISE) for the one that uses the same inputs as NRLMSISE.

Category	NRLMSISE	JB08	ML(JB08)	ML(NRLMSISE)
Overall	63.18	40.71	24.81	24.63
200km-250km	10.24	14.75	7.60	8.17
250km-300km	26.22	22.30	9.32	11.00
300km-350km	65.61	29.72	12.53	11.38
350km-400km	35.07	30.38	13.36	14.05
400km-450km	69.33	39.99	24.19	22.24
450km-500km	48.99	27.14	18.20	19.06
500km-550km	150.48	97.29	67.94	65.01
Quiet (Ap 0-15)	69.77	43.47	26.60	25.84
Mild (Ap 15-30)	25.20	24.61	14.14	16.19
Minor (Ap 30-50)	20.97	23.81	13.94	19.38
Major (Ap 50+)	14.32	20.42	13.45	23.92

Table 1: MAPE results for the empirical vs ML models.

As we notice, ML-based approaches significantly outperform empirical models, at all altitudes and storm conditions. Interestingly, we also observe that for the major storm conditions, the model with the same inputs as JB-08 greatly outperforms the one of NRLMSISE-00 (i.e., 13.45% vs 23.92% MAPE). This is probably attributable at the use of the Dst index, which is not present in NRLMSISE-00. This allows the ML model to better describe the density variations during major storms.

4.2 Solar Irradiance Proxies Ablation Study

In Fig. 5, we show the change in MAPE due to the removal of each of the solar irradiance proxies considered (in this case F10.7, Y10.7, S10.7, M10.7). In the thermospheric density modeling community, the F10.7 index has historically been always used as a proxy for solar irradiance, since it is related to the Sun's electromagnetic spectrum at 10.7cm wavelength and can therefore be measured from the ground (since it is not absorbed in the atmosphere). In this experiment, we would like to see if the addition of any of those inputs has a significant influence on the thermospheric density prediction accuracy. For doing this, we remove each of them at a time, while keeping the others. A consistent increase in MAPE would suggest that the removed proxy was actually useful for predicting the thermospheric density, and should therefore be considered; while a negative or around zero MAPE difference, would suggest that the given proxy is not adding any new relevant information to the model, for improving its ability to predict the density. The results from this ablation study suggest that the S10.7 is the most important of the 10.7 indices, since its removal as an input to the model results in the largest increase in model error.

4.3 OMIWeb Inputs Addition

This experiment involves comparing the performance of a model trained on the JB-08 inputs versus exactly the same but including 2 days of high-resolution NASA OMNIWeb data, sampled hourly. The hypothesis was that by including more information about solar wind magnetic field and plasma data, will improve the model performance.

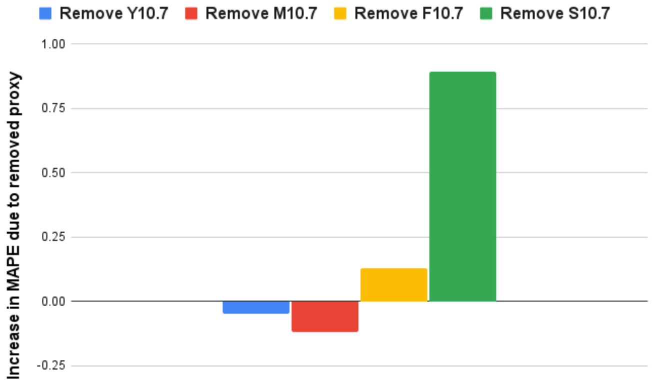

Fig. 5: Ablation study: Model MAPE performance drop removing 10.7 proxies

Category	ML(JB08)	ML(JB08) + OMNIWeb
Overall	24.81	21.32
200km-250km	7.60	6.36
250km-300km	9.32	7.01
300km-350km	12.53	10.43
350km-400km	13.36	11.09
400km-450km	24.19	21.27
450km-500km	18.20	15.17
500km-550km	67.94	60.27
Quiet (Ap 0-15)	26.60	23.01
Mild (Ap 15-30)	14.14	11.22
Minor (Ap 30-50)	13.94	11.33
Major (Ap 50+)	13.45	10.13

Table 2: MAPE results comparing an ML model trained on just JB08 inputs versus the inclusion of geomagnetic information from NASA OMNIWeb.

In Tab. 2, we show that the inclusion of NASA OMNIWeb data improves the model performance in all altitude ranges and geomagnetic conditions. This is not surprising as the JB-08 inputs use just the Dst and Ap indices to describe geomagnetic conditions, whereas the NASA OMNIWeb data includes information about the interplanetary magnetic field, the speed, and composition of the incoming solar wind, as well as various geomagnetic indices such as the AE, PC, and SYM/H indices.

4.4 FISM2 Stan Bands vs Solar Proxies

This section contains a direct comparison between a model using FISM2 stan bands versus a model using the irradiance proxies F10.7, M10.7, S10.7, Y10.7. The expectation was that the stan bands include more information about the incoming solar irradiance and therefore will improve model performance.

In Tab. 3, a noteworthy outcome is presented. On the whole, the employment of irradiance proxies exhibits superior performance compared to the FISM2 stan bands model. These findings imply that the utilization of 10.7 proxies yields comparable results to the FISM2 stan bands, albeit exclusively within the parameters defined by our model and experimental setup. Notably, the incorporation of FISM2 stan bands data necessitated the establishment of a defined temporal data ingestion window. Following preliminary experiments, we determined that a 24-hour lag for the FISM2 stan bands flare data serves as optimal input for the model. However, it is reasonable to anticipate that the inclusion of additional solar irradiance data will enhance density estimation beyond the use of proxies alone. Discrepancies could potentially arise due to disparities in model complexity or the selection of temporal lags. The integration of FISM2 stan bands data augments the volume of information that our models are required to process substantially, adding more complexity and potentially rendering them more susceptible to overfitting. Largely constrained by computational

Category	FISM2 stan bands	10.7 Proxies
Overall	21.78	21.32
200km-250km	9.85	6.36
250km-300km	7.77	7.01
300km-350km	10.21	10.43
350km-400km	11.30	11.09
400km-450km	21.09	21.27
450km-500km	14.78	15.17
500km-550km	60.81	60.27
Quiet (Ap 0-15)	23.29	23.01
Mild (Ap 15-30)	13.02	11.22
Minor (Ap 30-50)	11.93	11.33
Major (Ap 50+)	10.97	10.13

Table 3: MAPE results using FISM2 stan bands versus 10.7 proxies

limitations inherent in this study, we defer a more comprehensive exploration of this matter to future research. In the meantime, we believe that the software environment and the accessible data pipeline collectively offer a robust platform for benchmarking thermospheric density models, and for conducting analogous inquiries into the significance of solar irradiance inputs.

4.5 Physics Validation

In scientific ML models, it is important to test the soundness of the model with respect to known physics laws/behaviors. For instance, in the case of thermospheric density, we know that the day side of the thermosphere is hotter, which causes higher densities (and usually hottest in the sub-solar point), and the night side of the thermosphere is colder, which causes lower thermospheric density values. To confirm this, in Fig. 6, we show three different angle views for the same global plot of the ML model output for a fixed date (1st of January 2018, at midnight UTC). As we observe, the model correctly captures the day/night variations (the maximum densities occur around the sub-solar point) and manages to represent the thermospheric gradient caused by the temperature variations due to the day/night side transitions. This confirms to us that from thermospheric density measurements derived from satellite orbits (and not from global measurements at all longitudes and latitudes at each time), the model was able to learn and correctly represent the global thermospheric density variation patterns, in unseen geographical locations. We believe that the almost polar orbits of the satellites, combined with the time coverage of the measurements (which corresponds to almost two full solar cycles) have allowed the model to observe thermospheric density changes across a wide range of local solar times, latitudes and Sun conditions, therefore enhancing its generalization capabilities in geographical regions during times that were unseen during training.

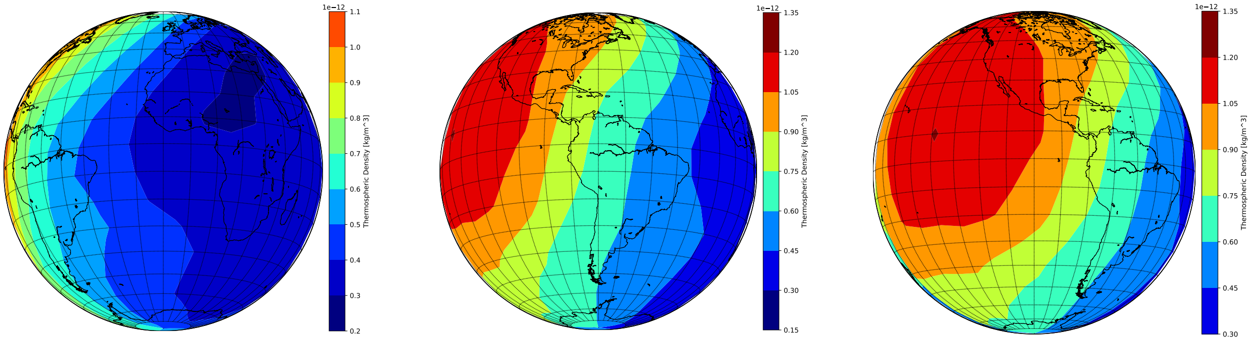

Fig. 6: Global thermospheric density plots produced through the ML model.

5. CONCLUSIONS

This study has led to the open-source release of Karman: a Python package for data-driven thermospheric density modelling using machine learning[6]. The contributions of the work can be summarized in three distint outcomes:

1. A pipeline for constructing a unified ML-ready data pipeline that contains irradiance inputs (both EUV irradiance and indices/proxies), geomagnetic inputs (both measurements and indices), empirical thermospheric density models inputs and values (for NRLMSISE-00 and JB-08), precise orbit determination-derived thermospheric density values, together with the corresponding position and times at which such measurements were derived (for the CHAMP, GRACE, GOCE, SWARM-A, SWARM-B, SWARM-C satellite missions). This is made available open-source, and in an analysis-ready fashion.
2. A framework for benchmarking density models, which allows a fair comparison between thermospheric density models. This framework enables future works to directly compare the performance of their models against the state-of-the-art empirical thermospheric density models, as well as the ML models developed in this study.
3. A series of experiments with informative outcomes for the community looking to develop thermospheric density models. Firstly, it establishes that ML models can outperform existing state-of-the-art empirical models (JB-08 and NRLMSISE-00) on exactly the same inputs. Secondly, our study compares the performance of a model trained using FISM2 stan bands data to a model trained on the 10.7 proxies. We find they perform similarly, with a slight edge to the model using the proxies. Then, an experiment studying the importance of NASA OMNIWeb inputs is also performed: this highlights that the addition of NASA OMNIWeb inputs can enhance the model prediction accuracy. Lastly, we showed that the ML models can effectively capture the physics behavior, by modeling the density gradients due to day/night variations.
4. Finally, this study introduces ML models for use in thermospheric neutral density prediction tasks: these are fast and accurate differentiable density models that can be easily integrated for downstream tasks (such as orbit propagation).

6. ACKNOWLEDGEMENTS

This work is the research product of the Benchmarking Data-Driven Thermospheric Density Estimation Models research. This has been funded and supported by the NASA heliophysics division, through a NASA sole source proposal and we would like to express our sincere gratitude. The research and its outputs have been designed, managed, and delivered by Trillium Technologies Inc (`https://trillium.tech`) Trillium is a research and development company with a focus on intelligent systems and collaborative communities for planetary stewardship, space exploration, and human health. The material is based upon work supported by NASA under award No NNX14AT27A. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Aeronautics and Space Administration (NASA). Trillium aspires to ensure that the latest tools and techniques in Artificial Intelligence (AI) and Machine Learning (ML) are applied to developing open science for all Humankind.

7. REFERENCES

[1] A BenMoussa, S Gissot, U Schühle, G Del Zanna, F Auchère, S Mekaoui, AR Jones, D Walton, CJ Eyles, Gérard Thuillier, et al. On-orbit degradation of solar instruments. *Solar Physics*, 288:389–434, 2013.

[2] TE Berger, MJ Holzinger, EK Sutton, and JP Thayer. Flying through uncertainty. *Space Weather*, 18(1):e2019SW002373, 2020.

[3] Stefano Bonasera, Giacomo Acciarini, J Pérez-Hernández, Bernard Benson, Edward Brown, Eric Sutton, Moriba K Jah, Christopher Bridges, and Atılım Günes Baydin. Dropout and ensemble networks for thermospheric density uncertainty estimation. In *Bayesian Deep Learning Workshop, NeurIPS*, 2021.

[6] `https://github.com/spaceml-org/karman`, date of access: August 2023.

[4] Bruce Bowman, W Kent Tobiska, Frank Marcos, Cheryl Huang, Chin Lin, and William Burke. A new empirical thermospheric density model jb2008 using new solar and geomagnetic indices. In *AIAA/AAS astrodynamics specialist conference and exhibit*, page 6438, 2008.

[5] Phillip C Chamberlin, Francis G Eparvier, Victoria Knoer, H Leise, Alicia Pankratz, Martin Snow, Brian Templeman, Edward Michael Benjamin Thiemann, Donald L Woodraska, and Thomas N Woods. The flare irradiance spectral model-version 2 (fism2). *Space Weather*, 18(12):e2020SW002588, 2020.

[6] Eelco Doornbos. *Thermospheric density and wind determination from satellite dynamics*. Springer Science & Business Media, 2012.

[7] John T Emmert. Thermospheric mass density: A review. *Advances in Space Research*, 56(5):773–824, 2015.

[8] Tzu-Wei Fang, Adam Kubaryk, David Goldstein, Zhuxiao Li, Tim Fuller-Rowell, George Millward, Howard J Singer, Robert Steenburgh, Solomon Westerman, and Erik Babcock. Space weather environment during the spacex starlink satellite loss in february 2022. *Space weather*, 20(11):e2022SW003193, 2022.

[9] T.R. George. The Use of Long Short-Term Memory Artificial Neural Networks for the Global Prediction of Atmospheric Density, 2020. MS Thesis, Unviersity of Kansas, Lawrence, KS.

[10] Changyong He, Yang Yang, Brett Carter, Emma Kerr, Suqin Wu, Florent Deleflie, Han Cai, Kefei Zhang, Luc Sagnières, and Robert Norman. Review and comparison of empirical thermospheric mass density models. *Progress in Aerospace Sciences*, 103:31–51, 2018.

[11] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[12] Fazlul I Laskar, Eric K Sutton, Dong Lin, Katelynn R Greer, Saurav Aryal, Xuguang Cai, Nicholas Michael Pedatella, Richard W Eastes, Wenbin Wang, Mihail V Codrescu, et al. Thermospheric temperature and density variability during 3 to 4 february 2022 minor geomagnetic storm: The spacex satellite loss event. *Authorea Preprints*, 2022.

[13] Richard J Licata and Piyush M Mehta. Uncertainty quantification techniques for data-driven space weather modeling: thermospheric density application. *Scientific Reports*, 12(1):7256, 2022.

[14] Richard J Licata, Piyush M Mehta, W Kent Tobiska, and S Huzurbazar. Machine-learned hasdm thermospheric mass density model with uncertainty quantification. *Space Weather*, 20(4):e2021SW002915, 2022.

[15] Piyush M Mehta and Richard Linares. A methodology for reduced order modeling and calibration of the upper atmosphere. *Space Weather*, 15(10):1270–1287, 2017.

[16] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[17] JM Picone, AE Hedin, D Pj Drob, and AC Aikin. Nrlmsise-00 empirical model of the atmosphere: Statistical comparisons and scientific issues. *Journal of Geophysical Research: Space Physics*, 107(A12):SIA–15, 2002.

[18] D. Pérez and R. Bevilacqua. Neural network based calibration of atmospheric density models. *Acta Astronautica*, 110:58–76, 2015.

[19] D. Pérez, B. Wohlberg, T.A. Lovell, and R. Bevilacqua. Orbit-centered atmospheric density prediction using artificial neural networks. *Acta Astronautica*, 98(1):9–23, May 2014.

[20] Liying Qian and Stanley C Solomon. Thermospheric density: An overview of temporal and spatial variations. *Space science reviews*, 168:147–173, 2012.

[21] AD Richmond, EC Ridley, and RG Roble. A thermosphere/ionosphere general circulation model with coupled electrodynamics. *Geophysical Research Letters*, 19(6):601–604, 1992.

[22] Christian Siemes, João de Teixeira da Encarnação, Eelco Doornbos, Jose Van Den Ijssel, Jiří Kraus, Radek Pereštỳ, Ludwig Grunwaldt, Guy Apelbaum, Jakob Flury, and Poul Erik Holmdahl Olsen. Swarm accelerometer data processing from raw accelerations to thermospheric neutral densities. *Earth, Planets and Space*, 68(1):1–16, 2016.

[23] Stanley C Solomon and Liying Qian. Solar extreme-ultraviolet irradiance for general circulation models. *Journal of Geophysical Research: Space Physics*, 110(A10), 2005.

[24] W Kent Tobiska, S Dave Bouwer, and Bruce R Bowman. The development of new solar indices for use in thermospheric density modeling. *Journal of Atmospheric and Solar-Terrestrial Physics*, 70(5):803–819, 2008.

[25] Herbert Turner, Maggie Zhang, David Gondelach, and Richard Linares. Machine learning algorithms for improved thermospheric density modeling. In Frederica Darema, Erik Blasch, Sai Ravela, and Alex Aved, editors, *Dynamic Data Driven Applications Systems*, pages 143–151, Cham, 2020. Springer International Publishing.

[26] A Vourlidas and S Bruinsma. Euv irradiance inputs to thermospheric density models: Open issues and path

forward. *Space Weather*, 16(1):5–15, 2018.

[27] Libin Weng, Jiuhou Lei, Jiahao Zhong, Xiankang Dou, and Hanxian Fang. A machine-learning approach to derive long-term trends of thermospheric density. *Geophysical Research Letters*, 47(6):e2020GL087140, 2020.