Data-driven Long-Landing Event Detection and Interpretability Analysis in Civil Aviation

XIONG YANG1, JIN REN1,2, JUNCHEN LI3, HAIGANG ZHANG1, and JINFENG YANG1

1Institute of Applied Artificial Intelligence of the Guangdong-Hong Kong-Macao Greater Bay Area, Shenzhen Polytechnic, Shenzhen (518055), China
2Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen (518055), China
3College of Engineering, Southern University of Science and Technology, Shenzhen (518055), China

Corresponding author: Jinfeng Yang (e-mail: jfyang@szpt.edu.cn).

This work has been supported by the National Natural Science Foundation of China (No. 62076166), China Postdoctoral Science Foundation (No. 2021M703371), Shenzhen Science and Technology Program (No. RCBS20200714114940262), Post-doctoral Foundation Project of Shenzhen Polytechnic (No. 6021330002K), and the General Higher Education Project of Guangdong Provincial Education Department (No. 2020ZDZX3082, No. 2020ZDZX3085).

ABSTRACT Long-landing events (LLEs) can lead to reduced available runway length and increased operating costs, which are primarily caused by the improper operation of the aircraft, i.e., by human error. The pilot’s operation data are comprehensively recorded in the quick access recorder (QAR) during aircraft takeoff and landing, and analyzing the QAR data can determine the causes of the LLEs. Traditionally, domain experts inspect LLEs by manually setting thresholds on uni-dimensional data. However, this approach cannot detect the defects caused by the pilot’s maneuvering technique because the potential mutual information between different features in the large amount of data is not considered. This paper proposes a data-driven LLE detection and causation analysis workflow, which can automatically mine and analyze this information, to overcome the existing problems. Firstly, a dataset is established based on the extracted QAR data from 2002 flights, considering the landing phase of the aircraft. Subsequently, a hybrid-feature-selection (HFS) method is implemented to select the features that are highly relevant to the LLEs using both unsupervised and supervised algorithms. A categorical Light Gradient Boosting Machine (LGBM) with a Bayesian optimization (LGBMBO) model is used to determine the performance improvement. Furthermore, the model is visualized to analyze the marginal effect of key parameters for the LLEs by using SHapley Additive exPlanations (SHAP). The experimental results demonstrate that the proposed HFS-LGBMBO model reduces computational cost and achieves better performance. Additionally, they demonstrate that LLEs can be avoided during the landing phase by maintaining the appropriate descent speed, the appropriate aircraft altitude, and proper descent angle.

INDEX TERMS Flight safety, Long-landing event, Anomaly detection, SHapley Additive exPlanations, Feature engineering, Machine learning

I. INTRODUCTION

The aircraft landing phase is the period with the highest accident rate, since it is closely associated with the geographical environment of the airport, weather conditions, pilot’s operational expertise, and level of maintenance of the aircraft parts [1]. The accident data analysis indicates that the Runway Excursion (RE) category is the most common type of accident reported each year, which occurs during the landing phase [2]. The RE accounted for 14.9% of all accidents and included 50% of all fatal accidents, with 44 fatalities in 2019 [3]. The long-landing event (LLE) is one of the corresponding RE accident indicators. LLEs occur when the aircraft’s landing distance lies beyond the normal range, thus reducing the available length of the runway and considerably increasing the risk. Even worse, the aircraft may run off if the pilots lack the necessary proficiency, leading to unpredictable consequences [4], [5]. Therefore, the detection of LLEs and their causes can help in preventing RE accidents, benefit airlines management and pilot training, and help in the development of aviation safety regulations.
The Quick Access Recorder (QAR) is an onboard data recording device which collects the entire flight information via sensors placed throughout the aircraft during the flight phases. Research on the QAR has gained considerable attention from civil aviation safety experts for the evaluation of flight quality and detection of risk events due to its characteristics of low cost and portability [6]. The Exceedance Detection (ED) method is one of the most efficient methods to monitor the QAR data; it includes various trigger logic expressions based on operational manuals, training programs, and risk assessment procedures on uni-dimensional data [7]. However, this approach depends on the experience of domain experts, making it difficult to efficiently utilize the potential mutual information between different features in the huge amount of data; it thus fails to identify the defects caused by the pilot’s maneuvering technique [8]–[11]. Consequently, the trade-off between the ability to detect anomalous events and the inference of their causality presents a considerable drawback in the analysis of large amounts of aviation data.

This study attempts to break down this problem into three sub-problems as follows: selecting the features most relevant to LLEs from a large number of features; determining whether the proposed anomaly detection model is optimal when compared with other machine learning algorithms; lastly, excavating the correlation between key parameters and LLEs. This study solves these problems by employing a data-driven workflow based on an interpretable machine learning approach. The main contributions of this paper are as follows:

- The QAR data from 2002 flights is collected to build a dataset of LLE anomalies that occur during the landing phase.
- A fast and accurate Hybrid Feature Selection (HFS) algorithm, which presents the advantages of both unsupervised and supervised algorithms, is adopted for the QAR data to reduce data redundancy and determine the most relevant features.
- This study tunes the hyper-parameters of the Light Gradient Boosting Machine (LGBM) by incorporating Bayesian Optimization (BO) to improve the anomaly detection capability of the model in order to overcome the problem of sample imbalance in the abnormal data.
- It employs the interpretable machine learning framework, i.e., SHapley Additive exPlanations (SHAP), to excavate the correlations between the parameter within the anomaly detection model.

The rest of the paper is structured as follows. Section 2 presents a description of the related works. Section 3 depicts the proposed system architecture, including dataset construction, feature engineering, anomaly detection model, and interpretability analysis. Section 4 presents the experimental results, and explains and analyzes the models using SHAP. Lastly, Section 5 presents the conclusion.
TABLE 1. The different kinds of anomaly detection approaches for aviation data

| Category                  | Characteristics                                      | Method     | Author          | Year |
|---------------------------|-------------------------------------------------------|------------|-----------------|------|
| Clustering-based          | Anomalies are distinguished based on similarity between samples | MKAD       | S. Dar [12], [13]| 2010 |
|                           |                                                       | ClusterAD  | L. Li [14]      | 2015 |
|                           |                                                       | LOF        | T. G. Puranik [15] | 2017 |
| Regression-based          | Continuous values are taken as outliers                | SMS-VAR    | I. Melnyk [16]  | 2016 |
|                           |                                                       | LSTM       | C. Tong [17]    | 2018 |
|                           |                                                       | RF         | T. G. Puranik [18] | 2020 |
| Classification-based      | Discrete anomaly labels are taken as supervision       | CVAE       | M. Memarzade [19] | 2020 |
|                           |                                                       | XGBoost    | M. Rey [20]     | 2021 |

IV. METHODOLOGY

A. SYSTEM ARCHITECTURE

In this study, it is assumed that the aircraft is in good operating condition, that the pilots are able to maneuver the aircraft normally, and that the data recorded by the QAR are consistent with the pilot’s maneuvers, so as to investigate the relationship between LLEs and pilot operation characteristics. In addition, the influence of external factors, such as runway conditions and weather conditions, are not the focus in this paper. The proposed anomalous event detection and analysis scheme consists of four steps: dataset construction, HFS-based feature selection, LGBMBO-based anomaly detection, and analysis of experimental results, including performance evaluation and interpretability analysis, as shown in the Figure 1.

B. DATASET CONSTRUCTION

1) Parameters

The parameters are selected based on two aspects: the types of parameters in the QAR data and the parameters related to the landing phase. The QAR records four types of flight parameters: alarm parameters, environment parameters, status parameters, and operation parameters. Among these, the status and operation parameters are the most important indicators, which represent the pilot’s operational expertise. Subsequently, the landing phase of the aircraft can be divided into three phases: the glide slope phase, the flare-out phase, and the deceleration taxiing phase [23]. During the glide slope phase, the aircraft transitions from gliding to an approximate flying state, generally employing the small throttle sliding method. During the flare-out phase, the aircraft mainly adjusts its speed, attitude, and angle for landing. Lastly, during the deceleration taxiing phase, the aircraft’s drag must be increased to reduce the speed as quickly as possible to ultimately reduce the landing glide distance. The glide slope phase is affected by the vertical height of the aircraft as it passes the runway threshold, the gliding angle, and other factors. The deceleration taxiing phase after grounding, is affected by the grounding speed of the aircraft, the deceleration device, the slope of the airport runway, the wind speed on the ground, and other factors. Consequently, the parameters which characterize the flight dynamics of the aircraft during the landing phase are selected from the aircraft status parameters and the operation parameters of the QAR data. These parameters can be classified into the velocity-type, position-type, status-type, engine-type, and deceleration-type. Subsequently, the relevant parameters are selected from more than 2000 parameters in the QAR data, with 151 columns; the selected parameter types and examples are presented in the Table 2.

2) Data extraction

The LLEs typically occur during the approach phase, which is the most common phase for aviation accidents. The aim of the data extraction process is to select the landing phase from the entire flight data frame. This process is performed according to the same rules for each flight in this study: each flight must have the same dimensions and a unified sampling point. During the landing phase, the unified point is defined as the maximum aircraft braking value. For each flight, 49 time-stamps are taken before the landing point, and ten time-
TABLE 2. The landing-related parameters

| Category       | Parameters                              | Unit   |
|----------------|-----------------------------------------|--------|
| Velocity       | Ground Speed (vGS)                      | KNOTS  |
|                | Computed Speed (vCAS)                    | KNOTS  |
|                | Instantaneous Vertical Velocity (vIVV)   | FT/MIN |
| Position       | Radio Altitude (vALT_RADIO)             | FEET   |
|                | Glide Angle Minus A/C Dev (vGA_M_3D)    | DEGS   |
|                | Glide Deviation in dots (vGLIDE)        | DOTS   |
| Status         | Flap Angle (vFLAP)                      | DEG    |
|                | Elevator Angle (vELEV)                  | DEG    |
|                | Pitch Angle (vPITCH)                    | DEG    |
| Engine         | Spd Brake Handle Posn (vSPDBRKPOS)      | DEG    |
| Deceleration   |                                        |        |

3) Labelling of anomalous event through regularized matching

The QAR database and the abnormal events database are obtained using the airline’s decoding software, AirFase. The QAR database includes multi-dimensional time-series data such as flight registration numbers, airline registration numbers, aircraft type numbers, flight execution dates, parameters records, and flight operation time. The AirFase software creates the abnormal events database by analyzing the documents of the Flight Standards Department of the Civil Aviation Administration of China (CAAC) [24] and the airline QAR monitoring standards, which contains the flight registration number, aircraft type number, abnormal phase, abnormal events occurrence events and abnormal level, and so on. In the abnormal events database, an LLE is defined as the aircraft landing which occurs when the landing point is more than 700 meters from the start of the runway. Lastly, the QAR and abnormal events databases are matched to obtain the abnormal datasets through regularization filtering.
matching flight numbers, determining abnormal time, labeling abnormalities, and so on.

C. HYBRID FEATURE SELECTION

Several sensors perform the same function in an aircraft. These sensors ensure that when one fails, the others remain in a working condition. This redundant mechanism ensures the safety of the aircraft. However, it adds unnecessary complexity in the data analysis. Therefore, this paper combines the Unsupervised Feature Selection (UFS) and Supervised Feature Selection (SFS) algorithm to design an HFS algorithm for high-dimensional aviation data.

1) Unsupervised feature selection to eliminate redundant features

The UFS evaluates the relevance of the features by analyzing the internal structure between the features [25]. The unsupervised filtering method of the Spearman rank correlation coefficient (SRCC) feature selection filters redundant features without using label information. The assumptions of Spearman’s correlation are that the data must be at least ordinal and that one variable must be monotonically correlated with another variable [26]. The QAR data exactly satisfy the assumptions of SRCC. Therefore, the variables with high monotonicity can be filtered by SRCC as a way to remove the redundant features from the QAR data. The SRCC includes values ranging from -1 to 1. The values close to 1 represent a strong monotonic increasing relationship between two variables, while those close to -1 represent a strong monotonic decreasing relationship. When the correlation value is close to 0, it indicates that there is no correlation between two variables [27]. The absolute value of the correlation matrix is taken because both monotonic increasing and monotonic decreasing correlations are relevant in the current data set. The QAR data exactly satisfy the assumptions of SRCC. The variables with high monotonicity are filtered out by SRCC as a way to remove redundant features from the QAR data. The SRCC is defined as follows,

\[
\rho = 1 - \frac{6 \sum_{i=1}^{N} d_i^2}{N(N^2 - 1)},
\]

where \( d \) represents the corresponding subtraction of the elements in the two variables, \( a \) and \( b \) to obtain a ranking difference set, and \( d_i = A_i - B_i \). \( N \) represents the total number of individual elements, which in this instance, is the number of rows of the QAR data. If the correlation coefficient of the two variables is closer to 1, it indicates that the two variables have a strong correlation, which can be considered to be redundant. The correlation threshold is set to 0.95 based on the experiments conducted in this study.

2) Supervised feature selection to select the most relevant features

The SFS selects more relevant features based on the correlation between the features and class labels, resulting in better performance in the subset [28]. This study proposes a high-performance automatic supervised feature selection method, which is obtained by cross-validation using the eXtreme Gradient Boosting (XGBoost) and Recursive Feature Elimination (RFE) methods. XGBoost is an ensemble machine learning algorithm based on a gradient boosting decision trees (GBDT) framework [29]. GBDT is a type of integrated learning boosting, where the boosting method trains a bunch of individual models continuously, each model learning from the error of the previous model [30]. Assuming that the GBDT model input sample is

\[
\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F.
\]

where \( F \) represents the set of all weak learners. Compared with GBDT, XGBoost introduces a regular term in the cost function for controlling the complexity of the model as follows.

\[
\Omega(f_k) = \mu T + \frac{1}{2} \lambda \sum_{j=1}^{T} \omega_j^2.
\]

The objective function of XGBoost is composed of two parts: training loss and regularization, as represented in the (5).

\[
Obj = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k).
\]

Then a Taylor second-order expansion of the loss function is performed to estimate the loss function. By approximating the objective function, the whole model has a faster convergence rate. Then, the optimal tree structure corresponding to the objective function is defined as follows,

\[
Obj = -\frac{1}{2} \sum_{j=1}^{T} G_j^2 \frac{H_j}{H_j + x} + \gamma T.
\]

where \( G_j \) and \( H_j \) represent the cumulative sum of the first-order partial derivatives and second-order partial derivatives concerning the approximate loss function, respectively. XGBoost then calculates the importance of the features by summing the number of splits of the features in each tree to select the important features.

RFE is a wrapper feature selection method, which aims to select the best or worst features based on the coefficients, remove the selected features, and repeat the process in the remaining set of features until all the features are traversed [31]. Firstly, the estimator is trained on the initial set of features to determine the importance of each feature. The least important features are then removed from the current set of features. This process is repeated recursively on the pruned set until the final number of features to be selected are obtained. The XGBoost and RFE feature selection methods form a set of embedding methods for feature selection. The procedure of the combined XGBoost and RFE feature selection algorithms are as follows:

\[
\}

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
Step1. The XGBoost model is trained on the original features, and each feature is assigned a weight, i.e., feature importance.

Step2. The features with the smallest absolute value weights are moved out of the feature set space.

Step3. This process is repeated recursively until the number of remaining features reaches a set feature count threshold.

Step4. The performance of the selected feature subset is evaluated and lastly, the feature subset with the best performance is selected.

D. ANOMALY DETECTION MODEL

1) LGBM model for Anomaly detection

The LGBM algorithm is an integrated algorithm, which is based on the gradient boosting framework proposed by Microsoft Research. The LGBM presents a higher accuracy and faster model training time when compared with XGBoost and is better optimized for the construction of decision trees [32]. Thus, the main computational principles of both algorithms are identical. The improvements made by the LGBM in the construction of decision trees are reflected in the proposed GOSS algorithm, EFB algorithm, Histogram algorithm, and leaf-wise leaf node generation method. The GOSS algorithm is used to reduce the training sample size, the EFB algorithm is used to reduce the sample dimension or reduce the sample features, and the Histogram algorithm is used to identify the best splitting point. Particularly, the leaf-wise growth approach is computationally less expensive and avoids overfitting when compared with the XGBoost approach of building decision trees using level-wise leaf node growth.

2) Bayesian optimization algorithm for hyperparameter tuning

The LGBM is tedious and complicated since the parameters can be only adjusted manually, which deteriorates the performance of the algorithm. Therefore, in this study, the parameters are automatically adjusted by using the Bayesian algorithm to ensure that the LGBM algorithm achieves the best results. The BO works by constructing a posterior distribution of functions (Gaussian process) that best describes the function that wants to optimize [33]. As the number of observations grows, the posterior distribution improves, then the model will focus on the region of the parameter space with the most promising validation scores.

The BO contains two core parts: the probabilistic surrogate model (PSM) and the acquisition function (AC). The PSM can be divided into parametric and non-parametric models based on the parameters of the model. The Gaussian process (GP), which is a non-parametric model with a variable number of parameters when compared with the parametric model, can better describe the unknown objective function and strongly fits the function due to its greater flexibility and scalability.

The acquisition function (AC) is an important basis to determine the next evaluation point using the Gaussian Process-Upper Confidence Bound (GP-UCB), as defined in the (7),

\[ x_{t+1} = \arg \max_{x} \mu_t(x) + \sqrt{\beta_t \delta_t(x)} \]

where the parameter \( \beta_t \) represents a constant for equilibrium exploration and exploitation, \( \mu_t(x) \) represents the mean, and \( \delta_t(x) \) represents the standard deviation.

E. MODEL INTERPRETATION

This paper uses the machine learning interpretability tool, SHAP, to analyze the LGBMBO model. SHAP consists of several machine learning interpretable strategies proposed by Lundberg and Lee in 2016 [34]. It is derived from a game-theoretic approach called Shapley values, which fundamentally interpret the output of any machine learning model in terms of whether it is parameterized or not. SHAP can explain the working concept of machine learning by reverse-engineering the output of any predictive model. SHAP interprets the model’s predicted value as the sum of the imputed values of each input feature, which is a linear model expressed as an additive feature imputation method. The additive feature imputation method is defined as follows,

\[ f(x) = g(x^{'}) + \sum_{j=1}^{M} \phi_j x^{'}, \]

where \( f(x) \) represents the original model, \( g(x^{'}) \) represents the explanatory model, \( x^{'}, \) represents the simplified input for variable \( x, \) and \( M \) represents the number of input features. In the SHAP interpretable model, \( \phi_0 \) represents the mean value of the mapped SHAP for all training samples, \( \phi_j \) represents the SHAP value corresponding to variable \( j, \) and \( x^{'}, \) indicates the value of 1 when variable \( j \) is selected and 0 otherwise.

The SHAP value \( \phi_j \) is defined as follows,

\[ \phi_j = \frac{[S]!(M - |S| - 1)!}{M!}[f_x(S \cup \{i\}) - f_x(S)] \]  

where \( N \) represents the set of all features, \( S \) represents a sequential subset of \( N, \) and \( M \) represents the number of input features in the (8).

The SHAP value is used to measure the degree of influence of a feature on the predicted result. A positive SHAP value indicates that the variable has a promoting effect on the prediction value in the sample. Whereas, a negative SHAP value indicates that the variable has a restraining effect on the prediction value in the sample. A larger absolute value of the SHAP value indicates that the variable has a greater influence on the predicted value in the sample.

IV. EXPERIMENTS AND ANALYSIS

A. EVALUATION MEASURES

In this paper, the confusion matrix description is shown in Table 3.
For unbalanced data, a single accuracy metric cannot sufficiently reflect the performance of the model because the accuracy (Acc) of the majority and minority sample classification must be considered together. Therefore, the F1 score (F1) and the Matthews correlation coefficient (MCC) is used to analyze the confusion matrix, and the area under the curve (AUC), which includes the receiver operating characteristic (ROC) curve and the precision-recall (PR) curve, is used to determine the performance of the classifier.

The following definitions give the evaluation indicators of the classifier:

1) Acc is defined as:

$$ Acc = \frac{TP + TN}{TP + TN + FN + FP} \quad (10) $$

2) F1 is defined as:

$$ F1 = \frac{2TP}{2TP + FP + FN} \quad (11) $$

3) MCC is defined as:

$$ MCC = \frac{TP/N - S \times P}{\sqrt{P \times S(1-S)(1-P)}} \quad (12) $$

where, $N$, $S$, $P$ can be defined as follows,

$$ N = TN + TP + FN + FP, \quad (13) $$

$$ S = \frac{TP + FN}{N}, \quad (14) $$

$$ P = \frac{TP + FP}{N}. \quad (15) $$

4) The ROC curve presents a curve with FPR as the y-axis and TPR as the x-axis, PR curve presents a curve with precision rate as the y-axis and recall rate as the x-axis, which can be defined as follows,

$$ FPR = \frac{FP}{FP + TN}, \quad (16) $$

$$ TPR = \frac{TP}{TP + FN}, \quad (17) $$

$$ precision = \frac{TP}{TP + FP}, \quad (18) $$

$$ recall = \frac{TP}{TP + FN}. \quad (19) $$

The area enclosed by ROC curve and PR curve are ROC_AUC and PR_AUC respectively.

Acc, F1, ROC_AUC and PR_AUC are in the range of [0,1], and MCC is in the range of [-1,1], with closer to 1 indicating better algorithm performance and vice versa.

### B. EXPERIMENTAL SETTINGS

In this experiment, a total of 20 GigaByte (GB) QAR data elements were collected from 2002 flights of an airplane during actual operation. Each flight obtained 40 time stamps during the landing stage through resampling, for a total of 120120 sample-points. In the constructed anomaly dataset, the ratio of long landing data to normal data is 1:17, indicating that this dataset is unbalanced. For the dataset processing, the original data set is first divided into the training set and test set through stratified sampling in the ratio of 8:2. Subsequently, the HFS algorithm is used to select the appropriate features in the training set, while the test set uses the feature matrix after the hybrid feature selection to ensure the objectivity of the test results. This experiment is conducted using a computer with Windows 10, 2.9 GHz, Intel i7 processor, RTX3080 GPU, python 3.8.10, and compiler as VSCodium environment using the jupyter notebook.

### C. FEATURE SELECTION ANALYSIS

1) Depiction of the result of feature selection

The feature number is reduced from 151 dimensions to 62 dimensions after the SRCC deletes the redundant features from the raw data. The following figure presents the comparison between the raw data and the SRCC features. In the heat map, the darker the color, the greater the correlation between the features. It can be observed from Figure 3 (a), that there are more dark areas in the heat map of the raw data, indicating that there are more redundant values in the raw data. Subsequently, it can be observed from Figure 3 (b) that the dark areas become significantly reduced after the SRCC processing, and the color becomes significantly lighter, indicating that the redundant values can be effectively eliminated after the raw data is subjected to the SRCC processing.

The XGB+RFE, a supervised feature selection method, is used to select features associated with anomalous events after removing redundant values using the UFS approach. This study adopts a cross-validation approach which can automatically select the best subset of features to ensure that the proposed algorithm automatically selects the appropriate features. Furthermore, the ROC_AUC score with 5-fold cross-validation is used as the objective function. Figure 4 presents the selected key feature parameters. It shows the number of different feature subsets and cross-validation scores; the point where the vertical line crosses the curve is the point with the largest cross score and the horizontal coordinate corresponding to the vertical line is the best feature subset selected. It can be observed from Table 4 that the algorithm selects the best feature subset number of 16. Figure 5 presents the hot map obtained after the HFS processing; it can be observed that there are fewer data dimensions and a lower correlation between features after the XGB+RFE processing when compared with the SRCC processing.

2) Feature selection performance comparison

This study compares different feature selection methods separately in terms of performance and computation time to
demonstrate the necessity of feature selection. Classical feature selection algorithms such as XGBoost, Random Forest (RF) [35], and Principal Component Analysis (PCA) [36] are selected and then compared with the algorithm proposed in this study. The features with a feature importance greater than 0.01 are selected when using RF in the feature selection. The decomposition vector with cumulative explained Variance greater than 95% is selected, while n_components is equal to 12, when using the PCA in feature dimensionality reduction. The classifier model used, is the LGBM algorithm without hyperparameter optimization, while the parameter values of LGBM are the original model values. Table 5 and Figure 6 present the comparison results.

The two aspects of performance and feature selection time can be compared by using the information presented in Table 5 and Figure 6. Firstly, in terms of performance, it can be observed by comparing the original data with the SRCC processed data, that Acc, F1, MCC, ROC_AUC, and PR_AUC do not significantly vary. Furthermore, the comparison of HFS with other feature selection algorithms demonstrates that the HFS method has the best performance with 98.04%, 78.20%, 0.7863%, 0.9955%, and 0.9408% for ACC, F1, MCC, ROC_AUC, and PR_AUC respectively. Particularly, the HFS effect is most pronounced for F1 scores and MCC values. The F1 scores of HFS are 4.59%, 3.037%, 2.84%, 1.13%, 60.35%, and 0.4% higher than the raw data, SRCC_FS, XGB_FS, RF_FS, PCA, and XGB+RFE_FS. The MCC values of HFS are 0.04, 0.0268, 0.0252, 0.0098, 0.5357, and 0.0032 higher than the raw data, SRCC_FS, XGB_FS, RF_FS, PCA, XGB+RFE_FS. Subsequently, the comparison of the feature selection time demonstrates that the PCA has the shortest feature selection time of 1.4 s, while XGB+RFE_FS is the algorithm with the longest feature selection time of 4301.8 s when compared with the time of XGB_FS, XGB+RFE_FS, and HFS, all three of which use the XGBoost feature selection algorithm, with 40.5 s, 4301.8 s, and 760.4 s, respectively. Therefore, the following conclusions can be drawn from the above data comparison:

1. For the data processed by using the SRCC when compared with the original data, the data dimension is significantly reduced, but the anomaly detection performance does not deteriorate;
TABLE 4. Best feature subset after the HFS processing

| Number | Feature name | Meaning                      |
|--------|--------------|------------------------------|
| 1      | _ALT_RADIO   | Radio altitude              |
| 2      | _CAS         | Computerd air speed         |
| 3      | _DIST_TO_THR_CONV | Distance from the runway threshold |
| 4      | _DME_DIST    | Distance from the DME       |
| 5      | _FLAPS       | Flaps angle                 |
| 6      | _FLAPS_2     | Flaps angle_2               |
| 7      | _GA_M_3D     | Glide angle                 |
| 8      | _GEO_SPEED   | Ground speed                |
| 9      | _HEADING     | Aircraft heading has        |
| 10     | _LATITUDE    | Aircraft latitude           |
| 11     | _LDG_LH      | Landing gear status         |
| 12     | _LOC         | Localizer deviation         |
| 13     | _LONGITUDE   | Aircraft longitude          |
| 14     | _N1          | Rotor speed of engine       |
| 15     | aSPDBRPOS    | Brake handle position       |
| 16     | _GLIDE       | Glide slope deviation       |

TABLE 5. Feature selection performance comparison

| FS methods | Number | Acc(%) | F1(%) | MCC   | FS time(s) |
|------------|--------|--------|-------|-------|------------|
| raw data   | 151    | 97.77  | 73.61 | 0.7463| 0          |
| SRCC_FS    | 62     | 97.82  | 75.17 | 0.7595| 40.5       |
| XGB_FS     | 46     | 97.83  | 75.36 | 0.7611| 9.4        |
| RF_FS      | 19     | 97.96  | 77.07 | 0.7765| 48.9       |
| PCA        | 12     | 95.98  | 17.85 | 0.2506| 1.4        |
| XGB+RFE_FS | 20     | 98.01  | 78.80 | 0.7831| 4301.8     |
| HFS        | 16     | 98.04  | 78.20 | 0.7863| 760.4      |

2. The HFS method of the SRCC and XGB+RFE significantly improves the anomaly detection performance when compared with the SRCC algorithm and reduces the feature selection time by 82.3% when compared with the XGB+RFE algorithm.

3. The HFS method has the least number of features and presents better performance than other feature selection algorithms.

D. ANOMALY DETECTION MODEL ANALYSIS

1) BO hyperparameter

The LGBM is first hyperparameterized using a BO algorithm when using the LGBM algorithm for LLE detection. The BO obtains the optimal hyperparameters of the LGBM model through iterative calculation by presetting the hyperparameter space, as shown in the following Table 6.

2) Anomaly detection model performance comparison and analysis

The idea is to choose the best among the best, which could be divided into two steps. At the beginning, some baseline models are chosen without optimization and the best model is selected through experiment. Then this best model is optimized with hyperparameters. In this study, ten groups of basic machine learning algorithms without optimization are selected for comparison experiments to compare the effectiveness of those models. The parameters of the ten machine learning classifiers are as follows: (1) The n_estimators of XGBoost (XGB) [20] is 100; (2) The min_samples_leaf of the decision tree (Tree) is 100; (3) RF [37] sets the number of n_estimators to 50; (4) The closest neighbor of KNN is 10; (5) SVM uses the poly kernel function; (6) The number of GBDT n_estimators is 500; and (7) The solver of LR is lbfgs, and the class_weight is balanced; (8) NB and MLP use default parameters. Table 7 and Figure 7 present the comparison results.

It can be observed from the above data that these tree-based model reach better performance than other machine learning models, indicating that these tree-based models are more suitable for aviation data anomaly detection; they include Tree<GBDT<RF<XGB<LGB. Among the ten groups of basic models, LGB has the best performance, whose Acc,F1,MCC,ROC_AUC and PR_AUC are 98.04%, 78.20, 0.9951% and 0.9408% respectively. Therefore, LGB is selected for further hyperparameter optimization. The F1, MCC, and PR_AUC values are significantly improved, increased by 20.35%, 19.47%, and 4.38% respectively by using BO for enhancing the LGB algorithm.

E. MODEL INTERPRETABILITY ANALYSIS

The SHAP tool is used to visualize the model feature importance and local dependency maps after using HFS and LGBMBO algorithm model on the training set. Feature importance is a global interpretive tool, which is defined as the magnitude of the SHAP value of all the feature value mapping, which indicates the degree of influence on LLEs.
### TABLE 6. BO hyperparameter and meaning

| Hyperparameter     | Initial value | Parameter space       | BO      | Meaning                                      |
|--------------------|---------------|-----------------------|---------|----------------------------------------------|
| num_leaves         | 31            | (25, 4000)            | 1223    | max number of leaves in one tree             |
| max_depth          | -1            | (5, 63)               | 38      | Maximum depth of the tree                    |
| lambda_L2          | 0             | (0.0, 0.05)           | 0.0307  | L1 regularization                            |
| lambda_L1          | 0             | (0.0, 0.05)           | 0.0489  | L2 regularization                            |
| min_data_in_leaf   | 20            | (100, 2000)           | 109     | minimal number of data in one leaf           |
| scale_pos_weight   | 1             | (1, 100)              | 19      | weight of labels with positive class         |

### TABLE 7. Anomaly detection model performance comparison

| Methods       | Acc(%) | F1(%) | MCC   | ROC_AUC | PR_AUC |
|---------------|--------|-------|-------|---------|--------|
| LGB+BO        | 99.40  | 94.11 | 0.9394| 0.9981  | 0.9820 |
| LGB           | 98.04  | 78.20 | 0.7863| 0.9955  | 0.9408 |
| XGB           | 97.85  | 75.35 | 0.7620| 0.9933  | 0.9151 |
| Tree          | 95.14  | 37.43 | 0.3807| 0.9273  | 0.4471 |
| RF            | 96.34  | 72.93 | 0.7304| 0.9913  | 0.9098 |
| KNN           | 94.85  | 10.82 | 0.1874| 0.6690  | 0.1622 |
| SVM           | 70.09  | 9.86  | 0.1576| 0.5158  | 0.0547 |
| GBDT          | 97.06  | 63.50 | 0.6572| 0.9859  | 0.8404 |
| LR            | 62.98  | 14.24 | 0.0982| 0.6547  | 0.1165 |
| NB            | 39.54  | 12.43 | 0.0849| 0.6388  | 0.0855 |
| MLP           | 94.87  | 20.31 | 0.2499| 0.7695  | 0.2518 |

In Figure 8, the vertical axis represents the name of each feature, and the horizontal axis represents the magnitude of the SHAP value. The feature names from top to bottom are obtained by ranking the importance of the predicted results. It can be observed that the distance from the start of the runway, ground speed, braking, compute airspeed, and heading angle are the five features that most affect the LLE.

The partial dependence plot (PDP) is a local interpretability tool which can be used to view each parameter’s marginal contribution to the predicted outcome. On the PDP, each point represents a row of data; the vertical axis represents the feature value, and the vertical coordinate indicates the effect of this feature value on the final landing result. A value greater than 0 on the vertical coordinate indicates that it promotes the LLE, and a value less than 0 indicates that this point in time is in a normal event. The horizontal coordinates represent the range of values of the feature. The whole graph shows the mapping correlation between the feature distribution and SHAP values.

In this study, the trend of each parameter is observed in two dimensions: altitude and runway plane, to conveniently observe the flight status of the aircraft during landing. The altitude is referenced by the radar altitude, considering the airport plane as the altitude origin. Furthermore, the runway plane considers the starting point of the runway as its origin. A distance greater than 0 indicates that the aircraft flies to the runway, while a distance less than 0 indicates that the aircraft flies from the starting point of the runway to the endpoint. These two dimensions are analyzed based on several key parameters which affect the flight state: ground speed, brake, glide slope deviation, and glide angle.

The yellow line in the Figure 9, Figure 10 and Figure 11 represents the regression line of the parameter distribution. The yellow line represents a trend throughout the data, and the black dashed line is the horizontal line where SHAP value
is 0. In these figures, if the yellow line is below the black dashed line, it indicates that the LLE will be suppressed at this time; otherwise, the occurrence of the LLE will be promoted. It can be observed from Figure 9 that the slope of the regression line is greater when the aircraft is at high altitude than at low altitude, and the higher the aircraft speed, the more likely is occurrence of an LLE. Therefore, maintaining the appropriate airspeed during the landing phase is crucial.

Brake value is an angle variable, and the greater the brake value, the greater the braking force. It can be observed from Figure 10 that the regression lines of the brake value distributions are typically lower than black dashed line, indicating that braking has an inhibitory effect on the LLE, which is consistent with physical logic. In fact, the brakes serve to mitigate the effects of the LLEs, i.e., to increase the actual length of the aircraft available on the runway and reduce the scheduling risk.

Figure 11 depicts the effect of glide slope deviation on the LLE. In fact, glide slope deviation and glide slope are the difference relation of vertical direction. Despite the multipath effect and near-field effect, the general trend of glide slope deviation can be observed. It can be observed from Figure 11 that, if glide slope deviation is less than 0 during the flare-out phase, i.e., the yellow line is above the black dashed line, it indicates that the aircraft is below the ILS glide path, which suppresses LLEs, and vice versa. The actual glide angle of the aircraft is related to the altitude of the aircraft and the distance from the ILS glide-path localizer, so the pilot should be pay attention to the aircraft altitude and the distance from glide-path localizer.

Figure 12 illustrates the glide angle of the aircraft, from which it can be observed that most aircraft land along a 2.8-degree angle of glide. It is consistent with the rule in the flight manual that the angle of glide of the aircraft typically lies...
between 2.5 and 4 degrees. Therefore, the pilot must ensure that the glide angle lies within the appropriate range during the approach phase. Essentially, a low glide angle may result in an LLE.

V. CONCLUSION

Risk event detection and cause analysis are necessary and sufficient for flight safety and accident prevention. Risk event detection is a prerequisite for risk cause explanation, and a model can only convincingly explain the causes of risk with a high detection rate. Conversely, risk cause explanation is a supplement to risk event detection and can reflect the practical value of detection. In this paper, the proposed workflow makes a trade-off between the ability to detect anomalous events and the inference of their causality.

The proposed HFS-LGBMBO model reduces computational cost and achieves better performance. Besides, with the help of SHAP, it proves that LLEs can be avoided during the landing phase by maintaining the appropriate descent speed, the appropriate aircraft altitude, and proper descent angle, which is consistent with the requirements of aviation regulations and pilot training manuals.

Despite the strengths of proposed workflow in this study, some limitations should be noted. The current work does not allow for some external environmental factors, such as weather and runway conditions, which may pose a risk of misjudgment. Furthermore, the lack of temporal correlation among events makes it difficult to depict a causal chain of accidents. In future work, the aviation daily monitoring data will be used to explore more challenging scenarios, such as the assessment of the pilot’s flight quality and the construction of aviation abnormal event knowledge graphs for causal inference of events.

REFERENCES

[1] B. C. Airplanes, “Statistical summary of commercial jet airplane accidents, worldwide operations 1959-2020.”, Boeing Corp., Chicago, USA, Tech. 2021.
[2] AIRBUS, “A statistical analysis of commercial aviation accidents 1958—2019.”, 2019. [Online]. Available: https://accidentstats.airbus.com/statistics/accident-categories.
[3] ICAO, ‘safety report (2020),” 2020.
[4] L. Wang, C. Wu, R. Sun, “An analysis of flight quick access recorder (qar) data and its applications in preventing landing incidents,” Reliability Engineering and System Safety 127 (2014) 86–96.
[5] S. Zhou, Y. Zhou, Z. Xu, W. Chang, Y. Cheng, “The landing safety prediction model by integrating pattern recognition and markov chain with flight data”, Neural Computing and Applications 31 (1) (2019) 147–159.
[6] ICAO, “Manual on Flight Data Analysis Programmes (FDAP) Doc 10000AN/501”, Standard, ICAO (2014).
[7] FAA, “AC 120-82-Flight Operational Quality Assurance”, Standard, FAA (April 2004).
[8] L. Li, R. Hansman, R. Palacios, R. Welsch, “Anomaly detection via a gaussian mixture model for flight operation and safety monitoring”, Transportation Research Part C: Emerging Technologies 64 (2016) 45–57. doi: 560 10.1016/j.trc.2016.01.007.
[9] J. Oehling, D. J. Barry, “Using machine learning methods in airline flight data monitoring to generate new operational safety knowledge from existing data,” Safety science 114 (2019) 89–104. doi:10.1016/j.ssci.2018.12.018.
[10] K. Sheridan, T. G. Puranik, E. Mangortey, O. J. Pinon-Fischer, M. Kirby, D. N. Mavris, “An application of dbscan clustering for flight anomaly detection during the approach phase,” in: AIAA Scitech 2020 Forum, 2020, p.1851.
[11] F. Mannering, C. R. Bhat, V. Shankar, M. Abdel-Aty, “Big data, traditional data and the tradeoffs between prediction and causality in highway-safety analysis,” Analytic methods in accident research 25 (2020) 100113.
[12] S. Das, B. L. Matthews, A. N. Srivastava, N. C. Oza, “Multiple kernel learning for heterogeneous anomaly detection: algorithm and aviation safety case study”, in: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, 2010, pp. 47–56.
[13] S. Das, B. L. Matthews, R. Lawrence, “Fleet level anomaly detection of aviation safety data”, in: 2011 IEEE Conference on Prognostics and Health Management, IEEE, 2011, pp. 1–10.
[14] L. Li, S. Das, R. John Hansman, R. Palacios, A. N. Srivastava, “Analysis of flight data using clustering techniques for detecting abnormal operations”, Journal of Aerospace information systems 12 (9) (2015) 587–598.
[15] T. G. Puranik, H. Jimenez, D. N. Mavris, “Utilizing energy metrics and clustering techniques to identify anomalous general aviation operations”, in:AIAA Information Systems-AIAA Infotech@ Aerospace, 2017, p. 0789.
[16] I. Melnyk, A. Banerjee, B. Matthews, N. Oza, “Semi-markov switching vector autoregressive model-based anomaly detection in aviation systems,” in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 1065–1074.
[17] C. Tong, X. Yin, S. Wang, Z. Zheng, “A novel deep learning method for aircraft landing speed prediction based on cloud-based sensor data,” Future Generation Computer Systems 88 (2018) 552–558.
[18] T. G. Puranik, N. Rodriguez, D. N. Mavris, “Towards online prediction of safety-critical landing metrics in aviation using supervised machine learning,” Transportation Research Part C: Emerging Technologies 120 (2020) 102819.
[19] M. Memarzadeh, B. Matthews, I. Avrekh, “Unsupervised anomaly detection in flight data using convolutional variational auto-encoder,” Aerospace 7 (8) (2020) 115.
[20] M. Rey, D. Aloise, F. Soumis, R. Pieupegue, ”A data-driven model for safety risk identification from flight data analysis,” Transportation Engineering 5 (2021) 100087.
[21] L. Wang, Y. Ren, C. Wu, “Effects of flare operation on landing safety: A study based on anova of real flight data”, Safety science 102 (2018) 14–25.
[22] D. J. Barry, "Estimating runway veer-off risk using a bayesian network with flight data", Transportation Research Part C: Emerging Technologies 128 (2021) 103180.

[23] F. A. Administration, "Airplane Flying Handbook (FAA-H-8083-3A)," Skyhorse Publishing Inc., 2011.

[24] CAAC, "The implementation and management flight operational quality assurance (FOQA) [S]." AC-121/135-FS-2012-45R1, Standard, CAAC (June 2015).

[25] S. Solorio-Fernández, J. A. Carrasco-Ochoa, J. F. Martínez-Trinidad, "A review of unsupervised feature selection methods," Artificial Intelligence Review 53 (2) (2020) 907–948.

[26] P. Schoder, C. Boer, L. A. Schwart, "Correlation coefficients: appropriate use and interpretation.", Anesthesia and Analgesia, 126(5)(2018), 1763-1768.

[27] Armstrong, Richard A. "Should Pearson’s correlation coefficient be avoided?" Ophthalmic and Physiological Optics 39.5 (2019): 316-327.

[28] S. H. Huang, "Supervised feature selection: A tutorial.", Artif. Intell. Res. 4 (2) (2015) 22–37.

[29] T. Chen, C. Guestrin, "Xgboost: A scalable tree boosting system," in: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016, pp. 785–794.

[30] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," Annals of statistics (2001) 1189–1232.

[31] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, "Gene selection for cancer classification using support vector machines," Machine learning 46 (1) (2002)389–422.

[32] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," Advances in neural information processing systems 30 (2017) 3146–3154.

[33] M. Pelikan, D. E. Goldberg, E. Cantú-Paz et al., "Boa: The bayesian optimization algorithm", in: Proceedings of the genetic and evolutionary computation conference GECCO-99, Vol. 1, Citeseer, 1999, pp. 525–532.

[34] M. T. Ribeiro, S. Singh, C. Guestrin, ""why should i trust you?" explaining the predictions of any classifier", in: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 1135–1144.

[35] J. Rogers, S. Gunn, "Identifying feature relevance using a random forest," in: International Statistical and Optimization Perspectives Workshop” Subspace, Latent Structure and Feature Selection”, Springer, 2005, pp. 173–184.

[36] H. Abdi, L. J. Williams, "Principal component analysis, Wiley interdisciplinary reviews.", computational statistics 2 (4) (2010) 433–459.

[37] Li, J., Zhang, H., Yang, J. (2019, October). Discrimination Model of QAR High-Severity Events Using Machine Learning. In International Conference on Intelligent Science and Big Data Engineering (pp. 430-441). Springer, Cham.