

## Article

# Automated Anomaly Detection and Causal Analysis for Civil Aviation Using QAR Data

Xin Dang <sup>†</sup>, Congcong Hua <sup>†</sup> and Chuitian Rong <sup>\*,†</sup> 

School of Computer Science and Technology, Tiangong University, Tianjin 300387, China; xindang@tiangong.edu.cn (X.D.); 2231081199@tiangong.edu.cn (C.H.)

\* Correspondence: chuitian@tiangong.edu.cn

<sup>†</sup> These authors contributed equally to this work.

**Abstract:** Flight Operations Quality Assurance (FOQA) is an internationally recognized solution to ensure the safety of civil aircraft flights based on Quick Access Recorder (QAR) data. The traditional approach to anomaly detection in civil aviation is to detect the over-limit values of monitoring parameters for each monitoring event based on the standards issued by civil aviation authorities. Usually, for each anomaly detection operation routine, this only works for one monitoring event. Furthermore, the causal analyses for the detected anomaly events are based on the relevant worker's expertise. In order to improve the efficiency of FOQA, this paper proposes an automated anomaly detection and causal analysis method called MAD-XFP. Due to the unique industry characteristics of QAR data and the requirements of FOQA, feature engineering and hyper-parameter optimization techniques are utilized to enhance the machine learning model. The proposed method can monitor multiple events in one routine and provide a causal analysis. In the causal analysis process, the Shapley additive interpretation method is applied to produce analysis report for detected anomalies. Experimental evaluations are conducted on real civil aviation datasets. The experimental results show that the proposed method can efficiently and automatically detect different abnormal events with high precision in the approach phase and produce preliminary causal analysis.

**Keywords:** anomaly detection; QAR data; civil aviation; causal analysis



Academic Editor: Wei Huang

Received: 22 January 2025

Revised: 14 February 2025

Accepted: 17 February 2025

Published: 20 February 2025

**Citation:** Dang, X.; Hua, C.; Rong, C. Automated Anomaly Detection and Causal Analysis for Civil Aviation Using QAR Data. *Appl. Sci.* **2025**, *15*, 2250. <https://doi.org/10.3390/app15052250>

**Copyright:** © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Flight Operations Quality Assurance (FOQA) [1] is an important aspect of airline safety management and has been widely recognized by the global civil aviation industry. By analyzing daily flight data collected by digital flight data recorders, these data are useful for detecting unsafe events and potential hazards during the flight process, optimizing air traffic control procedures, and improving flight quality. The Quick Access Recorder (QAR) is an important instance of digital equipment that captures a large amount of parameters during aircraft flights, including crew actions, aircraft status, engine performance, fuel consumption, and additional pertinent data. This recorded information [2] forms the essential data foundation (called QAR data) for FOQA.

However, QAR data record a large number of flight parameters (over 2200 parameters for Airbus A320), and for each parameter, its collecting frequency, accuracy, and data format are not consistent. In addition, there are many data quality issues such as noise, anomalies, and missing data in collected QAR data, which create great challenges for downstream applications such as data mining, model training, abnormal flight detection, and casual analysis. QAR data need to be cleaned before further analysis is performed.

On the other hand, the civil aviation industry in China has always been at the forefront of safety management. According to the 2022 China Civil Aviation Administration Statistical Bulletin, the ten-year rolling value of the major accident rate is 0.011 (per million flights), which is significantly lower than the average global industry. The vast majority of QAR data are within normal ranges. The quantity and proportion of data parameters related to certain anomaly events are relatively small, demonstrating an unbalanced data distribution. Overall, the unbalanced distribution and the low data quality, along with the unique industry characteristics of QAR data, present considerable challenges for FOQA.

An aircraft usually goes through several stages to complete a flight, including taxi in and take off, climb, cruise, descent, approach, and landing. Different aircraft types usually have different classification standards. The approach phase refers to the process during which an aircraft systematically descends from its cruising altitude, transitioning towards to the destination airport. During the approach phase, the flight operation is susceptible to the influence of multiple factors, including adverse weather conditions, crew actions, and technical malfunctions. These influences may contribute to a high likelihood of over-limit events [3]. The approach phase is the most prone to anomalies throughout the entire flight process. So, this work focuses on anomaly detection in the approach phase.

Traditional flight anomaly detection relies on the monitoring items and standards in the FOQA guidelines issued by the Civil Aviation Administration of China (CAAC) to identify over-limit parameter values. In this work, we focus mainly on the monitoring events involved in the approach phase, as shown in Table 1. We compare the values of monitoring parameters within the monitoring point range with the out-of-limit values to determine whether they belong to the categories of severe deviation, mild deviation, or normal data. The meaning of the monitoring parameters is shown in Table 2.

**Table 1.** FOQA monitoring items and requirements for Airbus A320 (Airbus, Leiden, The Netherlands) (in approach phase) [1].

| Monitoring Items          | Monitoring Parameters | Monitoring Point                | Deviation Limit Value       |                             |
|---------------------------|-----------------------|---------------------------------|-----------------------------|-----------------------------|
|                           |                       |                                 | Mild Deviation              | Severe Deviation            |
| Large Decent Rate         | IVV                   | 610 m (2000 ft)~305 m (1000 ft) | >457 m/min<br>(1500 ft/min) | >549 m/min<br>(1800 ft/min) |
|                           |                       | 305 m (1000 ft)~152 m (500 ft)  | >396 m/min<br>(1300 ft/min) | >457 m/min<br>(1500 ft/min) |
|                           |                       | 152 m (500 ft)~15 m (50 ft)     | >335 m/min<br>(1100 ft/min) | >396 m/min<br>(1300 ft/min) |
| Large Approach Roll Angle | ROLL                  | 457 m (1500 ft)~152 m (500 ft)  | >30°                        | >35°                        |
|                           |                       | 152 m (500 ft)~61 m (200 ft)    | >15°                        | >20°                        |
|                           |                       | 61 m (200 ft)~15 m (50 ft)      | >8°                         | >10°                        |
| High Approach Speed       | IAS, VAPP             | 152 m (500 ft)~15 m (50 ft)     | >(VAPP + 15) kn             | >(VAPP + 20) kn             |
| Low Approach Speed        | IAS, VAPP             | 305 m (1000 ft)~15 m (50 ft)    | <(VAPP – 5) kn              | <(VAPP – 10) kn             |
| ILS Glide Slope Deviation | GLIDE_DEVC            | below 305 m (1000 ft)           | >1.0 point                  | >1.5 point                  |
| ILS Heading Deviation     | LOC_DEVC              | below 305 m (1000 ft)           | >1.0 point                  | >1.5 point                  |
| Low-Altitude High Speed   | IAS                   | below 762 m (2500 ft)           | >230 kn                     | >250 kn                     |

However, the traditional working manner carries several inherent risks:

1. It can only detect one anomaly in one operation routine. It lacks effectiveness in identifying composite abnormal events that involve multiple anomalies.
2. It does not support automated causal analysis of abnormal events. It relies heavily on expert experience to identify abnormal events and generate causal analysis reports.

**Table 2.** Classification and description of QAR parameters.

| Categories                 | Parameters   | Units  | Description                |
|----------------------------|--------------|--------|----------------------------|
| Monitoring Parameters      | IAS          | KTS    | Corrected Airspeed         |
|                            | VAPP         | KTS    | Approach Reference Speed   |
|                            | GLIDE_DEVC   | DDM    | ILS Glideslope Deviation   |
|                            | LOC_DEVC     | DDM    | ILS Heading Deviation      |
|                            | ROLL         | DEG    | Roll Angle                 |
|                            | IVV          | FT/MIN | Decent Rate                |
| Pilot Operating Parameters | TLA          | DEG    | Throttle Resolver Angle    |
|                            | SEL_COURSE   | —      | Selected Course            |
|                            | FF1          | PPH    | Fuel Mass Flow Rate ENG.1  |
|                            | VB11         | %      | N1 VIB ENG.1 BRG1 or TRF   |
| Flight State Parameters    | HEIGHT       | FT     | Corrected Height           |
|                            | RALTC        | FT     | Radio Height               |
|                            | ALT_STDC     | FT     | Altitude Standard Recorded |
|                            | VRTG         | G      | Normal Acceleration        |
|                            | LATG         | G      | Lateral Acceleration       |
|                            | LONG         | G      | Longitudinal Acceleration  |
|                            | SPOIL_POS    | —      | Spoiler Position           |
|                            | VOR_FRQ      | HZ     | Vor Frequency              |
|                            | N11          | %      | Low Rotor Speed ENG.1      |
|                            | FLAPC        | DEG    | Flap Actual Position       |
|                            | HEAD         | DEG    | Magnetic Heading           |
|                            | PITCH        | DEG    | Pitch Angle                |
|                            | RUDD         | DEG    | Rudder Position            |
|                            | FPA          | DEG    | Flight Path Angle          |
|                            | VREF         | KTS    | Landing Reference Speed    |
| Environmental Parameters   | WIN_DIR      | DEG    | Wind Direction             |
|                            | WIN_SPD      | KN     | Wind Speed                 |
|                            | LONPC        | DEG    | Present Position Longitude |
|                            | LATPC        | DEG    | Present Position Latitude  |
| Indicating Parameter       | TIME_R       | —      | Time                       |
|                            | FLIGHT_PHASE | —      | Flight Phase               |

To address the increasing demands of FOQA, leveraging artificial intelligence and machine learning technologies to improve the efficiency and quality of FOQA has become a priority for civil aviation.

EXtreme Gradient Boosting(XGBoost) [4] is an ensemble machine learning algorithm based on decision trees, widely applied for regression and classification tasks. This model has notable advantages in handling anomalies, balancing data distribution, and providing interpretability [5,6]. However, there are still several challenges to apply it to QAR data in practice. Due to the inherent characteristics of QAR data, data pre-processing is essential and must be tailored to the specific requirements of the anomaly detection task. Additionally, the XGBoost model involves over 10 parameters, including general parameters, tree-specific parameters, learning task parameters, etc. Consequently, optimizing the model to adapt to anomaly detection in QAR data is also a complex task.

To achieve intelligent and automated flight anomaly detection and causal analysis in civil aviation, we propose a comprehensive method, MAD-XFP. This method is based on XGBoost, incorporating automated feature engineering techniques and AutoML model optimization. The proposed method can effectively identify various abnormal events and composite events in one operation routine, facilitating automated causal analysis.

In response to data quality issues, this paper first supplements missing values using the nearest neighbor principle. Then, we perform dimensionality reduction on the data and standardize the sampling rate. By combining the industry characteristics of QAR data with feature engineering techniques, new features can be generated to improve the accuracy of the results. In response to the shortcomings of traditional detection methods, we use machine learning and automated models to classify data and calculate contribution rates to achieve automated anomaly detection and intelligent factor analysis of civil aviation flight data.

In summary, the main contributions of this work are listed as follows:

1. An automated anomaly detection method is proposed, tailored to the unique characteristics of QAR data and the requirements of FOQA. This model can detect various abnormal events in one operation routine.
2. In order to achieve efficient convergence rates and lower over-fitting risks, the advanced feature engineering techniques and hyper-parameter auto-tuning techniques are utilized.
3. An automated and intelligent causal analysis technology for abnormal events is proposed to address the limitations of the traditional FOQA working method.
4. Method evaluation experiments are conducted on real QAR datasets. The evaluation and comparison results are presented with in-depth analysis.

## 2. Related Works

QAR data is essentially multidimensional time series data. There are many anomaly detection methods for multidimensional time series, which can be roughly divided into three categories: statistical methods, machine learning methods, and deep learning methods.

The statistics-based method is the earliest and most mature method for detecting anomalies in multidimensional time series. In past development and application [7,8], it has been found that the statistical method has many shortcomings, such as an inability to consider the order of events and difficulty in determining the boundaries of anomalies. These are fatal drawbacks for anomaly detection in time series.

In the big data era, with the rapid development of artificial intelligence, machine-learning-based technologies have been applied to solve anomaly detection problems in time series. Compared to deep learning, traditional machine learning algorithms typically have better interpretability in explaining how the model makes predictions or classifications. This is very important for causal analysis of QAR data anomalies.

Hu et al. [9] proposed a meta-feature-based anomaly detection method (MFAD) for identifying anomalous states in univariate and multivariate time series. The proposed MFAD method not only identifies local abnormal states in the original time series, but also reduces computational complexity. Lee et al. [10] proposed a semi-supervised anomaly detection algorithm using probabilistic labeling (SAD-PL) for unlabeled data, which can ensure stable detection efficiency regardless of changes in the ratio of normal to abnormal data in unlabeled data. Akhmedova et al. [11] proposed a hybrid clustering approach for anomaly detection based on fuzzy logic and evolutionary computation. This method uses a portion of known categories to label data and divides them into a given number of clusters, then attempts to determine whether new instances that may not have been labeled belong to known or new categories, or whether they are anomalies. It can find new classes outside of anomalies while achieving better classification accuracy. Although the above anomaly detection algorithms can effectively detect anomalies through training on large amounts of data, QAR data, as multidimensional time-series data with significant industry characteristics, have the characteristics of heterogeneity and non-periodicity, which make the above methods unable to adapt well. Sun et al. [12] proposed a new model for anomaly

detection based on QAR data, which utilizes autoencoders and the DBSCAN algorithm to extract features from QAR data, complementing traditional anomaly detection methods. Qiu et al. [13] proposed a three-stage QAR data anomaly detection algorithm considering complex spatial–temporal correlations in QAR data. They also proposed a dual K-means clustering algorithm that can automatically select the hyper-parameter K, which achieved good anomaly detection for QAR data.

In summary, machine learning-based methods can handle the problems of anomaly detection and causal analysis of time series data. However, these methods used for QAR data mostly focus on data anomalies without considering the requirements of FOQA and cannot produce the causal analysis report automatically. In order to solve the above issues and improve the efficiency of anomaly detection for FOQA, we proposed a machine learning-based method incorporating feature engineering and hyper-parameter optimization techniques.

### 3. Preliminary Preparation for Data and Causal Analysis

In this section, we first formalize the definition of the problem to be solved in this paper. Then, the details of data preprocessing techniques applied to QAR data are listed, including data cleaning and data balancing. Lastly, we elaborate on the preparations undertaken for causal analysis.

#### 3.1. Problem Statement

The training data are defined as  $X_{train} = \{x_1, x_2, \dots, x_L, Y\}$ , including the manual labels, where  $L$  is the number of attributes in the input data. For each timestamp  $t$ ,  $x^t \in R^L$  is an  $L$ -dimensional vector consisting of  $L$  numerical parameter values.  $x^t$  is defined as  $x^t = \{x_1^t, x_2^t, \dots, x_L^t\}$ , in which  $x_i^t$  is the value of the  $i$ -th parameter at timestamp  $t$ .

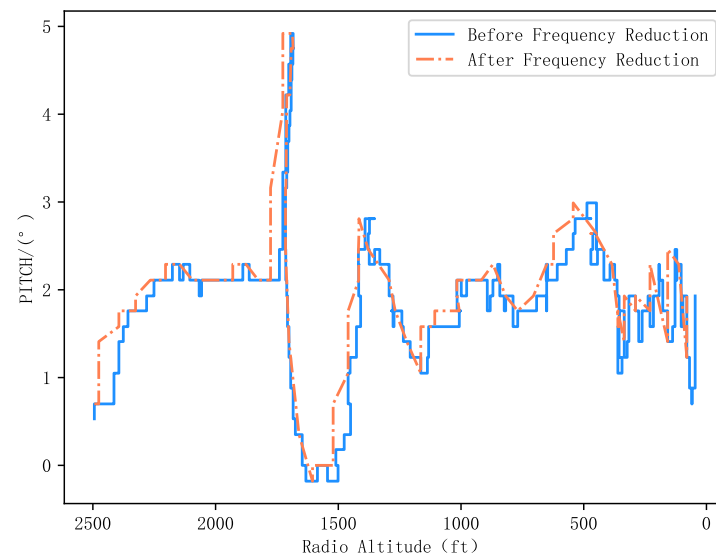
The aim of the work is to train a machine learning model  $M$  based on  $X_{train}$  to detect different anomalies from QAR data. Specifically, given the test data  $X_{test} = \{x_1, x_2, \dots, x_L\}$  input to  $M$ , it will produce an output vector  $y \in R^L$ , where  $y_t \in \{0, 1, \dots, k\}$  indicates the type of anomaly.

#### 3.2. Data Preprocessing

##### 3.2.1. Data Cleaning

As the parameters in QAR data are collected at different frequencies, there are many gaps without values when integrating them into one dataset. To stabilize the data, the first step of data cleaning is to fill up the gaps with the adjacent values. Then, the related parameters will be extracted from the approach phase. Finally, in order to eliminate redundant data and make data changes smoother, uniformly reducing the data to 1 Hz, we perform a down-sampling on the extracted data and make the maximum value of the sampling interval as the value for that second after reducing the sampling rate. The function of step is shown in Figure 1. Define a parameter  $x_i^{(m)} = (x_i^j, x_i^{j+1}, \dots, x_i^{j+m})$ , where  $i \in \{1, 2, \dots, L\}$  represents an attribute column, and  $m$  is the time step. The down-sampling conversion process is as follows:

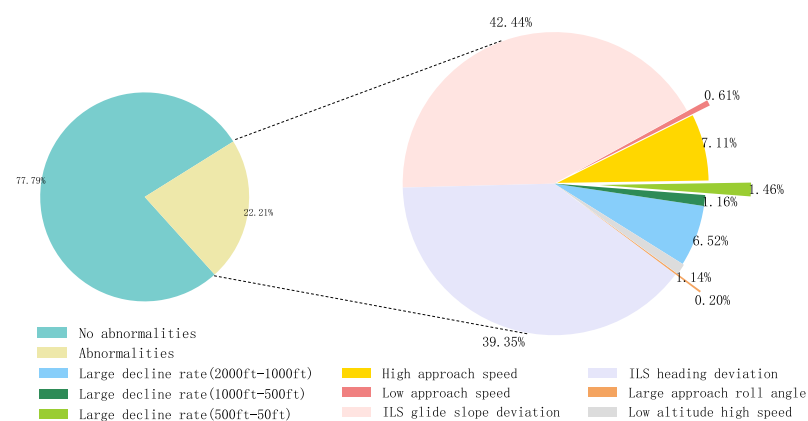
$$(x_i^j, x_i^{j+1}, \dots, x_i^{j+m}) = \max(x_i^j, x_i^{j+1}, \dots, x_i^{j+m}) \quad (1)$$



**Figure 1.** Down-sampling results of pitch angle parameters.

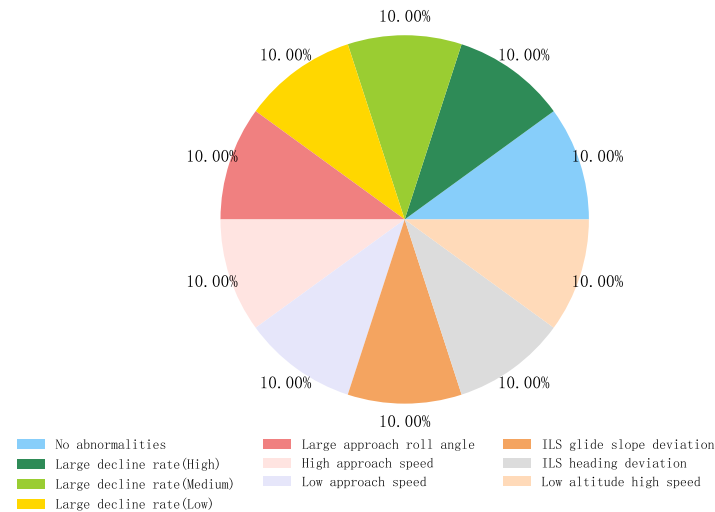
### 3.2.2. Data Balancing

The experimental data used in this paper are provided by the Civil Aviation Administration of China and are real flight data. Although the probability of occurrence of an anomaly is highest during the approach phase of a flight, it still accounts for a small portion compared to normal data. Details are shown in Figure 2; it can be seen that abnormal data account for 22.21%. Among them, *ILS Glide Slope Deviation* accounts for the largest proportion, accounting for 42.44% of the entire anomaly, while *Large Approach Roll Angle* accounts for the smallest proportion, only 0.20% of all anomalies. In conclusion, there is an extreme imbalance in the proportion of various anomalies in the approach phase.



**Figure 2.** Distribution of different anomalies in QAR data during approach phase.

For multi-class classification models, if there are insufficient training data for a certain category, the model will learn too little “information” about that category, ultimately resulting in difficulty in determining which category the data belong to. So, data imbalances will have a significant negative impact on the final classification results and also affect the accuracy of the model. Therefore, this paper uses the Borderline SMOTE algorithm [14] to balance the data, which only uses minority class samples on the boundary to synthesize new samples. The result after data balancing is shown in Figure 3.



**Figure 3.** Distribution of samples using data balance techniques.

### 3.3. Preparations for Causal Analysis

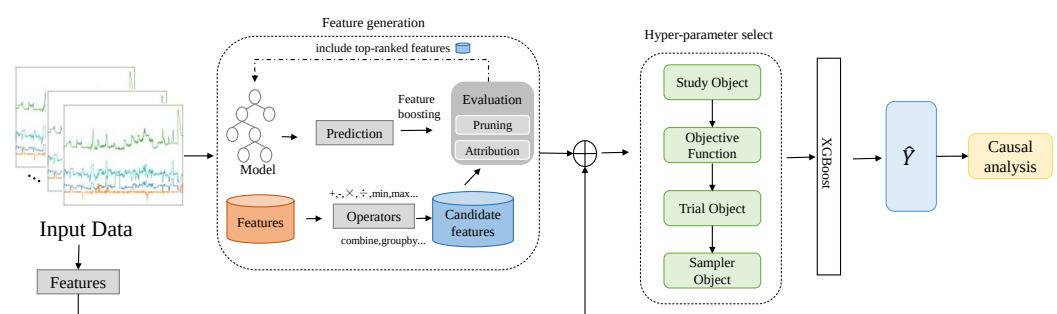
The workflow for flight anomaly event detection and causal analysis in the approach phase is as follows:

First, we classify the causes of abnormalities into two categories—human factor and environmental factor. Specifically, the initial determination of whether the abnormality was caused by a human factor or environmental factor is based on the contribution of the features of the abnormal events. The parameters and parameter categories used for abnormality detection and causal analysis in the approach phase are shown in Table 2.

## 4. Multi-Type Anomaly Detection in Civil Aviation

In this section, we provide a detailed description of our proposed **Multi-type Anomaly Detection** method based on XGBoost enhanced with Feature engineering and hyper Parameter optimization techniques, called MAD-XFP.

MAD-XFP mainly overcomes the shortcomings of the XGBoost model that make it not suitable for the inherent characteristics of QAR data by incorporating feature engineering and parameter optimization. By doing this, our method can identify multi-type anomalies in one operation routine. Specifically, MAD-XFP first performs feature engineering on each input sequence to enrich the representation of data features. Then, automated parameter optimization is utilized to set optimized values for parameters of the model. Finally, a decision tree is constructed using XGBoost to obtain the final prediction output. Figure 4 shows the general framework of MAD-XFP. The operation of the method will be described in detail below.



**Figure 4.** Overview of MAD-XFP.

#### 4.1. Model Design

The XGBoost algorithm is a machine learning algorithm based on Gradient Boosting Tree, which uses a technique called tree boosting to construct models. XGBoost, as an efficient and scalable gradient boosting algorithm, performs well in handling multiple classification problems. For multiple classification tasks, XGBoost trains a decision tree for each category, synthesizes the outputs of each sub-tree, and finally outputs the classification results for the samples.

Define the training dataset as  $\mathcal{D}$ , and the number of parameters in  $\mathcal{D}$  as  $d$ .

$$(X_{i,j}, y_i)_{i=1}^{\mathcal{D}} \quad (2)$$

where  $X_{i,j} = \{x_{1,j}, x_{2,j}, \dots, x_{d,j}\}$ ,  $j \in [1, L]$ ,  $L$  is the number of attribute columns in  $\mathcal{D}$ , and  $y_i \in \{0, 1, \dots, N-1\}$ .

If  $y_{i,n}$  represents the true value of the  $i$ -th sample on the  $n$ -th class, then the following statement can be made:

$$y_{i,n} = \begin{cases} 1 & y_i = n \\ 0 & otherwise \end{cases} \quad (3)$$

and then  $\hat{y}_{i,n}$  represents the predicted value of the  $i$ -th sample on class  $n$ .

XGBoost is an additive model of boost, where each iteration optimizes only the sub models in the current step. The step  $m$  can be defined as follows:

$$F_m(x_i) = F_{m-1}(x_i) + f_m(x_i) \quad (4)$$

where  $f_m(x_i)$  is the sub-model of the current step, and  $F_{m-1}(x_i)$  gives the first  $m-1$  sub-models that have completed training and are fixed. Therefore, its objective function is defined as follows:

$$L^{(t)} = \sum_{i=1}^M l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \quad (5)$$

where  $L^{(t)}$  represents the objective function of the  $t$ -th anomaly category,  $l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i))$  represents the loss function of the  $i$ -th sample, and  $\Omega(f_t)$  denotes the value of the penalty term to prevent overfitting.

In the method proposed in this paper, the Optuna module uses weighted F1 values as the objective function, and we use the cross-entropy loss function in the XGBoost model. The objective function of the MAD-XFP method is as follows:

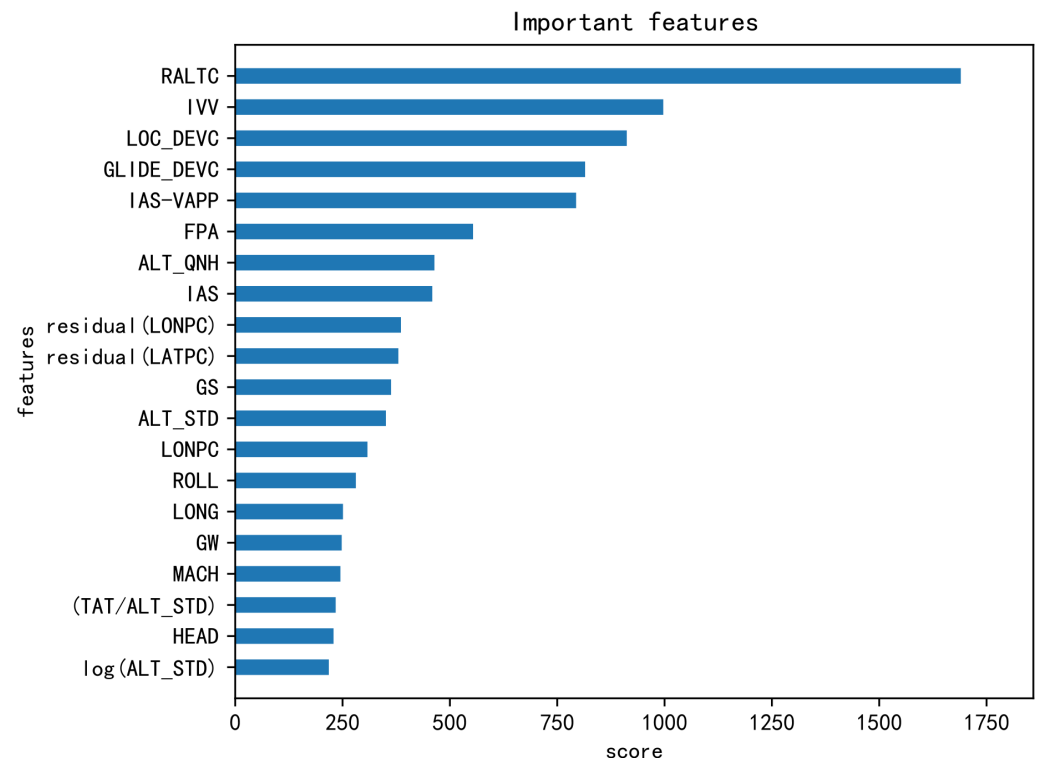
$$\begin{aligned} Objective &= L^{(t)} - \sum_{i=1}^n \sum_{j=1}^k w_{i,j} \log(p_{i,j}) \\ &= \Omega(f_t) - \sum_{i=1}^M \sum_{n=0}^{N-1} (y_{i,n} + w_{i,n}) \log(\hat{p}_{i,n}) \end{aligned} \quad (6)$$

where  $N$  is the number of the category of anomaly events,  $y_{i,n}$  represents whether the true label of the  $i$ -th sample belongs to the  $n$ -th class, and  $\hat{p}_{i,n}$  represents the predicted probability that the  $i$ -th sample belongs to the  $n$ -th class.

$$w_{i,n} = \frac{(1 + \beta 2) \cdot \text{Precision}_n \cdot \text{Recall}_n}{\beta 2 \cdot \text{Precision}_n + \text{Recall}_n} \quad (7)$$

where  $\beta$  is a weight parameter, and its default value is one.  $\text{Precision}_n$  and  $\text{Recall}_n$  are, respectively, the precision and recall of the  $n$ -th class.

The XGBoost model also provides a quantitative way to demonstrate the importance of features that affect the model, which measures the value of features in the construction of decision trees for model improvement. Figure 5 shows the characteristic parameters calculated by the XGBoost model that are very relevant to the abnormal events in the approach phase. The quantization sorting is based on the average gain of each feature on the predicted results in all trees and reflects the splitting ability of the feature at each node, because the stronger the splitting ability of a feature at each node, the greater its contribution to the final prediction result.



**Figure 5.** Distribution of feature importance of MAD-XFP (top 20).

Overall, the improved XGBoost model proposed in this paper is capable of establishing complex decision tree structures and performs well in capturing complex relationships in data, improving classification accuracy.

#### 4.2. Anomaly Detection

The XGBoost model iteratively builds multiple weak classifiers, each of which classifies the data and then combines them into a strong classifier. The final output is the result of the joint action of multiple classifiers. For a sample, the model gives  $K$  values, where  $k$  is the number of categories. The sum of these  $k$  values is one. In this paper, we choose the category with the highest probability as the prediction result. After the detection of the MAD-XFP method, the output results,  $\hat{Y}$ , represent different abnormal events, and the relationship between each abnormal event and the abnormal label is shown in the following Table 3.

During the training phase, we record the probability of abnormal Event  $B$  occurring under the conditions of abnormal Event  $A$ , denoted as  $P(B|A)$ . In the validation phase, we consider the case where the probability of the sample being predicted as a certain category is the highest as the final result, and record the result as Event  $A$ , with a probability of occurrence of  $P(A)$ . Then, according to the conditional probability formula  $P(AB) = P(B|A)P(A)$ , we can calculate the probability of both Event  $A$  and Event  $B$

occurring simultaneously, denoted as  $P(AB)$ . If  $P(AB) > \theta$  ( $0 < \theta < 1$ ), we believe that event A and event B occurred simultaneously in this sample. At this point, we have realized the identification of abnormal events of multi-event coupling in QAR data.

**Table 3.** Information of anomaly label.

| Abnormal Event                      | Anomaly Label |
|-------------------------------------|---------------|
| No Abnormal Event Occurred          | 0             |
| Large Decent Rate (2000 ft~1000 ft) | 1             |
| Large Decent Rate (1000 ft~500 ft)  | 2             |
| Large Decent Rate (500 ft~50 ft)    | 3             |
| Large Approach Roll Angle           | 4             |
| High Approach Speed                 | 5             |
| Low Approach Speed                  | 6             |
| ILS Glide Slope Deviation           | 7             |
| ILS Heading Deciation               | 8             |
| Low-Altitude High Speed             | 9             |

## 5. Optimization Techniques

### 5.1. Automated Feature Generation

Automated feature engineering aims to create candidate features automatically from a dataset and select several optimal features for training.

In this section, based on aircraft flight field knowledge, a feature interaction method is used to construct new composite features, focusing on the dataset of this application scenario. Using OpenFE (<https://github.com/IIIS-Li-Group/OpenFE>, (accessed on the 10 November 2024)), it is more efficient to achieve the construction of composite features. Specifically, it includes the following aspects:

1. Interaction between time and geographic features: the parameters involved are *TIME\_R*, *LONPC*, *LATPC*, *FLIGHT\_PHASE*.
2. Interaction between meteorological conditions and flight conditions: parameters involved are *RUDD*, *WIN\_SPD*, *TAT*, *ALT\_STD*.
3. Flight dynamic feature interaction: parameters involved are *FF1*, *GSC*.
4. Interaction between flight parameters and flight status: parameters involved are *N11*, *VB11*, *TLA1*.

The OpenFE framework [15] can help to identify useful cross features with a certain degree of interpretability. From the feature generation section in Figure 4, it can be seen that the core idea of OpenFE is to generate possible candidate features first. Then, redundant candidate features are removed. Finally, only features that are effective for the model are retained. In the feature generation stage, the two different original features are combined and transformed into candidate features through basic computation operations, including addition, subtraction, multiplication, division, maximum and minimum, etc. Next, the generated candidate features are validated and selected by gradient boosting to computing their incremental contribution to the model, as follows:

$$\Delta = L(f) - L(f')$$

$$L(f') = \sum_{i=1}^n l(y_i, \hat{y}_i + f'(x_i[\gamma'])) \quad (8)$$

In the above formulas,  $\gamma'$  represents a new feature set. However, this is only a rough screening. Further consideration need to be given to the relationship between newly generated features and original features. At this point, computing incremental contribution

needs to be used on the entire feature set, including original features and new features. The final results are shown in Table 4.

**Table 4.** Results information for candidate features.

| Number | Candidate Features |
|--------|--------------------|
| 0      | (LONPC*LATPC)      |
| 1      | max(LONPC,LATPC)   |
| 2      | (LONPC+LATPC)      |
| 3      | residual(LATPC)    |
| 4      | log(ALT_STD)       |
| 5      | freq(RUDD)         |
| 6      | max(TAT,ALT_STD)   |
| 7      | log(GSC)           |
| 8      | (FF1*GSC)          |
| 9      | freq(FF1)          |
| 10     | (N11*TLA1)         |
| ...    | ...                |
| 25     | (N11/VB11)         |
| ...    | ...                |

### 5.2. Automated Parameters Optimization

As we know, the XGBoost model has more than 10 parameters, including general parameters, tree parameters, learning task parameters, etc. The optimization of the parameters is a complex problem that requires the use of AutoML technology to improve the efficiency of model training and optimization.

In our method, Optuna (<https://optuna.org/>, (accessed on 10 November 2024)) [16] is used as an instrument to optimize model parameters, which is an advanced automatic hyperparameter tuning framework. Its core idea is to efficiently explore the hyperparameter space through an intelligent sampling algorithm and the accumulation of historical information, so as to find the best hyperparameter configuration.

Specifically, Optuna models the optimization problem as an objective function, which is the metric we need to optimize. In this paper, we set the objective function as the weighted F1 score that assigns different weights based on the proportion of each category of the model. Optuna defines a search space that contains hyperparameters that can be tuned and their range of values, which constitute the decision variables of the optimization problem. Optuna uses the idea of Bayesian optimization to guide the subsequent sampling direction by building a probabilistic model of the objective function to find the optimal solution faster. Common sampling strategies include Tree-Structured Parzen Estimator (TPE) [17], Cost-Function-Based Policy Optimization (CFPO) [18], etc. In this paper, the Categorical sampling strategy is used in the TPE method. The Categorical sampling strategy allows for setting a selectable list of parameters, and Optuna will sample from these options based on the specified sampling method. Optuna also provides support for parallelization, with multiple hyperparameter combinations enabling simultaneous evaluation and enhancing optimization efficiency. Moreover, Optuna is capable of acquiring intermediate results during the training process, including loss values for each epoch, and making adjustments to the subsequent search strategy accordingly. In general, Optuna achieves automatic and efficient optimization of hyperparameters through the construction of an objective function, definition of the search space, adoption of an efficient sampling strategy, and utilization of intermediate results for timely adjustments. The results of automatic parameter tuning are shown in Table 5.

**Table 5.** Results of automated adjustment of parameters.

| Parameter        | Default | Optimum | Description                           | Function                           |
|------------------|---------|---------|---------------------------------------|------------------------------------|
| learning_rate    | 0.1     | 0.02    | Learning rate                         | Help converge stably               |
| max_depth        | None    | 13      | Maximum depth of tree                 | Prevent over fitting               |
| min_child_weight | 1       | 4       | The smallest sample weight sum        | Control the stop of splitting      |
| subsample        | 1       | 0.8     | Subsampling rate                      | Balance the variance and bias      |
| colsample_bytree | 1       | 0.7     | Sampling columns while spanning trees | Improve the generalization ability |
| alpha            | 0       | 0.080   | L1 regularization coefficient         | Make more conservative             |
| lambda           | 0       | 0.086   | L2 regularization coefficient         | Make more conservative             |

## 6. Experimental Evaluations

The data for the experiments were extracted from the approach phase of 487 real-world flights of the same aircraft type. The number of unbalanced training sets and those that were balanced by the Borderline-SMOTE algorithm are shown in Table 6. Both the training and testing data were preprocessed and used for model training and validation. The same testing set is used to compare the experimental results.

**Table 6.** Information of datasets.

| Dataset                | Unbalanced | Balanced |
|------------------------|------------|----------|
| Training sample points | 48,858     | 50,000   |
| Testing sample points  | 10,251     | 10,251   |

### 6.1. Evaluation Metrics

This article uses multiple evaluation metrics to represent the performance of the model, including accuracy, precision, recall, and F1 score. The calculation formulas are as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (9)$$

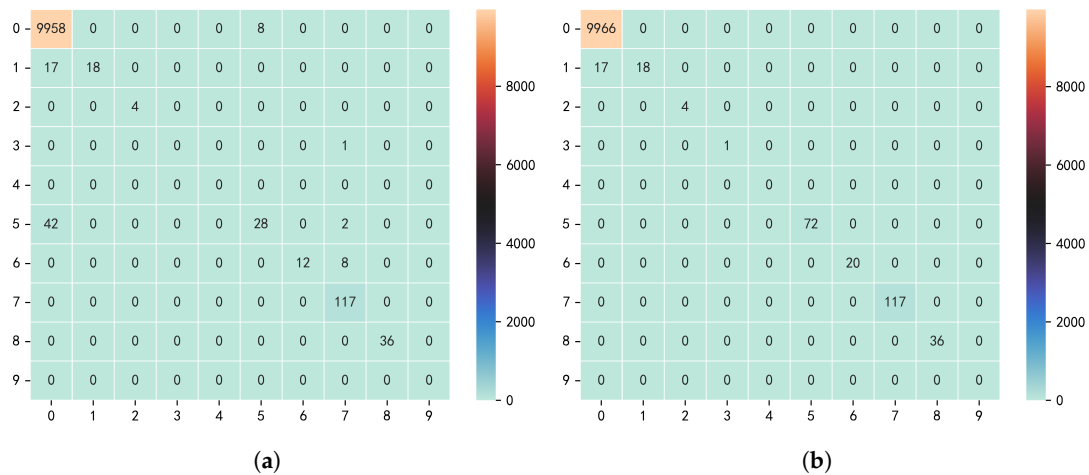
$$Precision = \frac{TP}{TP + FP} \quad (10)$$

$$Recall = \frac{TP}{TP + FN} \quad (11)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (12)$$

In the above formulas, *TP* represents *true positive*, *TN* represents *true negative*, *FN* represents *false negative*, and *FP* represents *false positive*. In the multi-class prediction results in this paper, *TP* represents the number of correctly classified predicted values, *FN* represents the number of other categories incorrectly predicted as a certain category, and *FP* represents the number of incorrectly predicted values as a certain category. Now, the data before and after balancing are used as training sets for experimental comparison, and the results are presented in the form of a heatmap, as shown in Figure 6.

The heatmap can be seen as a confusion matrix, where the sum of each row represents the number of true values for a category, and the sum of each column represents the number of predicted values for each category. Therefore, it can be inferred that the data on the diagonal of the confusion matrix is *TP*. According to Figure 6, the evaluation indicators can be calculated. Based on the binary classification, the average of the multiple classifications is taken as the overall result, as shown in Table 7.



**Figure 6.** Confusion matrix of the model: (a) unbalanced; (b) balanced.

**Table 7.** Model evaluation on test dataset.

| Dataset   | Precision | Recall | F1    | Accuracy |
|-----------|-----------|--------|-------|----------|
| Unbalance | 0.955     | 0.687  | 0.799 | 0.992    |
| Balanced  | 0.844     | 0.781  | 0.812 | 0.993    |

From the above results, it can be seen that the model can predict different abnormal events in the approach phase well. Compared with the prediction model using imbalanced samples as the training set, the balanced dataset greatly improved in recall rate. The overall improvement in accuracy and recall rate can be seen from the F1 score, which increased by 1.3%, and the accuracy also improved. To display as many types of anomalies as possible in the test set data, multiple routes were selected, with a total of 10,251 sample data. But the probability of all the above anomalies occurring simultaneously on a single route is extremely small. We selected the approach phase data of a certain flight as the validation set, and the results are shown in Table 8. The experimental results indicate that using balanced data as the training set is effective and stable in improving model performance.

**Table 8.** Model evaluation on validation dataset.

| Dataset   | Precision | Recall | F1    | Accuracy |
|-----------|-----------|--------|-------|----------|
| Unbalance | 0.965     | 0.663  | 0.786 | 0.914    |
| Balanced  | 0.991     | 0.812  | 0.893 | 0.990    |

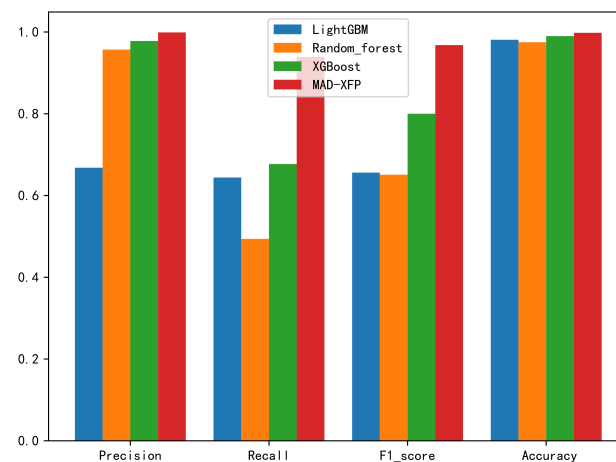
## 6.2. Results and Analysis

### 6.2.1. Overall Performance Evaluation

For multidimensional data classification and prediction problems, in addition to the XGBoost model, the Random Forest [19] and LightBGM [20] models are also tree-based ensemble learning methods widely used in data mining and machine learning tasks for prediction and classification problems. This section conducts an overall performance evaluation of the above methods, using different models to classify abnormal events for the same data.

The performance comparison is shown in Figure 7. It can be seen that all three models have an accuracy of over 90%, but their main contribution to accuracy comes from the fact that the model can almost recognize all normal data, which accounts for a very high

proportion of the overall data. In terms of precision and recall, the XGBoost model performs significantly better. At the same time, MAD-XFP outperforms the other three models.



**Figure 7.** Overall performance evaluation.

### 6.2.2. Ablation Study

This section outlines ablation experiments we conducted on the MAD-XFP model to investigate the impact of different components in the model on overall performance.

First, we disabled the feature engineering provided by OpenFE and parameter selection via Optuna to test the impact of temporal features and model hyperparameters on model performance. Then, the entire feature extraction and parameter optimization parts were completely removed, and the entire model was reduced to the most basic XGBoost model, in order to verify the necessity of the feature extraction part of the data in the entire model. The experimental results are shown in Table 9.

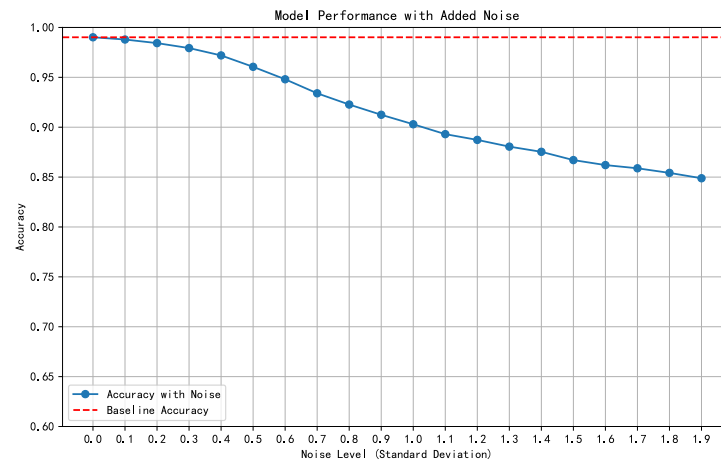
**Table 9.** Result of ablation study.

| Model                 | Metrics   |        |       |          |
|-----------------------|-----------|--------|-------|----------|
|                       | Precision | Recall | F1    | Accuracy |
| MAD-XFP               | 0.999     | 0.939  | 0.968 | 0.998    |
| w/o OpenFE            | 0.963     | 0.763  | 0.851 | 0.994    |
| w/o Optuna            | 0.868     | 0.939  | 0.902 | 0.997    |
| w/o OpenFE and Optuna | 0.844     | 0.781  | 0.812 | 0.993    |

The results indicate that removing different components from the model has a negative impact on the overall performance. Especially after disabling feature engineering via OpenFE, compared to the MAD-XFP model, the values of various evaluation indicators significantly decreased, with the F1 score decreasing by 11.7%. This indicates that data features have a significant impact on the prediction results.

### 6.2.3. Sensitivity Analysis

This section conducted sensitivity analysis experiments on the MAD-XFP model. To test the impact of data noise on model performance, Gaussian noise can be added to the eigenvalues to simulate common random noise behaviors and see how they affect the model's performance, in order to examine the robustness of the proposed model. The experimental results are shown in Figure 8.



**Figure 8.** Results of sensitivity analysis.

The impact of noise on the MAD-XFP model performance can be visually observed through Figure 8. The model can maintain stable performance even at high noise levels, with an accuracy of around 85%, proving its good robustness.

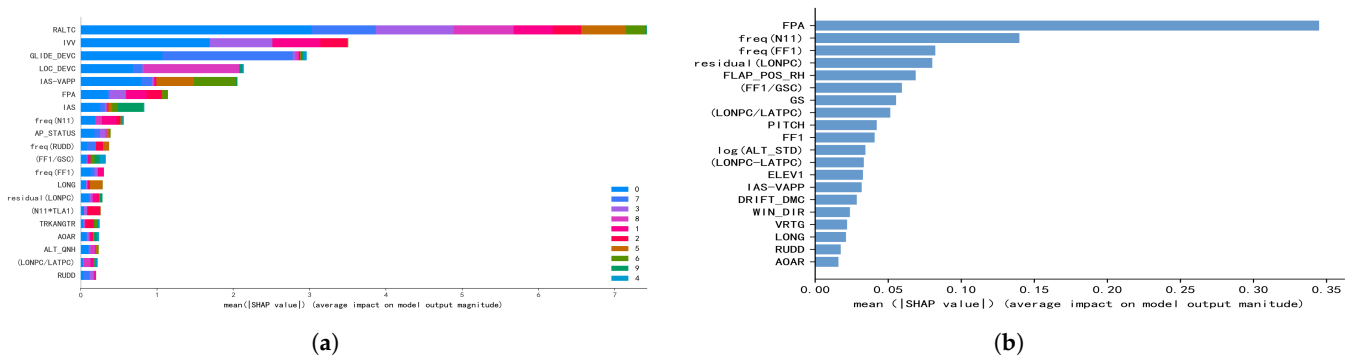
### 6.3. Automated Causal Analysis for Anomaly Event

The interpretability of machine learning refers to the degree to which humans can understand the relevant reasons for decision-making. The higher the interpretability of machine learning models, the easier it is for people to understand why certain decisions or predictions are made. In the interpretability research of machine learning models, Shapley Additive exPlans (SHAP (<https://shap.readthedocs.io/en/latest/>, (accessed on 13 October 2024))) can explain both global and local aspects and provides many excellent visualization methods that are increasingly favored by scholars [21]. The basic idea is to quantify the marginal contribution of each feature added to the model, take its mean, and record it as the contribution value of that feature to the model prediction.

Figure 9a is a global explanatory chart of the model presented in this paper. From the figure, it can be seen that the top parameters contributing to the output results of the model include radio altitude (*RALTC*), decline rate (*IVV*), ILS glide slope (*GLIDE\_DVC*), difference between air speed and approach speed target (*IAS – VAPP*), air speed (*IAS*), and so on. The result proves that all other parameters with high contribution are monitoring parameters for abnormal events, which is consistent with the facts and shows that the model is reliable.

In this paper, we equate the contribution of features to the local prediction of the model with the contribution of feature values to causing the abnormal event. Specifically, Figure 9b is a SHAP interpretation chart of the abnormal event with the *Large Decline Rate* (2000 ft~1000 ft). As expected, the top ranked features are related to the monitoring parameters *IVV* and the monitoring point radio altitude *RALTC* of the *Large-Dcline-Rate* (2000 ft~1000 ft) event. So, we removed these two features and their combinations from the SHAP interpretation chart, and the results are shown.

It can be seen that the features that contribute more to the abnormal event include flight path angle (*FPA*), low altitude speed ratio of engine 1 (*N11*), fuel mass flow rate of engine 1 (*FF1*), etc. These parameters belong to pilot operation parameters or flight state parameters, which are human factors. Based on this, we can preliminarily determine that the abnormal event was caused by human factors.

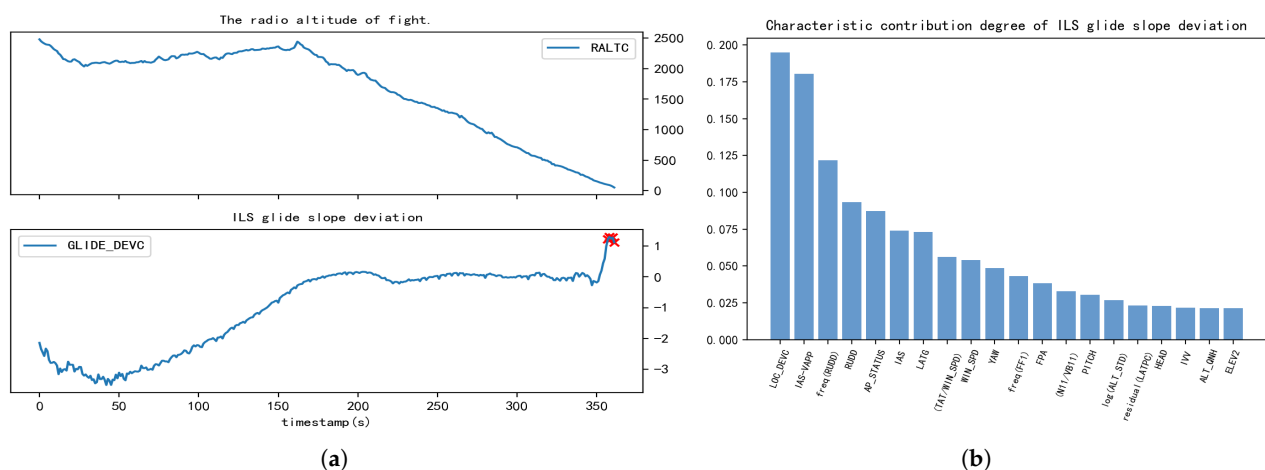


**Figure 9.** SHAP Interpretation Chart. (a) SHAP interpretation chart of MAD-XFP; (b) SHAP interpretation chart of anomaly label 1 (removed IVV, RALTC, and their combinations).

#### 6.4. Case Study

To verify whether the proposed model can be truly applied in reality, we conducted practical experiments on other flight data.

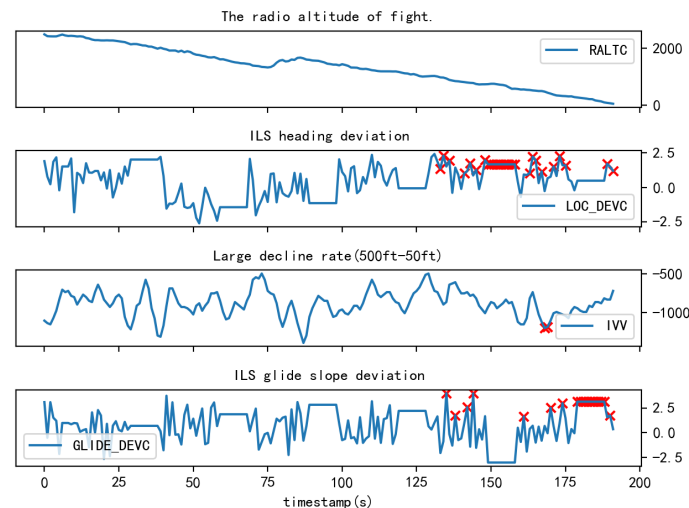
Figure 10 shows an example of anomaly detection and causal analysis. According to Figure 10a, it can be seen that there has been an *ILS glide slope deviation* on this flight route and the red markers in the figure indicate the outliers. Further analysis of the abnormal situation was performed and the results are shown in Figure 10b.



**Figure 10.** Example of anomaly detection and causal analysis. (a) high speed during approach phase; (b) causal analysis of high speed during approach.

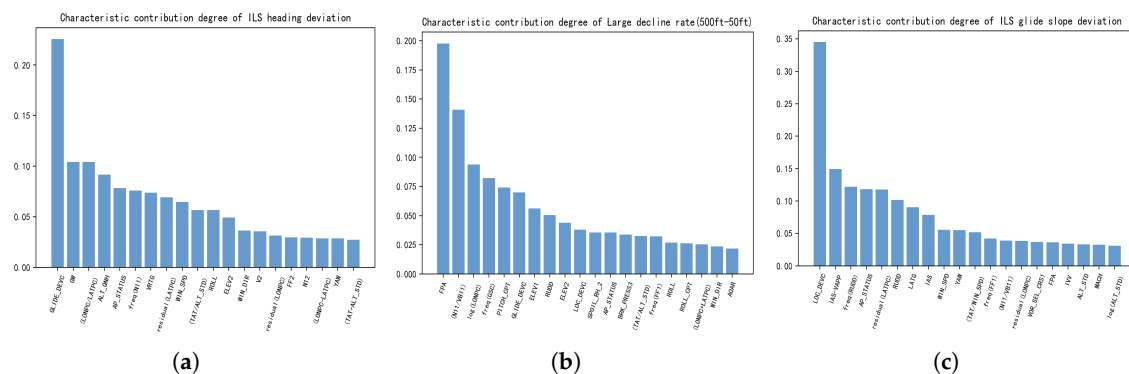
Specifically, it shows the feature contribution ranking of the abnormal event of *ILS glide slope deviation* during this flight. The parameters with the highest contribution ranking (excluding the radio altitude of the monitoring point and the monitoring parameters of each abnormal event) include ILS Heading (*LOC\_DEVC*), difference between air speed and approach speed target (*IAS – VAPP*), rudder position (*RUDD*), and so on. These parameters are all flight state parameters and monitoring parameters of FOQA in Table 2, formed by pilot operations. Therefore, we preliminarily conclude that the abnormal event of this flight was caused by human factors.

Figure 11 shows the detection results of multi-type anomalies. According to the figure, the red markers indicate outliers. The method in this paper detected multi-type abnormal events during the approach phase of this flight—namely, *ILS Heading Deviation*, *Large Decline Rate* (500 ft~50 ft), and *ILS Glide Slope Deviation*.



**Figure 11.** Multi-type anomaly detection.

Figure 12a–c are feature contribution ranking diagrams of the abnormal events that occurred during the flight of the aircraft shown in Figure 11. Similar to the above analysis method, parameters more related to the *ILS Heading Deviation* include ILS glide slope (*GLIDE\_DEVC*), gross weight of aircraft (*GW*), geographic location information (*LONPC/LATPC*), and speed of low-pressure turbine (*N11*). In addition to the parameters mentioned above, the parameters more related to *Large Decline Rate (500 ft~50 ft)* include flight path angle (*FPA*), interaction information between flight parameters and flight status (*N11/VB11*), ground speed (*GSC*), and so on. The parameters more related to the *ILS Glide Slope Deviation* include wind speed (*WIN\_SPD*). Based on the analysis of the above characteristics, the abnormality that occurred during this flight was the result of the combined effects of human factors and environmental factors.



**Figure 12.** Features ranking for detected anomaly events. (a) ILS Heading Deviation; (b) Large Decline Rate; (c) ILS Glide Slope Deviation.

## 7. Conclusions

In this paper, we mainly studied the problem of flight anomaly detection and causal analysis during the approach phase of civil aviation aircraft, and proposed an optimization method based on the XGBoost algorithm. A large amount of experimental data show that the method proposed in this paper can effectively detect abnormal events during the aircraft approach phase, making up for the shortcomings of traditional over-limit detection, which cannot simultaneously detect abnormal events coupled with multiple events. In addition, experts have only conducted scientific research on accident data, but our method can detect relatively small anomalies in daily flight processes. Through factor analysis, we

can clarify the source of anomalies and provide warnings for human factors involved in abnormal events. Frequent occurrences of the same abnormal event on the same model should be noted.

Considering that there are still a large number of unused features in QAR data, there is still room for improvement in constructing new composite features. In addition, there is a risk of overfitting when using the Borderline SMOTE algorithm for data balancing. Other balancing techniques should be considered.

Therefore, in future research, we will further explore abnormal events during other aircraft flight phases, such as the takeoff phase, landing phase, etc. We will further expand the feature usage of QAR data and explore the spatiotemporal information behind a large number of parameters in different flight stages, making the model more universal. In terms of data preprocessing, we will discuss the applicability of balanced weighted extreme learning machine for imbalance learning and a novel parallel-series data-driven model for IATA-coded flight delay prediction and features analysis, as well as QAR data balancing.

**Author Contributions:** Conceptualization, C.R.; Software, C.H.; Validation, X.D.; Formal analysis, X.D. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the Research Fund of the Tianjin Natural Science Foundation (No. 20JCYBJC00500).

**Institutional Review Board Statement:** This study does not require ethical approval and does not involve research on humans or animals.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The raw data supporting the conclusions of this article will be made available by the authors on request. The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Flight Standards Department of the Civil Aviation Administration of China. *Implementation and Management of Flight Operations Quality Assurance*; Civil Aviation Administration of China: Beijing, China, 2012.
2. Qing, G. The Era of Big Data—Multi-dimensional Use of QAR Data. In Proceedings of the WWW, Ljubljana, Slovenia, 19–23 April 2021.
3. Du, H.B.; Li, Z.X. Cause Analysis on Approach-and-Landing Loss Accidents and Their Countermeasures. *China Saf. Sci. J.* **2006**, *16*, 118–122.
4. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
5. Wang, X.; Lu, X. A Host-Based Anomaly Detection Framework Using XGBoost and LSTM for IoT Devices. *Wirel. Commun. Mob. Comput.* **2020**, *2020*, 8838571. [\[CrossRef\]](#)
6. Yang, M.; Liu, S.; Xu, J.; Tan, G.; Li, C.; Song, L. Achieving privacy-preserving cross-silo anomaly detection using federated XGBoost. *J. Frankl. Inst.* **2023**, *360*, 6194–6210. [\[CrossRef\]](#)
7. Hirata, T.; Kuremoto, T.; Obayashi, M.; Mabu, S.; Kobayashi, K. Time Series Prediction Using DBN and ARIMA. In Proceedings of the 2015 International Conference on Computer Application Technologies, Matsue, Japan, 31 August–2 September 2015; pp. 24–29.
8. Duraj, A.; Ludwicka, M. Detection of outliers in the financial time series using ARIMA models. In Proceedings of the 2018 Applications of Electromagnetics in Modern Techniques and Medicine (PTZE), Raclawice, Poland, 9–12 September 2018; pp. 49–52.
9. Hu, M.; Ji, Z.; Yan, K.; Guo, Y.; Feng, X.; Gong, J.; Zhao, X.; Dong, L. Detecting Anomalies in Time Series Data via a Meta-Feature Based Approach. *IEEE Access* **2018**, *6*, 27760–27776. [\[CrossRef\]](#)
10. Lee, K.; Lee, C.H.; Lee, J. Semi-Supervised Anomaly Detection Algorithm Using Probabilistic Labeling (SAD-PL). *IEEE Access* **2021**, *9*, 142972–142981. [\[CrossRef\]](#)

11. Akhmedova, S.; Stanovov, V.; Kamiya, Y. A Hybrid Clustering Approach Based on Fuzzy Logic and Evolutionary Computation for Anomaly Detection. *Algorithms* **2022**, *15*, 342. [[CrossRef](#)]
12. Sun, Q.; Ji, R. Flight anomaly detection model based on QAR data autoencoder and DBscan algorithm. In Proceedings of the 2021 IEEE 3rd International Conference on Civil Aviation Safety and Information Technology (ICCASIT), Changsha, China, 20–22 October 2021; pp. 309–312.
13. Qiu, R.; Yin, Y.; Su, Q.; Guan, T. An Anomaly Detection Algorithm of QAR Based on Spatial-Temporal Correlation. In Proceedings of the 2023 International Conference on Cyber-Physical Social Intelligence (ICCSI), Xi'an, China, 20–23 October 2023; pp. 7–12.
14. Han, H.; Wang, W.-Y.; Mao, B.-H. Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning. In Proceedings of the International Conference on Intelligent Computing, Hefei, China, 23–26 August 2005.
15. Zhang, T.; Zhang, Z.; Fan, Z.; Luo, H.; Liu, F.; Liu, Q.; Cao, W.; Li, J. OpenFE: Automated Feature Generation with Expert-level Performance. In Proceedings of the 40th International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023; Volume 202, pp. 41880–41901.
16. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna-A Next-generation Hyperparameter Optimization Framework. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; pp. 2623–2631.
17. Bergstra, J.; Bardenet, R.; Bengio, Y.; Kégl, B. Algorithms for hyper-parameter optimization. In Proceedings of the 25th International Conference on Neural Information Processing Systems, Granada, Spain, 12–15 December 2011; pp. 2546–2554.
18. Masahiro Nomura, M.S. CFPO: A Constrained Formulation of Pareto Optimization for Hyperparameter Tuning. *Mach. Learn. J.* **2021**.
19. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
20. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.Y. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 3149–3157.
21. Scott Lundberg, S.I.L. A Unified Approach to Interpreting Model Predictions. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 4768–4777.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.