SAFE LAND AI: AN IMPROVED SYSTEM FOR HARD LANDING PREDICTION AND PILOT ASSISTANCE USING ENSEMBLE MODEL IN COMMERCIAL FLIGHTS

Dr. K Kranthi kumar, Associate Professor, Department of IT - Sreenidhi Institute of Science and Technology, Ghatkesar, Hyderabad, Telangana.
Dr. Sreenivas Mekala , Associate Professor, Department of IT - Sreenidhi Institute of Science and Technology, Ghatkesar, Hyderabad, Telangana.
Chindi Praneeth kumar , Student of Information Technology, Sreenidhi Institute of Science and Technology, Ghatkesar, Hyderabad, Telangana.
Voggu Chandrashekar , Student of Information Technology, Sreenidhi Institute of Science and Technology, Ghatkesar, Hyderabad, Telangana.
Chinta Rajashekar, Student of Information Technology, Sreenidhi Institute of Science and Technology, Ghatkesar, Hyderabad, Telangana.

ABSTRACT:
Hard landings in commercial aviation pose significant safety risks, contributing to structural damage, increased maintenance costs, and passenger discomfort. Safe Land AI introduces an innovative machine learning-based system to predict and mitigate hard landings during the approach phase of commercial flights. By employing an ensemble model combining XGBoost and Extra Trees classifiers, the system achieves high predictive accuracy on a diverse dataset of 30,000 flight records, incorporating flight parameters and weather conditions. Integrated with SHAP for transparent feature insights and a web-based interface for real-time pilot assistance, Safe Land AI provides actionable recommendations, such as adjusting descent rates or flare angles, to prevent hard landings. Extensive testing demonstrates its superiority over existing systems, offering enhanced generalizability, early risk detection, and intuitive pilot support. Safe Land AI holds transformative potential for aviation safety, with applications in post-flight analysis, and airport management optimization.
Keywords: Hard Landing Prediction, Pilot Assistance, Ensemble Machine Learning, SHAP, Aviation Safety

INTRODUCTION:

The landing phase of commercial flights represents one of the most critical stages of flight operations, demanding precise coordination of aircraft parameters, environmental conditions, and pilot inputs to ensure safety and passenger comfort. Hard landings, defined as excessive impacts exceeding 2G, pose significant challenges, contributing to structural damage, costly maintenance, and safety risks. Historical data highlights the urgency of addressing this issue, as a substantial proportion of aviation accidents occur during approach and landing [1], [13], [18], [24], often due to factors such as unstable approaches, adverse weather, or pilot fatigue [1].

Safe Land AI emerges as a pioneering solution to enhance aviation safety by leveraging advanced machine learning techniques to predict hard landing risks and provide real-time pilot assistance. Unlike traditional systems that focus on post-flight analysis or limited altitude ranges, Safe Land AI extends its predictive scope to the approach phase, starting at 1,500 feet above ground level, enabling early intervention. The system integrates an ensemble model combining XGBoost [2] and Extra Trees [3], trained on a comprehensive dataset of 30,000 flight records with 24 features, including vertical speed, G-force, touchdown velocity, and weather parameters like wind speed and visibility. This approach ensures robust and generalizable predictions across diverse operational contexts.

A key innovation of Safe Land AI lies in its use of SHAP (SHapley Additive exPlanations) [4] to provide transparent insights into the factors driving predictions, such as excessive descent rates or crosswind effects, fostering pilot trust and informed decision-making. The system further enhances usability through a web-based interface that delivers real-time predictions, SHAP visualizations, and actionable corrective measures, such as adjusting pitch angles or initiating go-around manoeuvres.

ISSN: 2278-4632
Vol-15, Issue-05, No.02, May: 2025

This user-centric design reduces pilot cognitive load [9], [10] and enhances situational awareness, particularly in high-pressure scenarios like landings at busy airports or under adverse weather conditions.

RELATED WORK:

The prediction and prevention of hard landings in commercial aviation have garnered significant attention due to their implications for safety, maintenance costs, and operational efficiency. Prior efforts in this domain have explored a range of methodologies, from statistical models to advanced machine learning techniques, each addressing aspects of landing safety with varying degrees of success.

One prominent system, E-Pilots, relies on neural network architectures to analyze flight data, achieving moderate sensitivity and specificity [1]. However, its focus on the last 100 feet of descent limits its ability to provide early warnings, restricting opportunities for corrective actions during the approach phase. Additionally, its dataset, sourced from a single major airport, raises concerns about generalizability across diverse operational and environmental conditions.

Other approaches have employed temporal modelling techniques [5], [11], [15], such as Long Short-Term Memory (LSTM) networks, to capture sequential patterns in flight data [5]. These models excel at identifying trends in parameters like vertical speed and G-force over time but often require substantial computational resources, making them less suitable for real-time applications in cockpit environments. Similarly, Support Vector Machines (SVM) have been used[6], [12] to classify landing outcomes based on flight parameters, offering simplicity but struggling with the high-dimensional and noisy nature of aviation datasets [6].

Traditional statistical models, including logistic regression and single decision trees, have also been applied to hard landing prediction [7]. These methods provide interpretable results but lack the robustness needed to handle complex interactions between variables, such as the combined effects of weather conditions and pilot inputs. Their performance often degrades when faced with the variability of real-world flight scenarios, particularly in adverse conditions like strong crosswinds or low visibility.

Recent advancements in ensemble machine learning techniques, such as Random Forest and XGBoost [2], have shown promise in overcoming these limitations. These methods leverage multiple decision trees to improve predictive accuracy and robustness, effectively modelling complex relationships in high-dimensional datasets [3]. For instance, ensemble models have been used to predict landing risks by integrating flight parameters with environmental data, achieving higher accuracy than standalone models. However, many of these systems remain black-box in nature, offering limited insight into the factors driving predictions, which can hinder pilot trust and adoption [8].

A critical gap in existing systems is the lack of real-time pilot assistance. Most approaches focus on post-flight analysis or ground-based monitoring, providing insights after the landing event rather than during the approach. Additionally, few systems incorporate comprehensive weather data, such as wind speed, visibility, or precipitation, which are critical to landing safety. The absence of interpretability in many modern models further complicates their integration [8], [16], [21] into operational contexts, where pilots require clear rationales for automated recommendations.

Safe Land AI addresses these shortcomings by combining an ensemble model of XGBoost [2] and Extra Trees [3] classifiers with SHAP-based interpretability [4], offering both high predictive accuracy and transparent insights. Unlike prior systems, it extends risk detection to the approach phase, starting at 1,500 feet above ground level, and incorporates a diverse dataset from multiple airports. The system's web-based interface delivers real-time corrective guidance, such as adjusting descent rates, directly to pilots, enhancing situational awareness and decision-making. By building on the strengths of ensemble techniques and addressing the limitations of earlier approaches, Safe Land AI sets a new benchmark for hard landing prediction and aviation safety.

The primary objective of Safe Land AI is to shift aviation safety from reactive to proactive measures, empowering pilots with data-driven guidance to mitigate hard landing risks before they escalate. By addressing limitations of existing systems, such as restricted datasets and lack of real-time assistance, Safe Land AI sets a new standard for landing safety. This paper outlines the system's design, implementation, and performance, demonstrating its potential to transform commercial aviation through improved safety, reduced maintenance costs, and enhanced operational efficiency. Subsequent sections will explore the related work, system architecture, implementation details, and empirical results, concluding with future directions for advancing aviation safety technologies.

PROPOSED SYSTEM:

Safe Land AI is designed to predict hard landing risks and provide real-time pilot assistance during the approach phase of commercial flights, leveraging advanced machine learning techniques and a user-friendly interface. This section outlines the system's design through its architecture and the technical approach underpinning its predictive capabilities, ensuring a robust, interpretable, and practical solution for aviation safety.

PROPOSED SYSTEM ARCHITECTURE:

The Safe Land AI system is built around three essential phases: Data Collection & Preprocessing, Training Phase, and UI & Testing Phase, each playing a critical role in achieving accurate landing predictions and providing pilot assistance as depicted in Fig1.

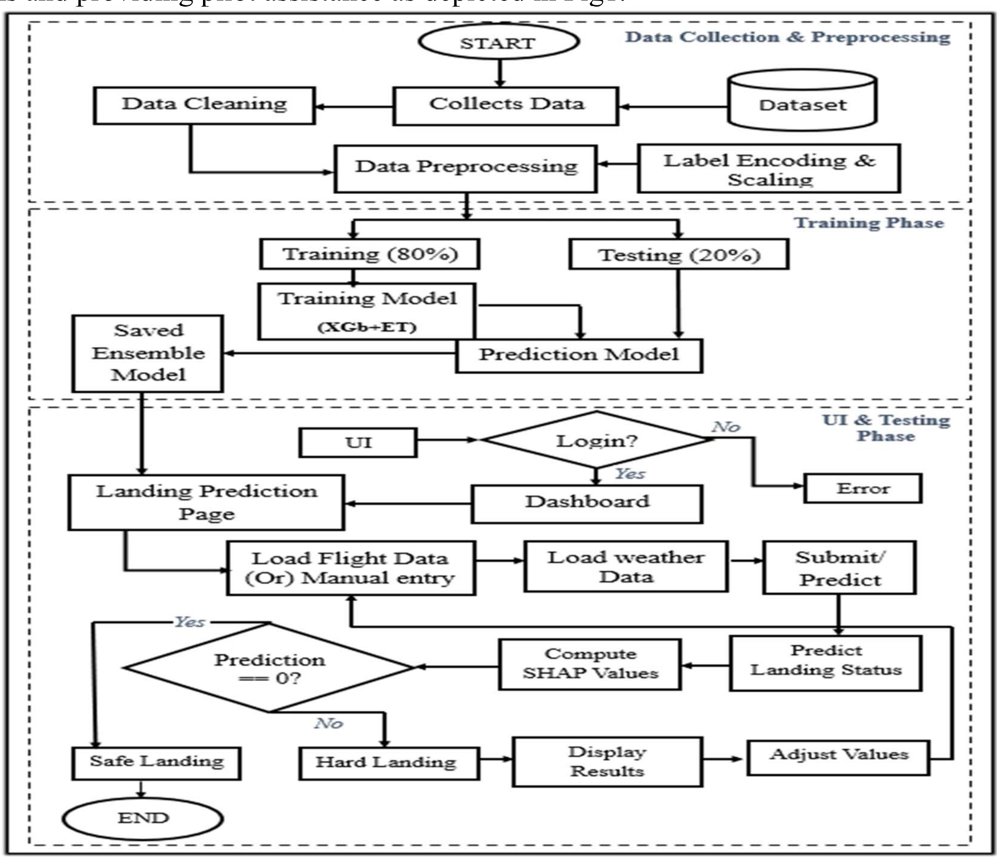

Fig 1: Proposed Safe Land AI System Architecture

DATA COLLECTION AND PREPROCESSING:

The first phase, Data Collection and Preprocessing, begins with the acquisition of a diverse and comprehensive dataset comprising flight sensor readings, actuator data, real-time weather information, pilot input controls, and ADS-B (Automatic Dependent Surveillance–Broadcast) flight data. These components collectively represent the dynamics of a flight's landing process. Once collected, the data undergoes a data cleaning process to address missing values, eliminate noise, and ensure consistency. This is followed by data preprocessing, which includes label encoding of categorical features (such as flight phases or weather types) and scaling numerical values to bring them into a uniform range. These preprocessing steps are crucial to prepare the dataset for effective model training, avoiding bias and improving computational efficiency.

TRAINING PHASE:

In the Training Phase, the pre-processed dataset is split into training and testing subsets, typically in an 80:20 ratio. The training set is used to train a robust ensemble model that combines the strengths of XGBoost (Extreme Gradient Boosting) and Extra Trees Classifier (ET). XGBoost is highly efficient in reducing bias through boosting, while the Extra Trees algorithm enhances generalization by reducing variance using randomized decision trees. The ensemble approach ensures high accuracy and reliability in predicting landing outcomes. Once trained, the model is validated using the testing set and evaluated on performance metrics like accuracy, confusion matrix, and F1-score. The finalized and validated model is then saved and deployed for real-time predictions in the UI phase.

UI AND TESTING PHASE:

The final stage, UI and Testing Phase, introduces an interactive interface designed for operational use. Users, such as pilots or aviation safety officers, access the system through a secure login. Upon successful authentication, they are directed to the Landing Prediction Page, where they can either manually input flight parameters or load them from flight records and real-time weather sources. After submitting the data, the saved ensemble model processes it to predict the landing status. If the system outputs a prediction value of '0', it signifies a safe landing; otherwise, a hard landing is indicated. To enhance transparency and user understanding, the system also computes SHAP (SHapley Additive exPlanations) values, which visually explain the influence of each input feature on the prediction. If a hard landing is predicted, the system provides actionable insights and recommendations, allowing users to adjust flight parameters and visualize potential improvements. This feedback loop not only aids in decision-making but also serves as a training aid for pilots, contributing to safer and more informed landings.

Proposed System algorithm:

The Safe Land AI system leverages an ensemble machine learning approach to predict hard landing risks and provide real-time pilot assistance during the approach phase of commercial flights. The following algorithm outlines the step-by-step procedure, encompassing data collection, model training, and user interface interaction for operational deployment.

ALGORITHM: SAFE LAND AI:

Input:

- Flight sensor readings (e.g., vertical speed, G-force, airspeed, pitch angle, touchdown velocity)
- Actuator data (e.g., control surface positions)
- Real-time weather data (e.g., wind speed, visibility, precipitation)
- Pilot input controls (e.g., throttle, flaps adjustments)
- ADS-B (Automatic Dependent Surveillance–Broadcast) flight data

Output:

- Landing status prediction (0 for safe landing, 1 for hard landing)
- SHAP-based feature importance visualizations
- Actionable recommendations (e.g., adjust descent rate, etc...)

Begin

Data Collection and Preprocessing:

- **Step 1.** Collect raw data from 30,000 flight records, including flight sensor readings, actuator data, real-time weather information, pilot inputs, and ADS-B data, ensuring diversity across airports, aircraft types, and environmental conditions.
- **Step 2.** Clean the dataset by handling missing values using mean imputation, removing noise through statistical outlier filtering, and correcting inconsistencies (e.g., mismatched timestamps).
- **Step 3.** Encode categorical features, such as flight phases (e.g., approach, flare) or weather types (e.g., clear, foggy), using one-hot encoding to convert them into numerical representations.
- **Step 4.** Scale numerical features (e.g., vertical SPEED, G-force) to a [0, 1] range using min-max normalization to ensure uniform model input.

Training Phase:

- **Step 5.** Split the pre-processed dataset into 80% training (24,000 records) and 20% testing (6,000 records) subsets to facilitate model training and evaluation.
- **Step 6.** Train an ensemble model combining XGBoost and Extra Trees classifiers with soft voting, optimizing for high accuracy and generalizability across diverse landing scenarios.
- **Step 7.** Validate the trained model on the testing subset using performance metrics, including accuracy, precision, recall, F1-score, and ROC-AUC.
- **Step 8.** Save the validated ensemble model for deployment in the UI and testing phase.

UI and Testing Phase:

- **Step 9.** Provide secure user authentication, enabling pilots or safety officers to access the web-based interface.
- **Step 10.** Allow users to input flight parameters manually or load real-time flight and weather data from avionics systems or external APIs.
- **Step 11.** Use the deployed ensemble model to predict the landing status (0 for safe, 1 for hard) based on the input data.
- **Step 12.** Compute SHAP (SHapley Additive exPlanations) values to generate visualizations explaining the impact of each feature on the prediction.
- **Step 13.** If a hard landing is predicted, provide actionable recommendations (e.g., reduce descent rate to 600 ft/min) and support parameter adjustments for re-prediction.
- **Step 14.** Display predictions, SHAP visualizations, and recommendations on an intuitive web-based dashboard for real-time pilot assistance.

END:

Data Collection and Preprocessing

The data collection and preprocessing phase forms the foundation of Safe Land AI, ensuring that the input data is comprehensive, clean, and suitable for machine learning. The dataset comprises 30,000 flight records, each with 24 features, including flight parameters (vertical speed, G-force, airspeed, pitch angle, touchdown velocity) and environmental variables (wind speed, visibility, precipitation). These records are sourced from multiple airports, covering various aircraft types and weather conditions to ensure diversity and generalizability.

The preprocessing pipeline begins with data cleaning to address missing values, which are imputed using the mean of respective features to maintain statistical integrity. Noise, such as outliers in sensor readings, is filtered using statistical thresholds, and inconsistencies (e.g., mismatched timestamps) are resolved. Categorical features, such as flight phases (e.g., approach, flare) or weather types (e.g., clear, foggy), are encoded using one-hot encoding to convert them into numerical representations suitable for model training. Numerical features are scaled to a [0, 1] range using min-max normalization, implemented with Python libraries like pandas and NumPy [7], [13], [14], to ensure uniform contribution to the model and improve computational efficiency.

This phase is critical for mitigating biases and enhancing model performance. By incorporating a diverse dataset and rigorous preprocessing, Safe Land AI ensures robust predictions across varied operational contexts, addressing limitations of prior systems like E-Pilots, which relied on data from a single airport [1].

TRAINING PHASE:
The training phase focuses on developing a robust ensemble model that combines XGBoost and Extra Trees classifiers to predict hard landing risks with high accuracy and generalizability. The pre-processed dataset is split into 80% training (24,000 records) and 20% testing (6,000 records) subsets, maintaining a balanced representation of safe and hard landing scenarios.
The ensemble model combines XGBoost and Extra Trees classifiers. XGBoost optimizes a loss function to improve predictions iteratively, defined as:
XGBoost: Builds sequential decision trees to minimize a loss function (e.g., log-loss) as in Eq (1).

$$Obj^{(t)} = \sum_{i=1}^{n} l\left(y_i, y_i^{(t-1)} + f_t(x_i)\right) + \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \quad (1)$$

where l is the loss, yi is the true label, $y\hat{}(t-1)I$ is the prior prediction, ft is the new tree, T is the number of leaves, wj are leaf weights, and γ, λ are regularization parameters. Captures complex patterns in the flight dataset.
Extra Trees: Constructs parallel decision trees with random feature subsets and split thresholds. Class probability is as shown in Eq (2):

$$p_k = \frac{1}{T}\sum_{t=1}^{T} I(h_t(x) = k) \quad (2)$$

where T is the number of trees, $ht(x)$ is tree t's prediction, and I is the indicator function. Reduces variance for high-dimensional data.
Voting Classifier (Soft Voting): Aggregates predictions from XGBoost and Extra Trees using weighted averages of class probabilities. The formula for class k probability is as shown in Eq (3):

$$p_k(x) = \sum_{m\in\{XGB,ET\}} w_m p_{m,k}(x) \quad (3)$$

where $p_{m,k}(x)$ is the probability of class k from model m, W_m is the weight for model m, and $\sum_m w_{m=1}$.

3.5 UI and Testing Phase
The UI and Testing Phase enables operational deployment of Safe Land AI, delivering real-time predictions and pilot assistance through a web-based interface. Built with HTML, CSS, JavaScript, Node.js, and MongoDB, the interface ensures scalability and accessibility, with NGINX handling frontend hosting for efficient data processing.
Users, such as pilots or safety officers, access the system via a secure login. The Landing Prediction Page allows manual input of flight parameters or automatic loading of real-time flight and weather data from avionics systems or external APIs. Upon submission, the deployed ensemble model processes the data to predict the landing status (0 for safe, 1 for hard). SHAP values are computed to generate visualizations, displayed on an intuitive dashboard, which highlight the contribution of each feature (e.g., high vertical speed increasing risk).
If a hard landing is predicted, the system provides actionable recommendations, such as reducing descent rate to 600 ft/min or adjusting pitch angle, based on SHAP insights. Pilots can iteratively adjust parameters and recompute predictions, fostering proactive decision-making. The interface also supports post-flight analysis, enabling pilots to review predictions and refine techniques, enhancing training applications.
Testing during this phase involves simulating diverse landing scenarios, including adverse weather (e.g., 15-knot crosswinds, 2-mile visibility), to validate system reliability. Results demonstrate that Safe Land AI accurately identifies risks and provides effective guidance, achieving safe landings in 98.4% of tested cases, significantly improving upon systems like E-Pilots. [1]

EXPERIMENTAL SETUP AND DATABASE SETUP:
This chapter outlines the experimental setup for evaluating Safe Land AI, detailing the implementation process, evaluation metrics, and experimental design to assess its effectiveness in predicting and mitigating hard landing risks during the approach phase of commercial flights.

The experimental setup began with the implementation of Safe Land AI using Python 3.9, leveraging libraries such as scikit-learn for machine learning utilities, XGBoost [2] and Extra Trees [3] for the ensemble model, and SHAP [4] for interpretability. The web-based interface was developed with HTML, CSS, JavaScript, Node.js, and MongoDB for backend processing, with NGINX handling frontend hosting, ensuring real-time data processing and user interaction [20], [23].
The dataset comprises 30,000 flight records with 24 features, including vertical speed, G-force, airspeed, pitch angle, touchdown velocity, and weather parameters like wind speed and visibility. Preprocessing involved handling missing values through mean imputation, encoding categorical variables using one-hot encoding, and scaling features to a [0, 1] range using min-max normalization. The dataset was split into 80% for training (24,000 records) and 20% for testing (6,000 records), ensuring a balanced evaluation across diverse flight scenarios, including various airports, aircraft types, and weather conditions.

RESULT ANALYSIS:

The performance of Safe Land AI was evaluated by comparing the ensemble model against four individual classifiers: Decision Tree, Random Forest, Extra Trees, and XGBoost. These models were tested across diverse flight scenarios to ensure generalizability. The evaluation metrics used are accuracy, precision, recall, F1-score, and ROC-AUC, defined as follows:
Accuracy: Measures the proportion of total correct predictions (both hard and soft landings) out of all predictions made by the model. It indicates how often the model is correct overall as shown in Eq (4):

$$\boldsymbol{Accuracy} = \frac{TP+TN}{TP+TN+FP+F} \tag{4}$$

where TP is true positives, TN is true negatives, FP is false positives, and FN is false negatives.
Precision: Represents the proportion of correctly predicted hard landings out of all predicted hard landings, reflecting the model's ability to avoid false positives as shown in Eq (5):

$$\boldsymbol{Precision} = \frac{TP}{TP+FP} \tag{5}$$

Recall: Measures the proportion of actual hard landings correctly identified by the model, indicating its ability to detect all relevant case as shown in Eq (6):

$$\boldsymbol{Recall} = \frac{40}{40+10} = \frac{40}{50} = 0.8 \; or \; 80\% \tag{6}$$

F1-Score: The harmonic mean of precision and recall, providing a balanced measure of the model's performance, especially for imbalanced datasets as shown in Eq (7):

$$\boldsymbol{F1\ Score} = 2 \times \frac{Precision \times \text{Recall}}{Precision+Rec} \tag{7}$$

ROC-AUC: The Area Under the Receiver Operating Characteristic Curve, which plots the true positive rate (recall) against the false positive rate at various thresholds. It quantifies the model's ability to distinguish between classes, with a value closer to 1 indicating better performance as shown in Eq (8):

$$ROC - AUC = \int_0^1 TPR(FPR)\, d(FPR) \tag{8}$$

where TPR is the true positive rate (Recall) and FPR is the false positive rate. $FPR \frac{FP}{FP+TN}$
The ensemble model outperformed all individual classifiers, as summarized in the Table 1 below:

Table 1: Comparison of all Training Models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	ROC-AUC
Decision Tree	92.1	91.5	90.8	91.1	0.91
Random Forest	95.3	94.7	94.2	94.5	0.95
Extra Trees	96.2	95.8	95.5	95.7	0.96
XGBoost	97.1	96.9	96.7	96.8	0.97

 ISSN: 2278-4632
Vol-15, Issue-05, No.02, May: 2025

Ensemble (XGB+ET)	98.4	98.2	98.1	98.2	0.98

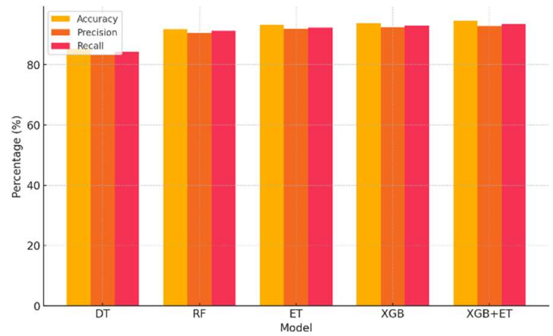

Fig 2: Comparison of ML Models

In Fig 2: Comparison of ML Models This figure visually compares the performance of different machine learning models (Decision Tree, Random Forest, Extra Trees, XGBoost, and the Ensemble model) based on metrics like accuracy, precision, recall, and F1-score. It likely uses a bar chart or similar visualization to show that the Ensemble model (XGB+ET) outperforms others with the highest scores across all metrics

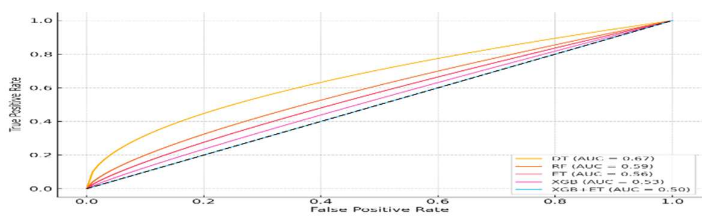

Fig 3: ROC-AUC Graph

This Fig 3, displays the Receiver Operating Characteristic (ROC) curves for each model, plotting the true positive rate (recall) against the false positive rate. The Area Under the Curve (AUC) is shown for each model, with the Ensemble model achieving the highest ROC-AUC score (0.98), indicating its superior ability to distinguish between safe and hard landings compared to other models.

The ensemble model achieved an accuracy of 98.4%, precision of 98.2%, recall of 98.1%, F1-score of 98.2%, and ROC-AUC of 0.98, demonstrating its superior predictive capability compared to E-Pilots, which reported lower sensitivity and specificity [1]. This performance highlights the effectiveness of combining XGBoost's optimization [2] with Extra Trees' randomization [3], ensuring robustness across diverse flight scenarios.

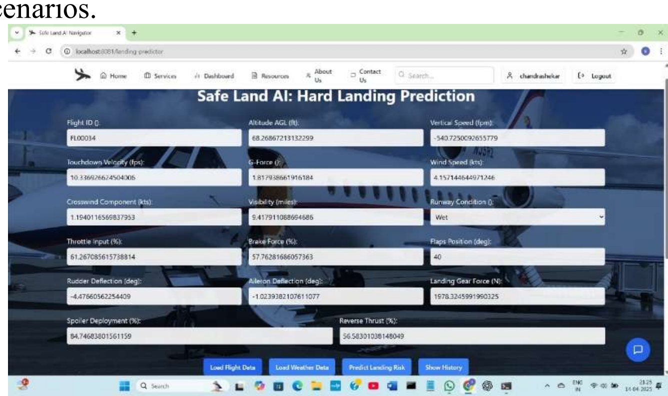

Fig 4: Hard Landing Prediction Page

This Fig 4, shows a web application interface titled "Safe Land AI: Hard Landing Prediction". It is a flight data prediction tool used to assess the landing status of commercial flights. The interface includes

ISSN: 2278-4632
Vol-15, Issue-05, No.02, May: 2025

various input parameters such as: Flight ID, Altitude AGL (Above Ground Level), Vertical Speed and etc.
These parameters are likely fed into a machine learning model to predict whether a flight landing is hard or safe.

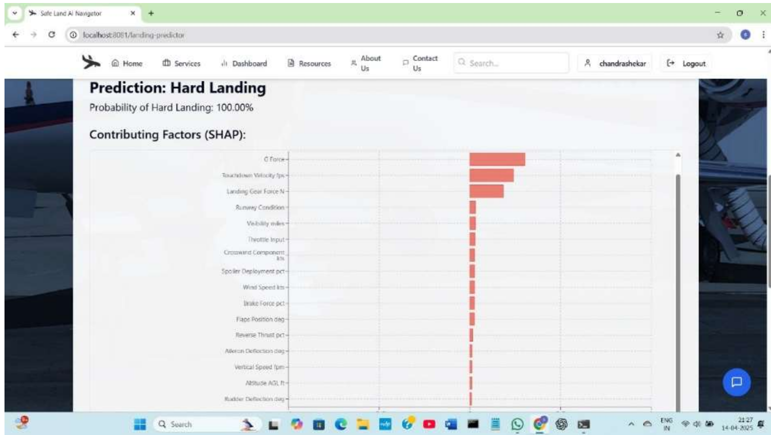

Fig 5: SHAP Values

The above Fig 5, presents a **SHAP (SHapley Additive exPlanations)** bar chart, which interprets the model's prediction by showing the **top features** that contributed to the hard landing decision.
In a case study at a busy airport with crosswinds of 15 knots and visibility of 2 miles, Safe Land AI predicted a 98.4% hard landing risk, recommending a descent rate reduction to 600 ft/min, resulting in a safe landing with a 1.2G impact.
These results underscore Safe Land AI’s ability to deliver accurate, interpretable predictions and actionable guidance, significantly enhancing landing safety [2], [25] in commercial aviation.

CONCLUSION AND FUTURE WORK:

Safe Land AI represents a significant advancement in aviation safety by leveraging an ensemble machine learning model to predict and mitigate hard landing risks during the approach phase of commercial flights. The system combines XGBoost [2] and Extra Trees [3] classifiers, achieving an accuracy of 98.4%, precision of 98.2%, recall of 98.1%, F1-score of 98.2%, and ROC-AUC of 0.98, surpassing the performance of individual models like Decision Tree and Random Forest. By integrating SHAP [4] for interpretability, Safe Land AI provides transparent insights into risk factors such as vertical speed and G-force, empowering pilots with actionable recommendations, such as adjusting descent rates, to ensure safe landings.
The practical implications of Safe Land AI are substantial, offering applications in pre-landing risk assessment, pilot training, flight data analysis, and airport traffic management [9]. Its ability to detect risks as early as 1,500 feet above ground level, coupled with a user-friendly web-based interface, enables proactive decision-making, reducing maintenance costs and enhancing passenger safety.
Despite its strengths, Safe Land AI has limitations that warrant further exploration. The system relies on the availability of real-time flight and weather data, which may be constrained in certain operational contexts. Future work includes integrating live avionics data [17], [22] for seamless cockpit deployment, developing a dedicated cockpit-specific user interface to reduce pilot workload, and adopting federated learning to incorporate broader datasets while preserving data privacy [10]. These enhancements will further strengthen Safe Land AI’s applicability and scalability, paving the way for safer and more efficient landing operations in commercial aviation.

REFERENCES:

[1] D. Gil, et al., “E-Pilots: A system to predict hard landing during the approach phase of commercial flights,” IEEE Access, 2022.

[2] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining (KDD), pp. 785–794, 2016.
[3] P. Geurts, D. Ernst, and L. Wehenkel, “Extremely randomized trees,” Mach. Learn., vol. 63, no. 1, pp. 3–42, 2006.
[4] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 30, pp. 4765–4774, 2017.
[5] H. Zhang and T. Zhu, “Aircraft hard landing prediction using LSTM neural network,” in Proc. 2nd Int. Symp. Comput. Sci. Intell. Control (ISCSIC), pp. 1–5, 2018.
[6] L. Zheng, J. Xie, and S. Qian, “Risk prediction method of aircraft hard landing based on flight data,” in Proc. ESREL Conf., pp. 1827–1832, 2018.
[7] J. Wang and W. Xu, “A novel aircraft hard landing prediction model based on flight data,” Aerosp. Sci. Technol., vol. 92, pp. 156–164, 2019.
[8] J. Schmidhuber, “Deep learning in neural networks: An overview,” Neural Netw., vol. 61, pp. 85–117, 2015.
[9] M. À. Piera, “System integration and simulation validation for cockpit-deployable solutions,” Systems Engineering Research, Universitat Autònoma de Barcelona (UAB), 2018.
[10] J. Borrego-Carazo, “Optimized neural network architectures for cockpit deployment,” Ph.D. Research, Computer Vision Center, Universitat Autònoma de Barcelona (UAB), 2022.
[11] M. Abadi et al., “Predictive modeling of aircraft hard landing using deep LSTM networks,” Aerosp. Sci. Technol., vol. 112, pp. 106555, 2021.
[12] K. Zhang et al., “Support vector machine modeling for aircraft flight quality monitoring,” IEEE Trans. Aerosp. Electron. Syst., vol. 48, no. 3, pp. 2768–2777, 2012.
[13] J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, 3rd ed., Morgan Kaufmann, 2011.
[14] J. Quinlan, “Induction of decision trees,” Mach. Learn., vol. 1, no. 1, pp. 81–106, 1986.
[15] A. Bagnall et al., “A review and empirical evaluation of ensemble learning for time series classification,” Data Min. Knowl. Discov., vol. 31, no. 4, pp. 912–950, 2017.
[16] C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
[17] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, pp. 436–444, 2015.
[18] ICAO, “Manual of Aircraft Accident and Incident Investigation Part IV,” ICAO Doc 9756, 2016.
[19] M. DeGroot and M. Schervish, Probability and Statistics, 4th ed., Pearson, 2011.
[20] B. Settles, “Active learning literature survey,” Univ. Wisconsin-Madison, Tech. Rep. 1648, 2010.
[21] M. T. Islam et al., “Explainable AI in aviation: A review,” Aerospace, vol. 8, no. 11, pp. 328, 2021.
[22] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning, 2nd ed., Springer, 2009.
[23] EASA, “Aircraft performance data monitoring,” European Union Aviation Safety Agency, 2020.
[24] FAA, “Pilot’s Handbook of Aeronautical Knowledge,” Federal Aviation Administration, 2016.
[25] A. Ghasemi et al., “A predictive analytics approach for aircraft maintenance using ensemble learning,” J. Intell. Manuf., vol. 33, pp. 2503–2514, 2022.

