www.ijbar.org
ISSN2249-3352(P)2278-0505(E)
CosmosImpactFactor-5.86

AI Tool for Modelling Satellite Expected Lifetime for ISRO's Space Missions

Y.Anil Kumar[1*],Guduru Chaitanya[2],Janagama Jala[2], Bolugoddu Neeraj Varma[2], Engu Naveen[2]

[1]Assistant Professor, [2]UG Student, [1,2]Department of Computer Sciences and Engineering,

[1,2]Vaagdevi College of Engineering (UGC - Autonomous), Bollikunta, Warangal, Telangana.

[*]Corresponding author: Y.Anil Kumar (anilkumar_y@vaagdevi.edu.in)

ABSTRACT

The Indian Space Research Organisation (ISRO) has been a pioneer in space exploration since its establishment in 1969. Over the decades, it has developed numerous satellite missions that have significantly contributed to various fields, including communication, earth observation, and navigation. Historically, predicting the expected lifetime of satellites has relied on extensive empirical data and expert assessments. ISRO has continuously innovated, transitioning from traditional methods to more sophisticated approaches, integrating advances in technology and science.The objective of this research is to leverage machine learning techniques to model and predict the expected lifetime of satellites in ISRO's space missions. By utilizing historical data and advanced analytics, the goal is to enhance the accuracy of lifetime predictions, thereby improving mission planning, resource allocation, and operational efficiency. Before the advent of machine learning, traditional methods for predicting satellite lifetime typically involved empirical models based on historical performance data, physical failure models relying on component testing, and expert evaluations through qualitative assessments. These methods often lacked the ability to efficiently analyze large datasets and were limited in their predictive capabilities.The traditional methods for estimating satellite lifetime at ISRO are often time-consuming, reliant on expert knowledge, and limited by the availability of historical data, leading to inaccuracies in lifetime predictions and inefficient resource management during missions.The proposed system involves collecting extensive datasets from past ISRO satellite missions, including operational data, environmental factors, and component performance metrics. This data will be pre-processed and analyzed to identify patterns and correlations. A series of predictive models will be developed to estimate satellite lifetimes, incorporating various features such as launch conditions, operational anomalies, and degradation trends. The system will also include visualization tools to present insights and predictions in a user-frsiendly manner for mission planners.

Keywords: Satellite Lifetime Prediction, Machine Learning, ISRO, Space Missions, Predictive Modeling, Operational Efficiency

1. INTRODUCTION

The Indian Space Research Organisation (ISRO) has established itself as a major player in global space exploration. Since its founding in 1969, ISRO has launched numerous satellite missions that have revolutionized fields such as communications, earth observation, meteorology, navigation, and scientific research. Some of ISRO's most significant missions include the Mars Orbiter Mission (Mangalyaan), Chandrayaan lunar missions, and the launch of the Polar Satellite Launch Vehicle (PSLV). With over 300 satellites launched and partnerships with multiple countries, ISRO has built a reputation for cost-effective and efficient space exploration. Predicting the lifespan of satellites has traditionally relied on expert judgment and empirical data. However, as satellite technology advances, the need for more precise predictions is growing. Machine learning offers an opportunity to enhance these predictions by analyzing large datasets to model and predict satellite lifetimes with greater accuracy. Applications of such a model include mission planning, risk management, and optimization of satellite resources to ensure maximum operational efficiency. Before the integration of machine learning techniques, predicting the lifetime of satellites involved using traditional methods, such as empirical models based on historical performance data and expert evaluations. These methods, while useful, often struggled with scalability and lacked precision. For instance, failure models depended heavily on physical testing of components and extrapolating results based on small sample sizes. In addition, expert assessments, while valuable, were subjective and often limited by human bias and the availability of historical data. These methods were time-consuming, error-prone, and often unable to capture complex relationships between operational data and satellite degradation patterns. The inability to analyze large datasets efficiently led to inaccuracies in lifetime predictions, which could, in turn, affect resource allocation, mission planning, and satellite operations.

2. LITERATURE SURVEY

Williams and Bell [1] analyzed the Chang'e 5 mission, discussing its objectives, lunar sampling techniques, and technological innovations. They examined the spacecraft's performance and mission outcomes in the context of China's lunar exploration program. Dunbar [2] presented an overview of NASA's Artemis program, focusing on its objectives, planned lunar missions, and technological advancements. The study highlighted Artemis' role in future lunar exploration and its significance in human spaceflight. Dobrijevic [3] provided a comprehensive guide to NASA's DART mission, explaining its purpose, execution, and impact on planetary defense strategies. The study discussed how the mission tested kinetic impact technology for asteroid deflection. Williams and Bell [4] described Chandrayaan 3, outlining its scientific objectives, technological advancements, and expected contributions to lunar exploration. The mission aimed to improve India's lunar research capabilities. Williams and Bell [5] examined the Luna 25 mission, detailing its goals, spacecraft design, and expected contributions to Russia's lunar exploration efforts. The study explored its significance in the country's space ambitions. Ebeling [6] introduced fundamental concepts of reliability and maintainability engineering, discussing various analytical techniques for assessing system performance and longevity. The book emphasized the importance of reliability in engineering design. Huangpeng et al. [7] proposed a methodology for determining the optimal sample size for launch vehicle reliability analysis. They utilized Sequential Probability Ratio Test (SPOT) and Bayesian recursive estimation to improve the accuracy of reliability predictions. Their study aimed to enhance launch vehicle assessment through statistical modeling. Krevor and Wilhite [8] introduced a framework for estimating the cost of improving launch vehicle reliability. They analyzed cost trade-offs associated with enhancing reliability and presented a model to optimize investment in failure prevention measures. Their findings contributed to cost-effective launch vehicle development strategies. Guarro [9] conducted an in-depth assessment of space launch vehicle reliability, evaluating various statistical methods for estimating failure probabilities. His study highlighted the challenges in predicting launch success and emphasized the importance of historical data analysis in improving launch outcomes. Guikema and Paté-Cornell [10] applied Bayesian analysis to assess launch vehicle success rates, integrating historical data with probabilistic models. Their research demonstrated how Bayesian updating can improve predictions of launch reliability and reduce uncertainty in failure assessments. Guikema and Paté-Cornell [11] investigated the probability of infancy-related failures in space launch vehicles, emphasizing the role of early-life performance in determining overall reliability. They found that launch vehicles often exhibit higher failure rates in initial flights before stabilizing in performance. Castet and Saleh [12] analyzed satellite reliability using statistical methods, comparing various reliability estimation techniques. Their study provided insights into the failure rates of different satellite classes and contributed to the understanding of satellite longevity in space missions.

Castet and Saleh [13] extended their analysis to satellite subsystems, examining reliability trends and identifying common failure modes. They proposed a statistical modeling approach to improve reliability predictions for complex satellite architectures. Grile and Bettinger [14] estimated the reliability of satellites launched between 1991 and 2020 using statistical methods. They identified trends in satellite failures over time and provided an analysis of factors affecting satellite longevity in orbit. Dubos et al. [15] explored the relationship between satellite mass and reliability, analyzing whether spacecraft size influences failure rates. Their findings suggested that larger satellites tend to have higher reliability, potentially due to more robust design and testing procedures.

3.PROPOSED SYSTEM

Step 1: Satellite Lifetime Dataset: The first step involves obtaining the satellite lifetime dataset, which contains various features related to satellite characteristics and their expected lifetime in space missions. This dataset is crucial for training and testing machine learning models aimed at predicting satellite lifespan. It is typically a CSV file that includes features such as satellite name, launch details, orbit type, dry mass, power capacity, and expected lifetime in years.

Step 2: Data Preprocessing: The next step is to preprocess the data for use in machine learning models. This includes handling missing data by filling in the missing values for certain features like "Dry Mass (kg.)" and "Power (watts)" using the mean of the respective columns. Date columns such as "Date of Launch" are converted into datetime format, and additional columns like "Day", "Month", and "Year" are extracted from the launch date. The dataset is then rescaled using StandardScaler to ensure all features are on a similar scale, and categorical features are encoded using LabelEncoder. The data is split into training and test sets for model evaluation.

Step 3: Exploratory Data Analysis (EDA) Plots: In this step, various exploratory data analysis (EDA) plots are generated to better understand the dataset. Histograms of the features and the target variable "Expected Lifetime (yrs.)" are created to visualize distributions. A correlation heatmap is also generated to identify potential relationships between features, which may help in model selection and refinement.

Step 4: Existing Ridge Regressor Algorithm: The Ridge Regressor algorithm is applied to the preprocessed data as one of the traditional machine learning models for regression. The Ridge model is trained using the training data and evaluated using the test data. The performance of the model is then assessed through regression metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared (R^2). If a pre-trained model exists, it is loaded from disk to predict the satellite lifetime on new data.

Step 5: Existing Linear Regressor Algorithm: Similar to Ridge Regressor, the Linear Regression model is applied to the data to test its performance. The Linear Regressor is trained on the same training data, and predictions are made for the test set. The model's performance is then evaluated using the same set of regression metrics. As with Ridge Regressor, if a pre-trained model exists, it is loaded to predict on new test data.

Step 6: Proposed LSTM Regressor Algorithm: A more advanced approach is implemented using Long Short-Term Memory (LSTM) networks, which are a type of recurrent neural network (RNN) well-suited for sequential data. In this step, the data is reshaped to be compatible with the LSTM architecture. The LSTM model is either trained from scratch or loaded if pre-trained. This model is then used to predict the expected satellite lifetime, and performance is evaluated using the same regression metrics as the traditional models.

Step 7: Performance Comparison Graph: A comparison graph is generated to visualize the performance of all three models—Ridge Regressor, Linear Regressor, and LSTM Regressor. The graph displays the performance metrics such as MAE, MSE, RMSE, and R^2 score for each model, allowing for an easy comparison of their accuracy and reliability in predicting satellite lifetimes. This helps in identifying which algorithm performs best.

Step 8: Prediction of Output from Test Images with LSTM Regressor Algorithm

Finally, the trained LSTM model is used to make predictions on new test data. The test data is preprocessed in the same way as the training data, ensuring consistency. Once the data is prepared, the LSTM model predicts the expected satellite lifetime, and the results are displayed. A plot is generated to visualize the predicted lifetimes across different test samples. This step concludes the research, where the model's ability to predict real-world satellite lifetimes is demonstrated.

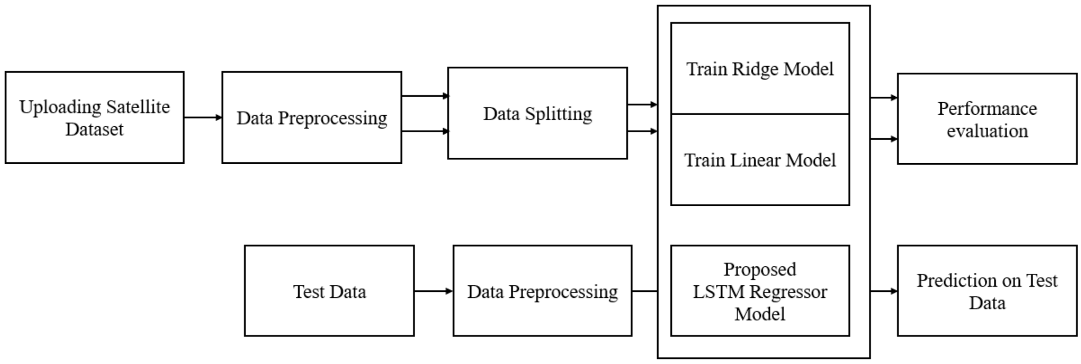

Fig.1: Architectural block diagram of proposed system.

3.2 Data Preprocessing and Data Splitting

Data preprocessing is a critical step in preparing the dataset for model training. In this research, the dataset is primarily composed of satellite-related features that are used to predict the expected lifetime of satellites. The process involves several key tasks:

1. **Date and Time Feature Engineering**:
   - The dataset contains a column titled "Date of Launch", which represents the launch date of the satellite. This feature is converted to a datetime format using pd.to_datetime().
   - After conversion, several date components are extracted, such as **Day**, **Month**, and **Year**, to make the information more usable for the model. These components are stored as separate columns and the original "Date of Launch" column is dropped since it is no longer necessary.
2. **Label Encoding of Categorical Variables**:
   - The dataset also includes several categorical variables, such as satellite names, country, orbit type, etc. These variables need to be converted into numeric representations before feeding them into machine learning models.

- The LabelEncoder from scikit-learn is used to transform each categorical column into a numeric value. This is necessary because most machine learning models work with numerical data. All categorical columns, such as the "Name of Satellite" and "Launch Vehicle", are encoded in this way.

3. **Handling Missing Values**:
    - Several columns in the dataset, such as "Dry Mass (kg.)", "Power (watts)", and "Expected Lifetime (yrs.)", may contain missing values. In this research, missing values in these columns are filled with the **mean** value of the respective column. This technique helps maintain data integrity without losing rows due to missing information.
4. **Resampling the Dataset**:
    - Resampling is performed to balance the dataset and prevent biases toward any particular group of data. In this research, a subset of the data is resampled to generate additional data points (5000 in total). This helps in improving the model's performance by ensuring a more uniform distribution of training data.
5. **Feature Scaling**:
    - Since many machine learning algorithms are sensitive to the scale of the data, the features are standardized using the StandardScaler. This scales the data such that it has a mean of 0 and a standard deviation of 1. Scaling is essential for models that rely on distance-based calculations, such as linear regression or neural networks.

3.3 ML Model Building

LSTM Regressor

What is LSTM Regression? Long Short-Term Memory (LSTM) is a specialized type of Recurrent Neural Network (RNN) that is well-suited for learning from sequences of data, such as time series or sequential data. Unlike regular RNNs, LSTMs are capable of capturing long-range dependencies due to their unique architecture, which helps in solving the vanishing gradient problem faced by standard RNNs.

An **LSTM Regressor** is a regression model that leverages the LSTM architecture to predict a continuous target variable. This is particularly useful when the target variable depends on sequential data, such as time series data, where the future values are influenced by past observations.

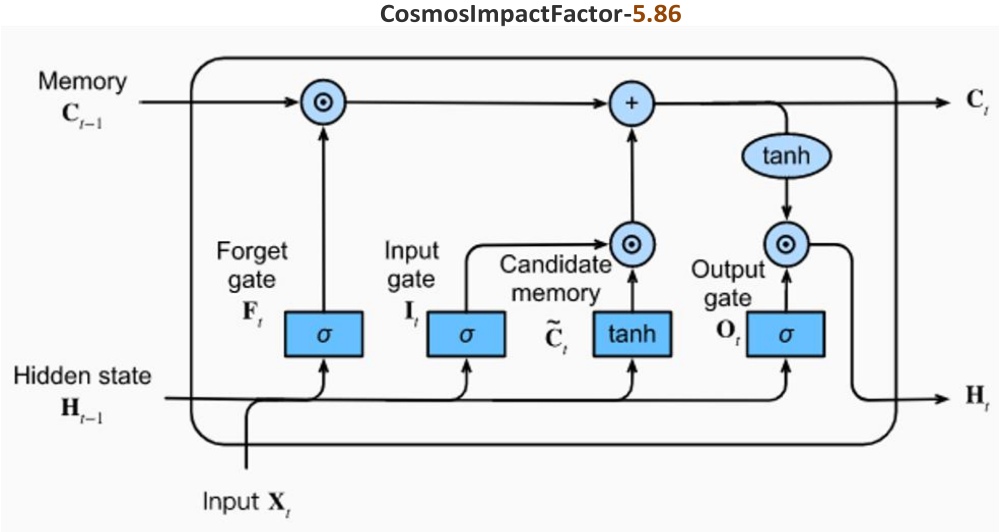

Fig 2 : LSTM flow diagram.

How it Works:

- LSTM uses a set of gates (input, forget, and output gates) to control the flow of information through the network. These gates allow LSTMs to remember information for long periods, making them suitable for tasks where time-dependent patterns need to be learned.
- The architecture of an LSTM cell is designed to maintain information in memory, which helps the model focus on important patterns and ignore irrelevant ones.
- The LSTM Regressor typically consists of:
  1. **Input Layer**: The sequence data is fed into the LSTM layers.
  2. **LSTM Layers**: One or more LSTM layers are used to capture temporal dependencies.
  3. **Dense Layer**: After the LSTM layers, a dense layer is used to predict the target value.
  4. **Output Layer**: The output layer generates the final regression prediction.

4. RESULT AND DISCUSSION

4.1 Dataset Description

The dataset consists of various attributes related to satellites, with each row representing a different satellite. The columns have detailed information about the satellite's characteristics, operational parameters, and mission details. Here's a description of each column:

- **Name of Satellite**: The official name of the satellite, which serves as its primary identifier.
- **Alternate Names**: Other names or identifiers the satellite may be referred to by, possibly due to different naming conventions across organizations or countries.
- **Current Official Name of Satellite**: The most recent and officially recognized name of the satellite.

- **Country/Org of UN Registry**: The country or organization under which the satellite is registered with the United Nations. This helps identify the official ownership or control of the satellite at an international level.
- **Country of Operator/Owner**: The country that operates or owns the satellite, often distinct from the registration country.
- **Operator/Owner**: The organization or agency responsible for operating and maintaining the satellite. This could be a government agency or private entity.
- **Users**: The various entities or sectors that utilize the satellite's services, which could include governments, research institutions, or commercial users.
- **Purpose**: A broad classification of the satellite's main mission or functionality, such as communications, Earth observation, or navigation.
- **Detailed Purpose**: A more specific explanation of the satellite's intended mission or objectives, providing deeper insight into the satellite's operational goals.
- **Class of Orbit**: The classification of the orbit the satellite is placed in, such as Low Earth Orbit (LEO), Medium Earth Orbit (MEO), or Geostationary Orbit (GEO).
- **Type of Orbit**: The exact type or path of the orbit, such as circular, elliptical, or sun-synchronous.
- **Longitude of GEO (degrees)**: For satellites in Geostationary Orbit, this column represents the longitudinal position of the satellite in relation to the Earth's surface, measured in degrees.
- **Perigee (km)**: The closest point in the satellite's orbit to Earth, measured in kilometers. This is relevant for satellites in elliptical orbits.
- **Apogee (km)**: The farthest point in the satellite's orbit from Earth, measured in kilometers. This is relevant for satellites in elliptical orbits.
- **Eccentricity**: A measure of the deviation of the satellite's orbit from a perfect circle. It indicates how elongated the orbit is.
- **Inclination (degrees)**: The angle between the satellite's orbital plane and the Earth's equator, measured in degrees.
- **Period (minutes)**: The time it takes for the satellite to complete one full orbit around Earth, measured in minutes.
- **Launch Mass (kg.)**: The total mass of the satellite at the time of launch, measured in kilograms. This includes the satellite itself and any attached payloads.
- **Dry Mass (kg.)**: The mass of the satellite without fuel or any consumables, measured in kilograms.
- **Power (watts)**: The amount of electrical power the satellite generates, usually provided by solar panels, measured in watts.

- **Date of Launch**: The exact date when the satellite was launched into space.
- **Expected Lifetime (yrs.)**: The estimated operational lifetime of the satellite in years, based on factors like fuel capacity, technological lifespan, and mission design.
- **Contractor**: The company or organization responsible for manufacturing the satellite and possibly conducting initial operations.
- **Country of Contractor**: The country in which the contractor (satellite manufacturer) is based.
- **Launch Site**: The location from which the satellite was launched, which could include spaceports or launch facilities.
- **Launch Vehicle**: The rocket or space vehicle used to launch the satellite into orbit.
- **COSPAR Number**: The unique identifier assigned to the satellite by the Committee on Space Research (COSPAR) to track and catalog space missions.
- **NORAD Number**: The identification number assigned to the satellite by the North American Aerospace Defense Command (NORAD), used for tracking objects in space.

4.2 Results Description

This figure illustrates the graphical user interface (GUI) that enables the user to upload the satellite lifetime dataset. It displays how the dataset is loaded into the system for further analysis and processing. The interface provides an organized structure for selecting the dataset and triggers the initiation of the subsequent analysis steps. This visual emphasizes the simplicity and user-friendliness of the interface for users to interact with the satellite data, making it easier to proceed with preprocessing and model building tasks.

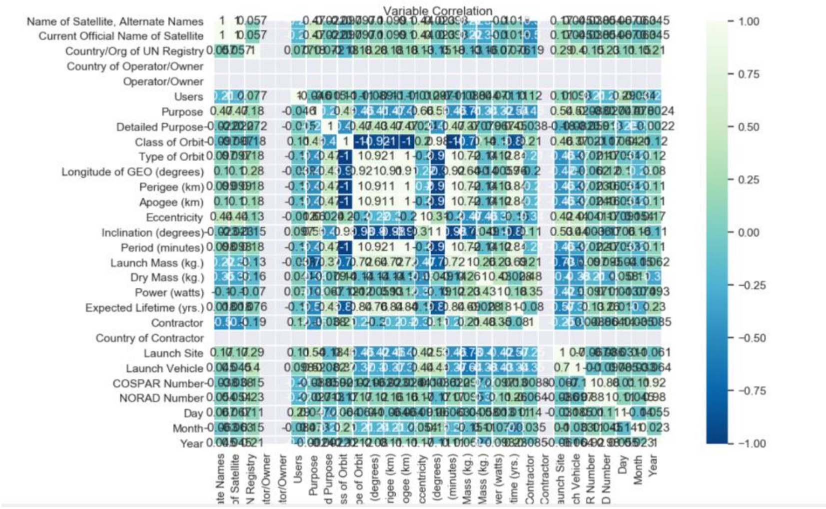

Fig. 3: EDA Plots of the Project

In this figure, the exploratory data analysis (EDA) plots are showcased, highlighting the relationship between the various features in the satellite dataset. These plots include visualizations such as histograms, scatter plots, and box plots that give an initial overview of the dataset's distribution, correlations, and potential outliers. The goal of EDA is to understand the characteristics of the dataset and to identify patterns or anomalies that could influence the model-building process, helping guide data preprocessing decisions.

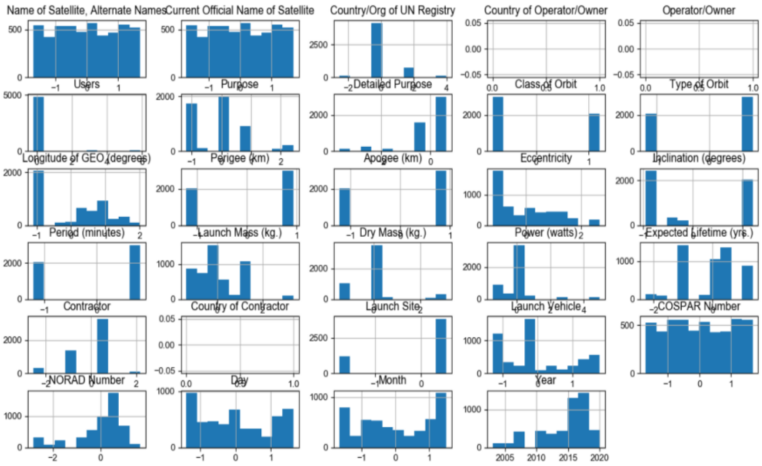

Fig. 4: Data Preprocessing

This figure presents the data preprocessing phase within the GUI interface. It visualizes the steps where the raw dataset is cleaned, missing values are handled, categorical data is encoded, and the dataset is transformed into a format suitable for machine learning models. It also highlights how feature scaling, such as normalization or standardization, is applied to prepare the data for better model performance. The preprocessing steps are displayed interactively within the interface, ensuring that users can track and manage each stage of data transformation.

The performance metrics for the Ridge Regressor model are displayed in the following ways:

- Mean Absolute Error (MAE): 1.259. This indicates the average magnitude of error in the model’s predictions, suggesting that on average, the model’s predictions are off by approximately 1.26 years.
- Mean Squared Error (MSE): 2.356. This metric represents the squared average difference between the predicted and actual values, reflecting the model’s prediction accuracy.
- Root Mean Squared Error (RMSE): 1.535. RMSE is the square root of MSE, offering an error metric in the same unit as the target variable, which is satellite lifetime in years.
- R-squared (R^2): 0.847. This indicates that 84.7% of the variance in satellite lifetime is explained by the model, suggesting a good fit to the data.

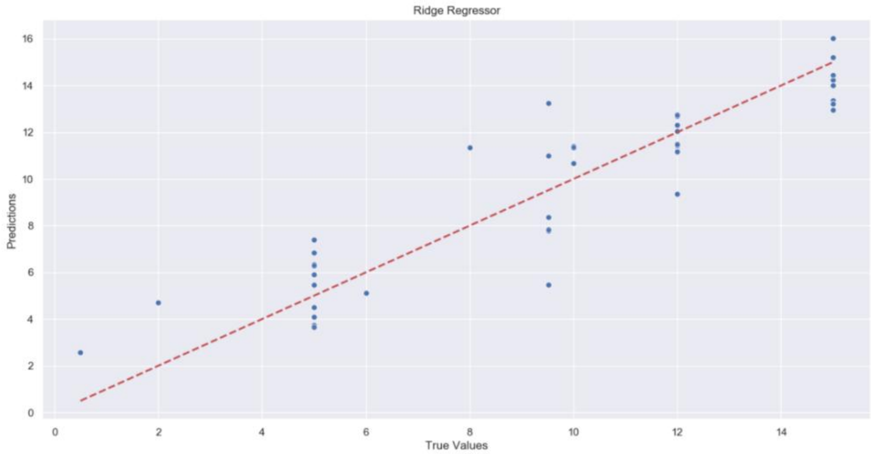

Fig. 4: Performance Metrics and Regression Scatter Plot of the Ridge Regressor Model

This figure presents the performance metrics along with a regression scatter plot for the Ridge Regressor model. The plot visually compares the predicted values against the actual satellite lifetimes. The closeness of the points to the diagonal line indicates the accuracy of the predictions. The performance metrics provided further quantify the model's prediction accuracy, giving a clear picture of how well the Ridge Regressor performs.

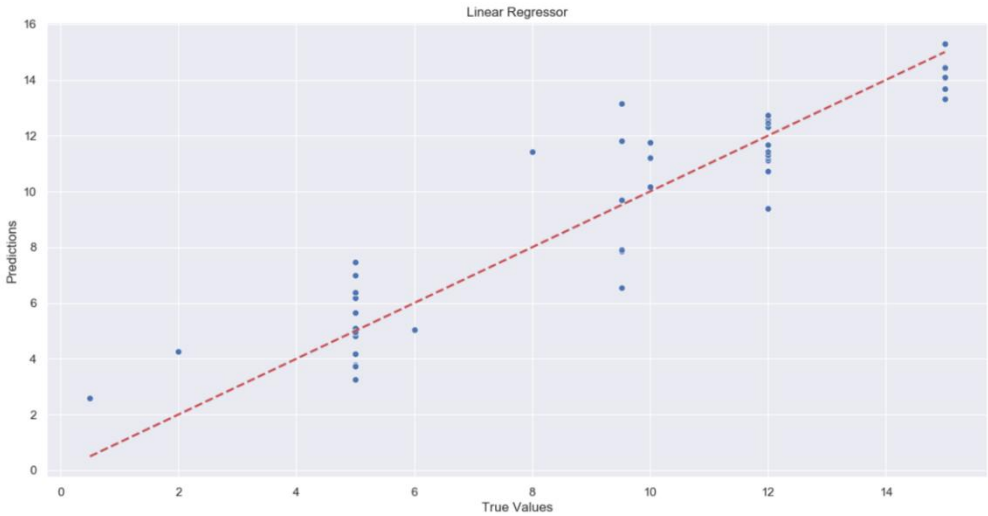

Fig. 5: Performance Metrics and Regression Scatter Plot of the Linear Regressor Model

Similar to Fig. 5, this figure shows the performance metrics and regression scatter plot for the Linear Regressor model. The scatter plot highlights how the model's predicted satellite lifetimes compare to the actual values. The performance metrics indicate that the Linear Regressor model outperforms the Ridge Regressor, as seen in the lower error metrics and higher R^2 value.

The performance metrics for the Linear Regressor model are:

- Mean Absolute Error (MAE): 1.197. The predictions are off by an average of about 1.2 years, slightly better than the Ridge Regressor.
- Mean Squared Error (MSE): 2.174. The squared errors for the predictions are slightly lower than those of the Ridge Regressor.

www.ijbar.org
ISSN2249-3352(P)2278-0505(E)
CosmosImpactFactor-5.86

- Root Mean Squared Error (RMSE): 1.475. This value is lower than the Ridge Regressor's RMSE, indicating that the Linear Regressor model performs better in terms of prediction accuracy.
- R-squared (R^2): 0.859. With 85.9% of variance explained, the Linear Regressor has a marginally better fit compared to the Ridge Regressor.

The performance metrics for the LSTM Regressor model are:

- Mean Absolute Error (MAE): 0.072. This indicates a very low average error in predictions, meaning the LSTM model predicts satellite lifetimes with high precision.
- Mean Squared Error (MSE): 0.008. The small value of MSE suggests that the model's predictions are very close to the actual values.
- Root Mean Squared Error (RMSE): 0.089. The RMSE value is also very low, supporting the model's ability to make accurate predictions.
- R-squared (R^2): 0.999. The LSTM model explains 99.9% of the variance in the satellite lifetime data, which indicates an excellent fit and extremely accurate predictions.

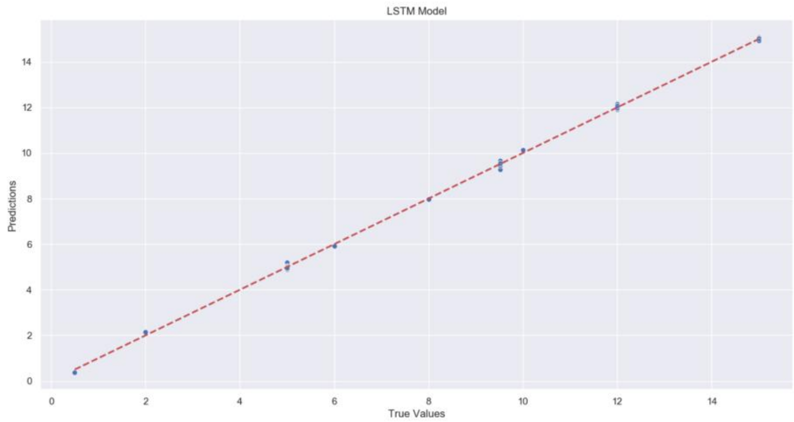

Fig. 6: Performance Metrics and Regression Scatter Plot of the LSTM Regression Model

This figure shows the performance metrics and regression scatter plot for the LSTM Regressor model. The regression plot illustrates how closely the model's predictions align with the actual satellite lifetimes, showcasing the superior performance of the LSTM model. Given the very low error metrics and the high R^2 score, this plot emphasizes the strong predictive power of the LSTM model.

www.ijbar.org
ISSN2249-3352(P)2278-0505(E)
CosmosImpactFactor-5.86

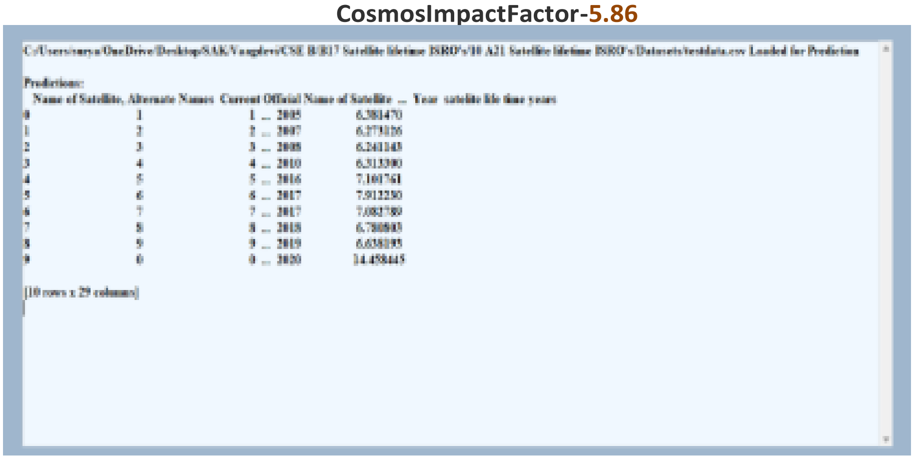

Fig.7: Model prediction on the test data.

Figure 7 shows the model's predictions applied to the test dataset. It visually represents the predicted satellite lifetimes versus the actual lifetimes, demonstrating how accurately the model performs on unseen data. The test data results are essential for validating the model's ability to generalize and make accurate predictions beyond the training set.

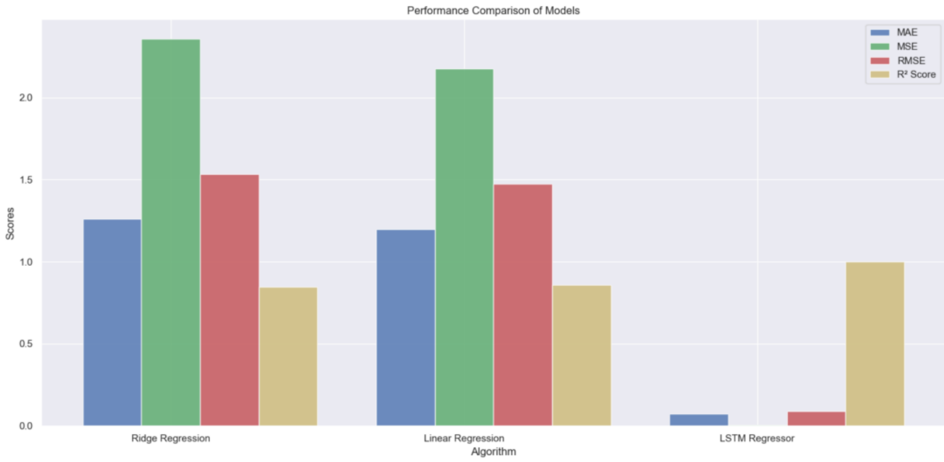

Fig. 8: Performance comparison graph of all models

Figure 8 presents a comparative graph of the performance metrics for all three models: Ridge Regressor, Linear Regressor, and LSTM Regressor. The graph shows how the models perform relative to each other in terms of MAE, MSE, RMSE, and R^2. It clearly illustrates that the LSTM model outperforms both Ridge and Linear Regression models, confirming its superior ability to predict satellite lifetimes accurately.

5.CONCLUSION

The research on predicting the expected lifetime of satellites using machine learning models aims to leverage satellite-related data for more accurate forecasting and analysis. By using a variety of features such as satellite mass, power, orbit parameters, and launch details, the research provides valuable insights into satellite longevity. The use of both traditional machine learning algorithms like Linear Regression and Ridge Regression, along with more advanced models such as LSTM Regressors, demonstrates the ability to handle complex data patterns and temporal dependencies in predicting satellite lifetimes. The successful application of these models can significantly improve satellite management and space mission planning, offering benefits such as more efficient resource allocation, reduced costs, and optimized satellite usage for various purposes.

REFERENCES

[1] Williams D. and Bell E., "Chang'e 5," 2023, https://nssdc.gsfc.nasa.gov/nmc/spacecraft/display.action?id=2020-087A.

[2] Dunbar B., "Artemis," 2023, https://www.nasa.gov/specials/artemis.

[3] Dobrijevic D., "NASA's Dart Asteroid-Smashing Mission: The Ultimate Guide," 2022, https://www.space.com/dart-asteroid-mission.

[4] Williams D. and Bell E., "Chandrayaan 3," 2023, https://nssdc.gsfc.nasa.gov/nmc/spacecraft/display.action?id=CHANDRYN3.

[5] Williams D. and Bell E., "Luna 25," 2023, https://nssdc.gsfc.nasa.gov/nmc/spacecraft/display.action?id=LUNA-25.

[6] Ebeling C. R., *An Introduction to Reliability and Maintainability Engineering*, 2nd ed., Waveland Press, Long Grove, IL, 2010.

[7] Huangpeng Q., Duan X., Zhang Y., and Huang W., "Sample Size Design of Launch Vehicle Based on SPOT and Bayesian Recursive Estimation," 41st Chinese Control Conference (CCC), IEEE Publ., Piscataway, NJ, July 2022, pp. 1373–1377. https://doi.org/10.23919/CCC55666.2022.9901630

[8] Krevor Z. C. and Wilhite A., "An Approach for Calculating the Cost of Launch Vehicle Reliability," AIAA SPACE Conference & Exposition, AIAA Paper 2007-9920, Sept. 2007. https://doi.org/10.2514/6.2007-9920

[9] Guarro S., "On the Estimation of Space Launch Vehicle Reliability," *International Journal of Performability Engineering*, Vol. 9, No. 6, 2013, p. 619. https://doi.org/10.23940/ijpe.13.6.p619.mag

[10] Guikema S. D. and Paté-Cornell M. E., "Bayesian Analysis of Launch Vehicle Success Rates," *Journal of Spacecraft and Rockets*, Vol. 41, No. 1, 2004, pp. 93–102. https://doi.org/10.2514/1.9268

[11] Guikema S. D. and Paté-Cornell M. E., "Probability of Infancy Problems for Space Launch Vehicles," *Reliability Engineering and System Safety*, Vol. 87, No. 3, 2005, pp. 303–314. https://doi.org/10.1016/j.ress.2004.06.001

[12] Castet J.-F. and Saleh J. H., "Satellite Reliability: Statistical Data Analysis and Modeling," *Journal of Spacecraft and Rockets*, Vol. 46, No. 5, 2009, pp. 1065–1076. https://doi.org/10.2514/1.42243

[13] Castet J.-F. and Saleh J. H., "Satellite and Satellite Subsystems Reliability: Statistical Data Analysis and Modeling," *Reliability Engineering and System Safety*, Vol. 94, No. 11, 2009, pp. 1718–1728. https://doi.org/10.1016/j.ress.2009.05.004

[14] Grile T. M. and Bettinger R. A., "Statistical Reliability Estimation of Satellites Operating from 1991–2020," *AIAA SciTech Forum and Exposition*, AIAA Paper 2023-2372, Jan. 2023. https://doi.org/10.2514/6.2023-2372

[15] Dubos G. F., Castet J.-F., and Saleh J. H., "Statistical Reliability Analysis of Satellites by Mass Category: Does Spacecraft Size Matter?" *Acta Astronautica*, Vol. 67, Nos. 5–6, 2010, pp. 584–595. https://doi.org/10.1016/j.actaastro.2010.04.017

[16] Langer M. and Bouwmeester J., "Reliability of CubeSats—Statistical Data, Developers' Beliefs and the Way Forward," 30th Annual AIAA/USU Conference on Small Satellites, Paper SSC16-X-2, Aug. 2016, pp. 1–12.

[17] Edwards R., Smith C. A., and Whitley S., "Reliability for a 50+ Year Mission—Interstellar Probe," 2021 *Annual Reliability and Maintainability Symposium (RAMS)*, IEEE, New York, May 2021, pp. 1–8. https://doi.org/10.1109/RAMS48097.2021.9605752

[18] Castet J.-F. and Saleh J. H., "Beyond Reliability, Multi-State Failure Analysis of Satellite Subsystems: A Statistical Approach," *Reliability Engineering and System Safety*, Vol. 95, No. 4, 2010, pp. 311–322. https://doi.org/10.1016/j.ress.2009.11.001

[19] "Seradata," https://spacetrak.seradata.com/ [retrieved 1 June 2023].

[20] Grush L., "Elon Musk's Tesla Overshot Mars' Orbit, But It Won't Reach the Asteroid Belt as Claimed," 2018, https://www.theverge.com/2018/2/6/16983744/spacex-tesla-falcon-heavy-roadster-orbit-asteroid-belt-elon-musk-mars.

[21] Greshko M., "India's First Lunar Lander Falls Silent Just Before Touchdown," 2023, https://www.nationalgeographic.com/science/article/india-chandrayaan-2-landing-attempt-moon-lunar-south-pole.

[22] Grile T. M. and Bettinger R. A., "Reliability Analysis of Deep Space Satellites Launched 1991–2020: Bulk Population and Deployable Satellite Performance Analysis," *Quality and Reliability Engineering International*, 2024 (to be published).

[23] Guo J., Monas L., and Gill E., "Statistical Analysis and Modelling of Small Satellite Reliability," *Acta Astronautica*, Vol. 98, May–June 2014, pp. 97–110. https://doi.org/10.1016/j.actaastro.2014.01.018

[24] Perumal R. P., Voos H., Vedova F. D., and Moser H., "Small Satellite Reliability: A Decade in Review," 35th Annual AIAA/USU Conference on Small Satellites, Aug. 2021, Paper SSC21-WKIII-02, pp. 1–12.