BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 72(6), 2024, Article number: e151677
DOI: 10.24425/bpasts.2024.151677

ARTIFICIAL AND COMPUTATIONAL INTELLIGENCE

Flight quality assessment in full flight phase based on KOA-CNN-GRU-self-attention

Tianyi WU[1], Zichun LIN[2], Jianan HUANG[1], Yuxi DING[1], Ansaierding MAIHEMUTI[1], and Xiaowei XU[1]*

[1] School of Automobile and Traffic Engineering, Wuhan University of Science and Technology, Wuhan 430065, China
[2] Faculty of Materials, Wuhan University of Science and Technology, Wuhan 430081, China

Abstract. The main causes of aviation accidents in recent years are mostly related to pilot operational errors and pilot operational characteristics directly reflect flight quality. Hence, flight quality and flight safety are inseparable. Improving the assessment method of flight quality is of great significance for building a competency-based and evidence-based flight training system as well as enhancing flight safety. However, the problem is that some of the existing research is one-sided, and the assessment accuracy is not high. We propose a flight quality assessment method based on KOA-CNN-GRU-self-attention for the whole flight phase to accurately assess the flight quality and to improve and supplement the existing system. Firstly, the QAR data of the whole flight phase is selected and divided into three data sets according to the three indexes of operational smoothness, accuracy, and promptness, which are respectively substituted into the PCA comprehensive evaluation model to assess flight quality. Then, the evaluation results are labelled with the rating as the input of CNN-GRU-self-attention, and the parameters are optimized using KOA. Finally, the evaluation of flight quality for the three indexes was achieved by training the KOA-CNN-GRU-self-attention model. The test results show that the accuracy of operational smoothness, accuracy, and promptness reaches 98.73%, 95.07%, and 97.18%, respectively, and the assessment outcome is better and higher than the existing model. The model is also compared and analyzed with three base models CNN, QDA, XGBoost, and three fusion models CNN-self-attention, GRU-self-attention, CNN-GRU-self-attention, which show overall better results in accuracy, recall, precision, and F1-Score.

Keywords: KOA (Kepler optimization algorithm); CNN; GRU; self-attention; flight quality.

1. INTRODUCTION

With the swift growth of the civil aviation sector, civil aviation flight safety has emerged as a key concern for society, vital to the business and revenue of airlines, and directly relating to the safety of individuals. Indeed, flight safety stands as the bedrock of progress of the civil aviation industry. Over the recent years, enhancements have been implemented across various facets of civil aviation safety. In tandem with these advancements, civil aviation safety is witnessing a shift towards digitization and artificial intelligence, propelled by the modernization of aircraft equipment, ongoing innovation in production technologies, and the exponential growth of AI capabilities [1]. Although the global aviation industry has been tirelessly committed to improving flight safety, the potential for risks remains, with the capacity to be profoundly disruptive [2]. On March 21, 2022, a Boeing 737–800, with the registration number B-1791, operated by China Eastern Airlines Yunnan Limited Company, was flying from Kunming to Guangzhou as flight MU5735. The aircraft was involved in a catastrophic accident in the airspace controlled by Guangzhou, leading to the tragic loss of all 123 passengers and nine flight crew members [3]. The catastrophe led to a combined tally of direct and indirect economic damage estimated at around 6 billion yuan [4]. Statistics indicate that human operational issues have been a primary cause of aviation accidents. There is an urgent need to bolster the safety of civil aviation by training pilots who are highly skilled, capable, and exhibit consistent operational stability. Throughout the flight, it is essential for pilots to maintain stable flight conditions and to be equipped with the necessary skills to effectively counter aviation threats, thus ensuring the safety of the flight [5].

Human-induced operational errors have a direct bearing on the flight parameters of an aircraft, which in turn can compromise the quality of the flight. Consequently, the focus of flight safety research can shift from human operational errors, which are in the subjective realm, to studying aircraft flight parameters, which is a more objective domain, that is, the study of flight quality. QAR (quick access recorder) is a flight recorder installed on civil aviation aircraft, and its recorded data covers most of the important data on flight quality and engine performance status [6]. The Department of Transportation of the Civil Aviation Administration of China (CAAC) has issued the "Guidance Opinions on Comprehensively Deepening the Reform of Flight Training of Transport Airlines" in recent years, which requires relevant personnel to make good use of QAR data, and encourages airlines to introduce data sources such as analysis of aircraft flight parameters and training data, to analyze flight safety issues and build a competency-based and evidence-based

*e-mail: xuxiaowei@wust.edu.cn

Manuscript submitted 2024-02-13, revised 2024-08-15, initially accepted for publication 2024-08-17, published in November 2024.

flight training system [7]. This study did the following four main tasks:

1. The QAR data of the whole flight phases, such as the take-off phase, initial ascent phase, climb phase, cruise phase, descent phase, initial approach phase, final approach phase, and land phase, were selected for the experiment, which avoided the one-sidedness of the previous studies.
2. Based on the existing documents and classification methods, the QAR (quick access recorder) data is meticulously segmented into three distinct datasets focusing on operational smoothness, operational precision, and operational promptness. This stratification facilitates a more holistic and thorough evaluation of flight quality.
3. The comprehensive evaluation of QAR data using PCA overcomes the influence of human subjective factors, and the evaluation is accurate and effective, making the experimental analysis and results objective. The PCA algorithm is only affected by the data set itself, not affected by other factors, and it can eliminate the factors that affect each other among the original data components so that the model can be generalized.
4. We established a holistic flight quality evaluation model grounded in KOA-CNN-GRU-self-attention, capable of categorizing and assessing the flight quality of flights with remarkable accuracy and efficiency. This model not only elevates the efficacy and precision of the assessment process but also provides a valuable resource and reference for addressing flight quality evaluation challenges.

The layout of this paper unfolds as follows: Section 2 introduces related work. Section 3 describes the proposed method in detail. Section 4 conducts the experimental evaluation and comparative analyses. Section 5 concludes this paper and provides future work.

2. RELATED WORK

Within the realm of civil aviation, human-induced errors, encompassing those stemming from human operational mishaps, have remained a predominant factor in the occurrence of aviation accidents. Human-induced errors are difficult to analyze directly. The post-competency model of civil aviation flight cadets proposed by Lin *et al*. [8] uses a questionnaire to analyze the four dimensions of pilots' core qualities, ethical discipline, psychological quality, and leadership. Although this method can evaluate pilot competency to a certain extent, it is too subjective and is most likely to be influenced by the pilot's personal emotions, environment, and other factors at that time, and lacks reliability and persuasive power.

Utilizing QAR data for the analysis of flight quality adeptly shifts the investigative focus from the subjective aspects, human operational errors, to the more objective aspects, flight parameters, namely the study of flight quality. Over the past few years, a substantial number of researchers have delved deeply into the evaluation of flight quality, leveraging QAR data. The emergence of artificial intelligence has transformed the once daunting task of swiftly pinpointing operations that surpass safety thresholds from a complex array of aviation safety data into a tangible possibility. Weiliang Yuan *et al*. [9] proposed a flight quality assessment method based on PCA-PSO-SVM (principal component analysis-particle swarm optimization-support vector machine), which analyses the take-off climb phase and approach landing phase, the two most dangerous phases in the whole flight, but the accuracy is 90%, which needs to be improved. Benchi Wang *et al.* [10] combined the hierarchical analysis method with TOPSIS (a technique for order preference by similarity to an ideal solution) to determine the integrated weights to evaluate the flight quality of the takeoff phase, and Zhong *et al.* [11] proposed a method to analyze the flight quality of the actuator in the case of no failure. Xing *et al.* [12] proposed a time series method to analyze the flight quality in the landing phase, but the shortcomings are that only the landing phase is studied, and the assessment points of the landing phase of flight training only include the attitude and speed control of the aircraft, so the research is relatively one-sided.

For related evaluation issues, Rizvi [13]] summarized some commonly used methods, including XGBoost, RF, SVM, CNN, and so on. XGBoost and RF rely on data features and have lower accuracy, which makes them prone to overfitting when the data is noisy; the accuracy of SVM [8] in evaluating flight quality is only 90%. In contrast, CNN can effectively extract features from data using its convolutional structure, retaining useful information while reducing parameters and accelerating convergence, thereby improving classification accuracy, and offering a distinct advantage [14]. Traditional CNNs often require a large amount of data for effective training, and their max pooling layers reduce network dimensionality by selecting the maximum value within a given kernel. However, this approach only retains the most active neurons, which may lead to the loss of valuable inter-layer information and spatial details, and there are issues with gradient descent or explosions [15]. Adding attention mechanisms [16] and model fusion [17] are both popular and effective methods for improving convolutional neural networks.

Optimization algorithms work by adjusting the parameters of the model during model training to minimize or maximize a certain loss function. In recent years, optimization algorithms such as Harris hawk optimization (HHO), grey wolf optimization (GWO), and particle swarm optimization (PSO) have become quite common and popular, known for their excellent performance and widespread citation [18]. In 2023, the Kepler optimization algorithm (KOA), proposed by Abdel-Basset *et al.* was comprehensively validated for its faster convergence rate and its ability to approach the optimal solution more readily for the vast majority of optimization problems compared to other optimization algorithms [19].

After considering a number of factors and ensuring the generalizability of the model, in this paper, the PCA comprehensive evaluation method is used to analyze the data in terms of weights and explained variance ratio, etc. to derive a composite score that can reflect the degree of data deviation, and thus assess the quality of the flight. The main advantages of the PCA algorithm include the need to measure the amount of information only in terms of variance, which is not affected by factors outside the data set; and the orthogonality of principal components, which removes the factors that influence each other among the components of the original data [20]. These advantages allow us to analyze the different flight stages without considering the specific meaning of the data, but only the data itself, which facilitates the analysis of the overall data without the necessity to evaluate them separately in stages, saving analysis costs and time.

Based on the above, to accurately assess the flight quality of civil aviation airliners during flight more effectively and easily, to provide a reliable method for analyzing flight safety, and to improve flight quality, we constructed a KOA-CNN-GRU-self-attention (Kepler optimization algorithm-convolutional neural networks-gate recurrent unit-self-attention) evaluation model based on the QAR data to evaluate the flight quality with high precision. In this method, the QAR data of the whole flight phases, such as the take-off phase, initial ascent phase, climb phase, cruise phase, descent phase, initial approach phase, final approach phase, and land phase, were selected to be analyzed. The data are divided into three datasets of operational smoothness, accuracy, and promptness indicators according to the existing documents and division methods. A comprehensive evaluation of the three datasets is conducted respectively through the PCA comprehensive evaluation method and the three sets of data are divided into four categories and labelled according to the normal distribution law separately. The labelled data are fed into the CNN-GRU-self-attention classification model for training, and the loss function is calculated using KOA to optimize the model hyperparameters. Finally, the data are brought into the KOA-CNN-GRU-self-attention model to evaluate the flight quality.

3. METHODS

3.1. KOA optimizes the CNN-GRU-self-attention process

The KOA-CNN-GRU-self-attention classification model proposed in this paper mainly consists of the Kepler optimization algorithm (KOA) and the CNN-GRU-self-attention classification model. The KOA has the advantages of strong optimization and faster convergence [19]. The CNN-GRU-self-attention classification model can better extract data features and thus improve classification accuracy. KOA is used for hyper-parameter optimization to find the optimal combination of convolution kernel size, number of gated recurrent unit hidden states, and learning rate. The rationale for our choice of hyperparameters is as follows:

1. Convolution kernel size [21]: Larger convolutional kernel can bring a better receptive field, that is, to retain more data features, but the complexity will increase significantly, smaller convolutional kernel is less complex, with faster computation, but too small convolutional kernel cannot fully represent the original obvious features of the data.
2. Number of gated recurrent unit hidden states (hidden units) [22]: A smaller number of hidden units can reduce the model complexity, accelerate the training speed, and decrease the risk of model overfitting. However, when dealing with more complex data, this might not be sufficient to accurately extract the internal information on the data, leading to model distortion. Conversely, an excessive number of hidden units may lead to model overfitting, impairing the ability to make reasonable predictions and reducing the generalization capability of the model.
3. Learning rate [23]: It determines the step size of each training iteration. When the learning rate is set too small, the rate of descent is too slow, and it may take a long time to find the minimum value of the loss function. When the learning rate is too large, although the convergence is very fast, it will ignore the minimum value of the loss function, resulting in oscillating back and forth and being unable to converge.

Concurrently, focusing on the parameters of the Kepler optimization algorithm (KOA), we have selected a maximum of 20 evaluations as the benchmark for assessment. Additionally, we have set the number of populations for each dataset at 100 or more, ensuring the optimization requirements are met while reducing computational load and enhancing computational efficiency. As for the convolution kernel size, number of gated recurrent unit hidden states, and learning rate, we have established them within a reasonable range based on the explanations provided earlier. More specific initial parameter settings are illustrated in Table 3 and Table 4.

The overall process of model building is shown in Fig. 1 and is implemented as follows:

Step 1: Data processing

Extract the QAR data of the whole flight stage as the original data, divide the data into three data sets according to the three indices of operational smoothness, operational accuracy, and operational promptness, and finally clean the data separately.

Step 2: Comprehensive evaluation design

The cleaned data are firstly subjected to principal component analysis, and after obtaining the results of principal component analysis, the comprehensive evaluation of principal components is carried out to obtain the comprehensive score of each flight, and finally, the data are classified into four levels according to the σ law of normal distribution and labelled.

Step 3: Optimization of model parameters

Divide the processed data into a test set and training set, bring the training set into the CNN-GRU-self-attention model training, output the classification results, and carry out the loss function calculation to further train the model hyper-parameters until reaching the specified number of training times. After the optimization is completed, the optimization result is obtained and the model is optimized, i.e. the KOA-CNN-GRU-self-attention model is constructed.

Step 4: Derive results

Bring the test set into the KOA-CNN-GRU-self-attention model, and obtain the KOA adaptation curves, accuracy iteration curves, error iteration curves, confusion matrix, and accuracy prediction results for the three data sets of operational smoothness, operational accuracy, and operational promptness.

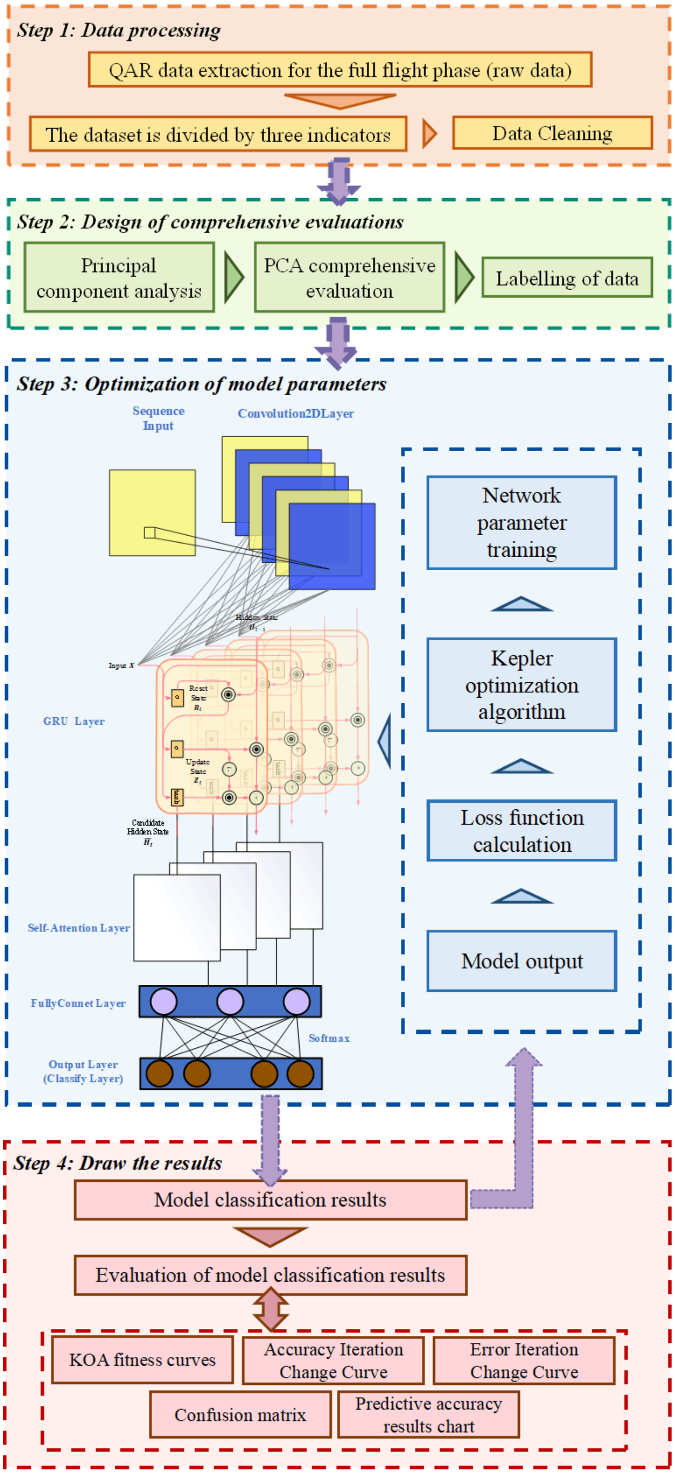
Step 1: Data processing
QAR data extraction for the full flight phase (raw data)
The dataset is divided by three indicators
Data Cleaning
Step 2: Design of comprehensive evaluations
Principal component analysis
PCA comprehensive evaluation
Labelling of data
Step 3: Optimization of model parameters
Sequence Input
Convolution2DLayer
Input X
Hidden State
Reset State
Update State
Candidate Hidden State
GRU Layer
Self-Attention Layer
FullyConnet Layer
Softmax
Output Layer (Classify Layer)
Network parameter training
Kepler optimization algorithm
Loss function calculation
Model output
Step 4: Draw the results
Model classification results
Evaluation of model classification results
KOA fitness curves
Accuracy Iteration Change Curve
Error Iteration Change Curve
Confusion matrix
Predictive accuracy results chart

Fig. 1. Training process of KOA-CNN-GRU-self-attention model

3.2. Data segmentation

A complete flight process is shown in Fig. 2, where the phases that account for a greater percentage of accidents are Land, Final approach, and Take-off, which account for 29.3%, 19.8%, and 12.8%, respectively. Therefore, the analysis of these stages is essential. However, in the cruise phase, which has the lowest accident rate of all phases of off-ground flight, the probability of an accident is still as high as 5.8%. As aviation safety is related to people's health and safety, the 5.8% accident rate should not be ignored. Therefore, this paper selects the QAR data of the whole flight phase to be analyzed based on the perspective of aviation safety [24].

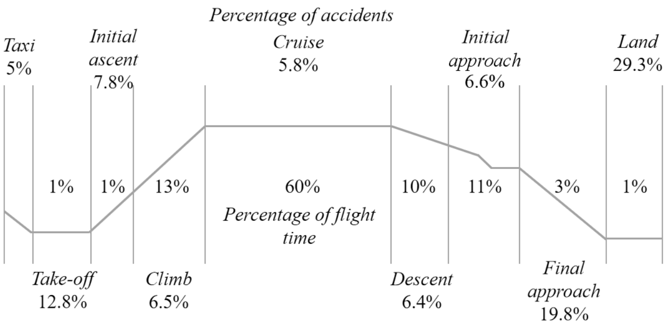

Fig. 2. Chart of the percentage of accidents and flight time by flight phase

The quality of the pilot's operational characteristics directly reflects the flight quality. In determining the indexes of operating characteristics, the extracted QAR data are used as the basis, and according to the CAA document "Flight Quality Monitoring (FOQA) Implementation and Management" and other information, and regarding the flight quality indexes proposed by Sun Ruisan *et al*. [25], this paper takes the operational smoothness, accuracy, and promptness as the pilot's operating characteristics. The interpretation is as follows:

1. Operational smoothness

Passenger comfort is the intuitive feedback of good flight quality, smoothness requires pilots to minimize overly violent operations, the number of controllable events occurring is small, and the operation should try to ensure that the aircraft is smooth; if the aircraft receives external disturbances (sudden winds, unstable airflow, etc.), it should be maintained as much as possible in the original flight state. For example, Max(g)G2toG3, i.e. the maximum g value during the climb from 1000 ft to the approach to 1500 ft.

2. Operational accuracy

To reduce the occurrence of accidents, pilots should strictly follow the operation manual when operating, and control the aircraft pitch, roll, and yaw attitude as well as flight parameters such as flight and taxiing speeds within certain limits at different flight stages. For example, TO_Trim, i.e. the pitch levelling position when lifting the front wheels.

3. Operational promptness

During the flight operation, the pilot is required to manoeuvre in time for each action, and the timing should not be too early or too late. For example, TimeToGearSelUp, i.e. time from take-off to retracting the landing gear (in seconds).

3.3. PCA comprehensive evaluation

Since the selected indicators do not have exactly the same scale, the QAR data are first de-scaled. In this paper, the data are normalized by z-score to achieve the purpose of de-quantification, and the formula is as follows:

$$z = \frac{x - \mu}{\sigma}, \tag{1}$$

where x is the raw data, μ is the mean of all data, and σ is the standard variance.

We used PCA (principal component analysis) for the comprehensive evaluation of QAR data. The principal component analysis method was first proposed by Karl Pearson in 1901. The analysis includes the special values of the covariance matrix [26]. Its main advantage is that it can effectively deal with high dimensional data, analyze the correlation between variables, and achieve data compression without losing too much information from the original data while ensuring the original trend of the data. The advantages of applying PCA in this paper are the following. It allows the experiment to consider only the data itself and ignore other factors, saving the cost and time of the experiment and making the method highly generalizable; through the analysis of the variance, the degree of data deviation is derived to evaluate its flight quality, which is a simple and objective method with strong practical significance. The principal component explained variance ratio graph is shown in Fig. 3.

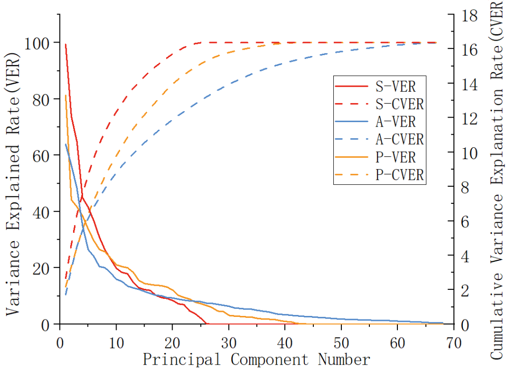

Fig. 3. Plot of explained variance by principal components

From Table 1 and Fig. 3, based on the calculation of the explained variance of the principal components and the cumulative explained variance, 12 principal components were selected for the smoothness indicator, 26 principal components for the accuracy indicator, and 18 principal components for the timeliness indicator. Thus, the principal component score matrix is obtained as

$$PC = (PC_1, PC_2, \ldots, PC_e) = Z\,(a_1, a_2, \ldots, a_e). \tag{2}$$

In turn, the integrated evaluation model, weighted by the explained variance ratio, was found to be

$$PC_z = \frac{\sum\limits_{d=c}^{e} PC_d \lambda_d}{\sum\limits_{d=1}^{e} \lambda_d}, \tag{3}$$

where PC_1, PC_2, PC_e denotes the 1st, 2nd, …, k-th principal components; a_1, a_2, a_e is the unit eigenvector of the matrix of correlation coefficients between variables; λ_c is the eigenvalue of the matrix of correlation coefficients between variables.

Table 1
Table of cumulative explained variance by each principal component

Principal compo-nent	S-EVR	S-CEVR	A-EVR	A-CEVR	P-EVR	P-CEVR
1	16.259	16.259	10.453	10.453	13.301	13.301
2	12.088	28.347	9.246	19.699	7.223	20.523
3	10.591	38.938	7.883	27.583	6.821	27.345
4	7.345	46.283	5.712	33.295	6.294	33.639
5	6.791	53.074	4.329	37.624	5.517	39.156
6	5.995	59.07	3.93	41.555	4.849	44.004
7	5.045	64.115	3.341	44.896	4.335	48.339
8	4.304	68.419	3.278	48.174	4.203	52.542
9	3.768	72.187	2.964	51.138	3.832	56.374
10	3.246	75.433	2.613	53.751	3.447	59.821
11	3.006	78.439	2.48	56.231	3.344	63.165
⋮	⋮	⋮	⋮	⋮	⋮	⋮
26	0.015	100	1.221	80.704	1.083	93.519

Note. EVR: explained variance ratio, CEVR: cumulative explained variance ratio, S: operational smoothness, A: operational accuracy, P: operational promptness. e.g. S-EVR: explained variance ratio for the data set of operational smoothness, S-CEVR: cumulative explained variance ratio for the data set of operational smoothness

Substituting the principal component score matrix into equation (3), the composite score of each flight can be obtained. The comprehensive score is sorted in the order from the smallest to the largest to get the comprehensive score of each flight, and the lower the score, the higher the flight quality of the flight. The results of the operational smoothness evaluation are shown in Table 2.

Table 2
Operational smoothness principal component composite evaluation score

Flight	Principal component score					Ran-king	Aggre-gate score
	1st	2nd	3rd	⋯	12th		
1	–1.2367	–0.9744	0.3728	⋯	0.1603	494	–0.2707
2	–0.5292	0.0612	1.6399	⋯	–0.6719	1331	0.0640
3	–0.5525	0.2367	1.1646	⋯	–0.5977	630	–0.2067
4	–1.0499	0.4196	–0.5754	⋯	–0.9421	387	–0.3309
5	–0.7373	–1.9436	0.2090	⋯	–0.6290	33	–0.7890
6	0.3332	–1.9296	0.6539	⋯	–0.8934	220	–0.4492
7	–0.6222	–0.3690	–0.3872	⋯	1.0888	524	–0.2533
8	–0.5752	0.0102	–1.1684	⋯	–0.8344	259	–0.4213
9	0.3402	0.4300	1.0689	⋯	–0.7789	1605	0.1616
⋮	⋮	⋮	⋮	⋱	⋮	⋮	⋮
2367	0.6120	–0.5614	0.9078	⋯	0.2522	2009	0.3359

To improve the evaluation efficiency, according to the comprehensive evaluation ranking of PCA, the raw data were classified into four categories: A, B, C, D, i.e. excellent, good, moderate, and poor, according to the normal distribution σ rule, which accounted for 15.865%, 34.135%, 34.135%, and 15.865% of the original data, respectively, and were marked with labels as inputs to the KOA-CNN-GRU-self-attention.

3.4. Kepler optimization algorithm (KOA) model

The Kepler optimization algorithm (KOA) is a new physics-based meta-heuristic algorithm said to be inspired by Kepler's laws of planetary motion for predicting the position and velocity of a planet at any given time. Candidate solutions (planets) behave differently from the Sun (optimal solution) at different times, allowing for a more efficient exploration and utilization of the search space. KOA has faster speed and better results than the more widely used genetic algorithm (GA), particle swarm optimization (PSO), etc. [19]. Using the KOA, we can rationally optimize different parameters in the deep learning network. The steps are as follows:

1. Initialization process

The population is first initialized by placing each planet at a random position on the orbit and giving a random value to the orbital eccentricity e and the orbital period T.

2. Defining the gravitational force (F)

Gravity is then defined according to the universal gravitation formula:

$$F_{g_i}(t) = e_i \times \mu(t) \times \frac{\bar{M}_S \times \bar{m}_i}{\bar{R}_i{}^2 + \varepsilon} + r_1, \tag{4}$$

where $\bar{M}_S$ and $\bar{m}_i$ denote the normalized values of M_S and m_i, which represent the masses of X_S and X_i, respectively, and ε is a small value; μ is the universal gravitational constant; e_i is the eccentricity of the planetary orbits, which is a value between 0 and 1 designed to give the KOA a stochastic character; r_1 is a randomly generated value between 0 and 1 to give more variability to the values of the gravitational forces during the optimization process; and $\bar{R}_i$ is the normalized value of R_i, which represents the Euclidean distance between X_S and X_i.

3. Calculating the object velocity

The speed of celestial bodies is affected by the gravitational pull of the sun, as a planet moves closer to the sun, its speed increases, and as it moves farther away, its speed decreases.

$$V_i = \begin{cases} \ell \times \left(2r_4\vec{X}_i - \vec{X}_b\right) + \ddot{\ell} \times \left(\vec{X}_a - \vec{X}_b\right) \\ \quad + (1 - R_{i\text{-}norm}(t)) \times \mathscr{F} \times \vec{U}_1 \times \vec{r}_5 \\ \quad \times \left(\vec{X}_{i,\text{up}} - \vec{X}_{i,\text{low}}\right), \qquad \text{if } R_{i\text{-}norm}(t) \le 0.5, \\ r_4 \times \mathscr{L} \times \left(\vec{X}_a - \vec{X}_i\right) \\ \quad + (1 - R_{i\text{-}norm}(t)) \times \mathscr{F} \times \vec{U}_2 \times \vec{r}_5 \\ \quad \times \left(r_3\vec{X}_{i,\text{up}} - \vec{X}_{i,\text{low}}\right), \qquad \text{else}, \end{cases} \tag{5}$$

$$\ell = \vec{U} \times \mathscr{M} \times \mathscr{L}, \tag{6}$$

$$\mathscr{L} = \left[\mu(t) \times (M_S + m_i)\left|\frac{2}{R_i(t)} - \frac{1}{a_i(t) + \varepsilon}\right|\right]^{\frac{1}{2}}, \tag{7}$$

$$\mathscr{M} = (r_3 \times (1 - r_4) + r_4), \tag{8}$$

$$\vec{U} = \begin{cases} 0 & \vec{r}_5 \le \vec{r}_6, \\ 1 & \text{else}, \end{cases} \tag{9}$$

$$\mathscr{F} = \begin{cases} 1 & \text{if } \vec{r}_4 \le 0.5, \\ -1 & \text{else}, \end{cases} \tag{10}$$

$$\ddot{\ell} = \left(1 - \vec{U} \times M \times \mathscr{L}\right), \tag{11}$$

$$\vec{M} = (r_3 \times (1 - r_5) + r_5), \tag{12}$$

$$\vec{U}_1 = \begin{cases} 0 & \vec{r}_5 \le \vec{r}_4, \\ 1 & \text{else}, \end{cases} \tag{13}$$

$$\vec{U}_2 = \begin{cases} 0 & \vec{r}_3 \le \vec{r}_4, \\ 1 & \text{else}, \end{cases} \tag{14}$$

$$a_i(t) = r_3 \times \left[T_i^2 \times \frac{\mu(t) \times (M_S + m_i)}{4\pi^2}\right]^{\frac{1}{3}}, \tag{15}$$

$$R_{i\text{-}norm}(t) = \frac{R_i(t) - \min(R(t))}{\min(R(t)) - \min(R(t))}, \tag{16}$$

where $\vec{V}_i(t)$ represents the velocity of the i-th planet; r_3 and r_4 are two numerical values chosen at random between 0 and 1 according to the uniform distribution; and $\vec{r}_5$ and $\vec{r}_6$ stands for two vectors of numerical values that are generated arbitrarily in the range (0, 1). $\vec{X}_a$ and $\vec{X}_b$ are two objects chosen arbitrarily from the individuals in the current population; $\mathscr{F}$ is a variable including randomly one value of [1, −1] to alter the search direction.

4. Escaping from the local optimum

In our solar system, most objects rotate counterclockwise around the sun, but some rotate around the sun in a clockwise direction. KOA simulates this behavior by using a flag F that changes the direction of the search, giving the agent a good chance of scanning the search space accurately, using this behavior to escape from the local optimum.

5. Updating the positions of objects

Objects rotate around the sun in their own elliptical orbits. During rotation, an object moves closer to the sun for a certain amount of time and then moves away from the sun. KOA models this behaviour through two main phases: the exploration and exploitation phases. KOA explores objects far from the Sun to find new solutions while using solutions closer to the Sun more accurately when searching for new locations near the best solutions.

Update the new position of each object away from the sun with the following equation:

$$\vec{X}_i(t+1) = \vec{X}_i(t) + \mathscr{F} \times \vec{V}_i(t) + \left(F_{g_i}(t) + |r|\right) \times \vec{U} \\ \times \left(\vec{X}_S(t) - \vec{X}_i(t)\right), \tag{17}$$

where $\vec{X}_i(t+1)$ is the new position of object i at time t+1; $\vec{V}_i(t)$ is the velocity of object i required to reach the new position;

$\vec{X}_S(t)$ is the best position of the sun found so far, and F is used as a flag to change the direction of the search.

6. Updating distance with the Sun

To further improve the exploration and exploitation of planets, the algorithm mimics the typical behavior of the distance between the sun and the planets; when the planets are close to the sun, the KOA will focus on optimizing the exploitation operator; when they are far from the sun, the KOA will optimize the exploration operator. The conversion of the rule will depend on the regulation parameter h

$$\vec{X}_i(t+1) = \vec{X}_i(t) \times \vec{U}_1 + \left(1 - \vec{U}_1\right) \times \left(\frac{\vec{X}_i(t) + \vec{X}_S + \vec{X}_a(t)}{3.0} + h \times \left(\frac{\vec{X}_i(t) + \vec{X}_S + \vec{X}_a(t)}{3.0} - \vec{X}_b(t) \right) \right), \tag{18}$$

where h is the adaptive factor used to control the distance between the Sun and the current planet at time t, defined as follows:

$$h = \frac{1}{e^{\eta r}}, \tag{19}$$

r is a randomly generated number based on a normal distribution, and η is a linear decreasing factor from 1 to -2

$$\eta = (a_2 - 1) \times r_4 + 1. \tag{20}$$

The cyclic control parameter a_2 gradually decreases from -1 to -2 throughout the $\bar{T}$ cycles of the optimization process and is defined as follows:

$$a_2 = -1 - 1 \times \left(\frac{t \% \frac{T_{\max}}{\bar{T}}}{\frac{T_{\max}}{\bar{T}}} \right). \tag{21}$$

7. Elitism

This step establishes an elite strategy to ensure that the Sun and planets are always in optimal localized positions, defined as

$$\vec{X}_{i,\text{new}}(t+1) = \begin{cases} \vec{X}_i(t+1), & \text{if } f\left(\vec{X}_i(t+1)\right) \le f\left(\vec{X}_i(t)\right), \\ \vec{X}_i(t) & \text{else.} \end{cases} \tag{22}$$

The general flow is shown in the following diagram.

In Fig. 4, the initialization of the object population is carried out first, including selecting random positions, orbital eccentricity, and orbital period. Then, evaluate fitness values for the initial population and determine the global best ($\vec{X}_S$) solution as the Sun. Next, calculate the Euclidian distance and gravitational force between the Sun and all the objects and calculate the velocity of all the objects. Generate two random numbers r_1, r between 0 and 1. If r is less than r_1, use equation (18) to update the distance between the object and the sun. If r is greater than r_1, use equation (5) and equation (17) to update the position of the object, and apply an elite strategy through equation (22) to ensure the optimal position of the planet and the sun. Finally, evaluate the fitness values of all objects and the sun and determine the global best ($\vec{X}_S$) solution as the Sun, thus obtaining the best solution. If the final solution is not acceptable, recalculate the Euclidean distance, gravity, and velocities of all objects between the sun and all objects and repeat the previous process until the best solution is achieved.

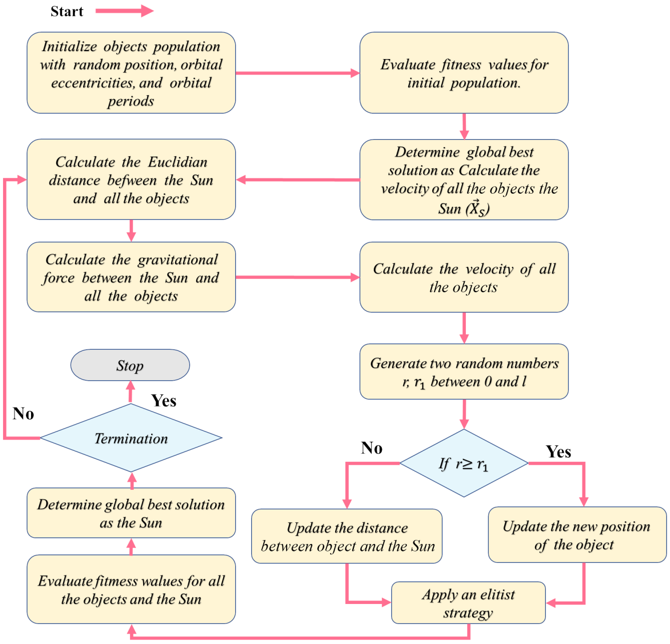
Start
Initialize objects population with random position, orbital eccentricities, and orbital periods
Evaluate fitness values for initial population.
Determine global best solution as Calculate the velocity of all the objects the Sun ($\vec{X}_S$)
Calculate the Euclidian distance befween the Sun and all the objects
Calculate the gravitational force between the Sun and all the objects
Calculate the velocity of all the objects
Generate two random numbers r, r_1 between 0 and l
Stop
Yes
No
Termination
No
If $r \geq r_1$
Yes
Determine global best solution as the Sun
Update the distance between object and the Sun
Update the new position of the object
Evaluate fitmess walues for all the objects and the Sun
Apply an elitist strategy

Fig. 4. KOA optimization flowchart

The initial parameter settings of KOA are shown in Table 3.

Table 3

KOA initial parameter table

Data set	Number of searching individuals (planets)	Maximum number of function evaluations	Variable lower bounds	Variable upper bounds
Accuracy	100	20	[0.001, 1, 50]	[0.01, 5, 100]
Promptness	300	20	[0.001, 1, 50]	[0.01, 5, 100]
Smoothness	200	20	[0.001, 1, 50]	[0.01, 5, 100]

where the lower bound of the variable and the upper bound of the variable contains the learning rate, the convolutional kernel size, and the hidden layer unit.

3.5. CNN-GRU-self-attention classification model

The CNN-GRU-self-attention model utilizes the convolutional layer of CNN for partial effective data feature extraction and parameter sharing, which effectively improves the efficiency and speed of feature extraction. Next, a GRU layer is combined with the CNN layer, which has high training efficiency. Finally, integrates the results with the self-attention mechanism. This facilitates better prediction of data. The process of realization is as follows. Firstly, the data are fed into the convolutional layer to extract features and activate them using the Relu function; next, the sequence layer is used as the input to the GRU layer for

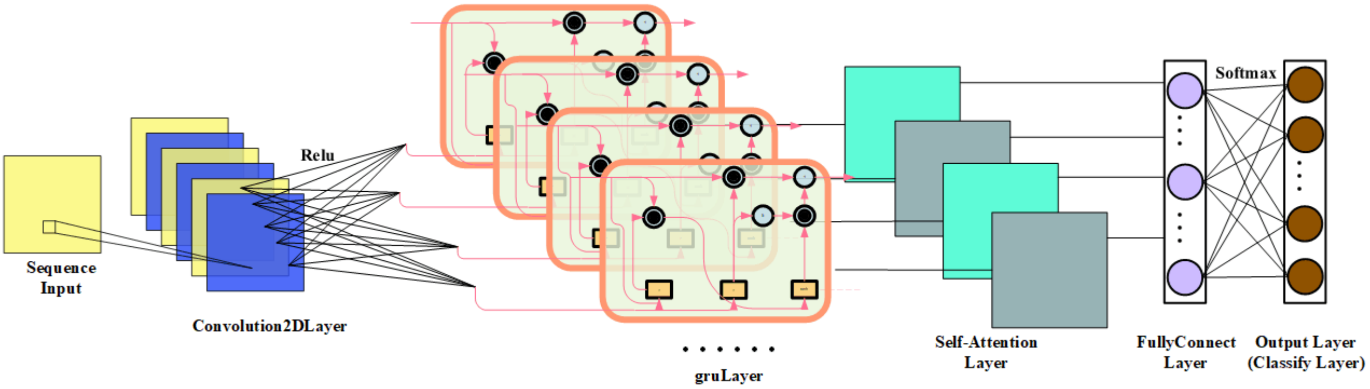

Fig. 5. CNN-GRU-self-attention model

unfolding and tiling; after that, the model is trained by accessing the discard layer with a discard rate of 0.1; finally, the fully connected layer is used for the integration of the results and the linear transformation. The model structure is shown in Fig. 5.

And in our model, we divide the three datasets of operational accuracy, promptness, and smoothness into three sets of experiments, and optimize the hyperparameters of the CNN-GRU-self-attention model with KOA, respectively, and the optimization results are shown in Table 4.

Table 4
CNN-GRU-Self-Attention parameter optimization results

Data set	Learning rate	Convolution kernel size	Hidden units
Accuracy	0.0010	[2, 1]	88
Promptness	0.0021	[2, 1]	84
Smoothness	0.0012	[1, 1]	77

3.5.1. CNN model

Since Lecun proposed convolutional neural networks in 1989 [27], they have been widely used in the fields of data prediction and image recognition [28]. They usually consist of an input layer, a convolutional layer, a pooling layer, a fully connected layer, an activation layer, and an output layer.

The pooling layer is usually used for data downscaling, preserving salient features. However, our data has been dimensionally reduced by PCA. So we omit the pooling layer (maxPooling2D), to retain the characteristics of the data at the same time, and to improve the operation speed. The core formula of CNN is shown as

$$Input_{n+1} = ReLU\left(\Sigma w_n * Input_n + b_n\right), \tag{23}$$

where $Input_{n+1}$ and $Input_n$ denote the input of the next layer and the current layer, respectively; w_n denotes the weight of the current layer; and b_n denotes the bias function for the current layer.

3.5.2. GRU model

GRU has been widely used since its introduction in 2014 [29]. Compared with LSTM and RNN, it can alleviate the problem of gradient disappearance of RNN and achieve the same effect as the LSTM model with longer training time [30]. The number of units of GRU is searched by the Keplerian optimization algorithm. For the data set of accuracy, promptness and smoothness are 88, 84, and 77. The calculation of the GRU unit can be shown as the formula

$$R_t = \sigma\left(X_t W_{xr} + H_{t-1} W_{hr} + b_r\right), \tag{24}$$

$$Z_t = \sigma\left(X_t W_{xz} + H_{t-1} W_{hz} + b_z\right), \tag{25}$$

$$\tilde{H}_t = \tanh\left(X_t W_{xh} + \left(R_t \odot H_{t-1}\right) W_{hh} + b_h\right), \tag{26}$$

$$H_t = Z_t \odot H_{t-1} + (1 - Z_t) \odot \tilde{H}_t, \tag{27}$$

where R_t denotes the reset gate; Z_t denotes the update gate; $\tilde{H}_t$ denotes the candidate's hidden state; H_t denotes the hidden state; $\odot$ denotes Hadamard product; W_{xr}, W_{hr}, W_{hz}, W_{xz}, W_{xh}, W_{hh} denote the weight of the corresponding state; b_r, b_z, b_h denote the bias function for the corresponding state; σ denotes the sigmoid active function.

3.5.3. Self-attention model

The self-attention mechanism proposed by the Google machine translation team in 2017 began to become a research hotspot for attention neural networks. Each unit of self-attention captures the information of the whole sentence, and the output can also be computed in parallel. Self-attention is a special case of general attention, where $Q = K = V$. Attention is computed for each unit in a sequence and for all units in that sequence.

An attention function can be described as mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are vectors. The output is computed as a weighted sum of the values, where the weight assigned to each value is computed by the compatibility function of the query with the corresponding keyword. The matrices Q, K, and V are the query, key value, and value, respectively.

The input consists of queries and keys of dimension d_k, and values of dimension d_v. We compute the dot products of the query with all keys and divide each by $\sqrt{d_k}$, and apply a softmax function to obtain the weights on the values [31].

The output matrix is calculated as

$$Attention(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \tag{28}$$

4. FLIGHT QUALITY CLASSIFICATION AND EVALUATION EXPERIMENT

4.1. Data cleaning

The data in this article comes from the QAR data provided by the Chinese Society of Optimization, Overall Planning, and Economic Mathematics for 2370 flights on the Boeing 737 series aircraft. In the process of data collection, transmission and storage, the raw data contains some outliers, poor integrity, high noise, and low consistency due to a variety of factors such as hardware equipment limitations, variable collection environment, transmission validity, and storage reliability, which creates a great obstacle to the subsequent analysis. Therefore, it is necessary to clean the data and improve the quality of the data [32].

1. Data streamlining. Remove objective data items in the data table that are not relevant to the study, such as “time”, “airport of departure” and “aircraft weight”.
2. Missing value processing. For the missing values of some fixed characteristics, this paper deletes the object; for the rest of the data, this paper carries out Hermite interpolation on the missing data items.

4.2. Model training

We used Matlab R2023b, SPSSPRO1.1.21, and Origin 2021 for data analysis and experiments. We divided the data into three datasets according to operational smoothness, operational accuracy, and operational promptness for three separate experiments, and the number of samples in each experimental dataset was 2367. 70% of the data were randomly selected as the training set and the remaining 30% as the test set according to the ratio of 7:3. Each principal component score was brought into the model as input and the composite score as output for training. The operational smoothness, operational accuracy, and operational promptness KOA fitness curves are as follows, respectively:

As can be seen from Fig. 6, the degree of convergence is different for different datasets and different numbers of planets (number of searching individuals). Therefore, we take 200, 300, and 100 as the number of searching individuals for the Kepler optimization algorithm for smoothness, accuracy, and promptness datasets, respectively, i.e. for smoothness, accuracy, and promptness datasets, the number of planets for KOA is 200, 300 and 100, respectively. At this time, the error is fundamentally converged.

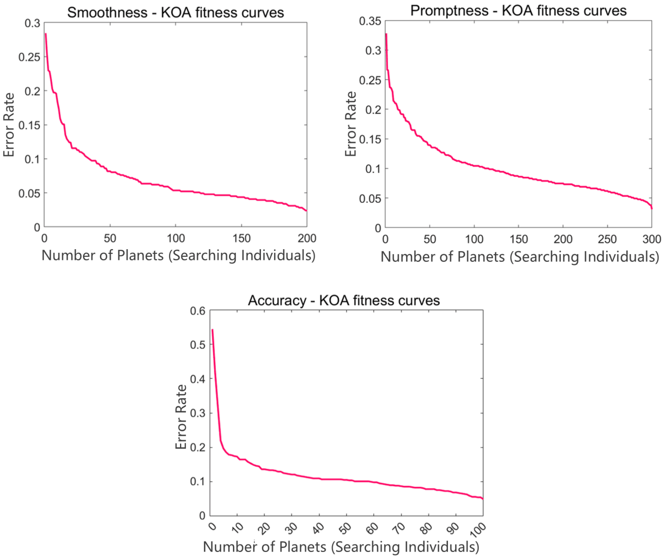

Fig. 6. KOA fitness curves

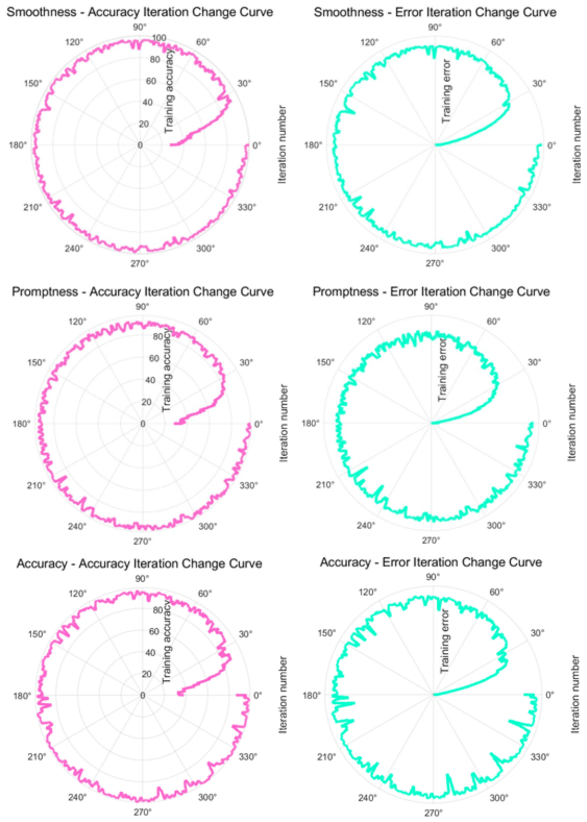

Fig. 7. Iteration change curve of accuracy & error

Figure 7 shows the iterative change curves of the accuracy of operation smoothness, accuracy, and promptness classification with the iterative change curves of the error. As can be seen from the figure, when the number of iterations is small, the accuracy of the model is low, the error is large, and the fluctuations are all large, and the model is still learning. With the growth of the number of iterations, the model accuracy shows an upward trend, but the fluctuations are large and unstable. Although the error decreases, the fluctuations are still large. The number of iterations approximates the set value (operation smoothness data for 200 times, operation accuracy data for 100 times, operation promptness data for 200 times). We can see that the accuracy of the model fluctuates in a higher range, the amplitude of the iteration is smaller than the beginning of a lot of times, and it basically tends to be stable. The error diminishes, and in addition to a few big fluctuations it tends to be stable, and the model has completed learning. To facilitate training and graphing, we transform the categories A, B, C, D into 1, 2, 3, 4 for training.

4.3. Model performance

After training, the prediction set data is forecast. To further validate the effectiveness of the model effectively, we evaluate the effect of the KOA-CNN-GRU-self-attention model by four evaluation indexes, namely, ACCURACY, RECALL, PRECISION & F1-SCORE [33], which are formulated as follows [34]:

$$precision = \frac{TP}{(TP+FP)}, \tag{29}$$

$$recall = \frac{TP}{(TP+FN)}, \tag{30}$$

$$accuracy = \frac{(TP+TN)}{(TP+FP+TN+FN)}, \tag{31}$$

$$F1\text{-}score = \frac{2P*R}{(P+R)}, \tag{32}$$

where

TP – True Positive, predicted positive, actual positive, successfully predicts positive samples as positive;

FP – False Positive, predicted as positive, actually negative, incorrectly predicted the negative sample as positive;

TN – True Negative, Predicted Negative, Actual Negative, Successfully Predicted Negative Sample as Negative;

FN – False Negative, predicting with negative and actually positive, incorrectly predicting positive samples as negative;

P – precision;

R – recall.

The confusion matrices for the three datasets were derived as shown in Figs. 8–10.

From Figs. 8–10, it can be seen that for the operational smoothness, accuracy, and promptness dataset, the accuracy of the model is above 95%, with operational smoothness of 98.7%, operational accuracy of 95.1%, and operational promptness of 97.2%, which is much better than the 90% of the existing model [4]. For the fourth category, the recall rates are all above 98%, and the precision is higher than or equal to 94.1%, which is better than the 94.1% of the existing model [4].

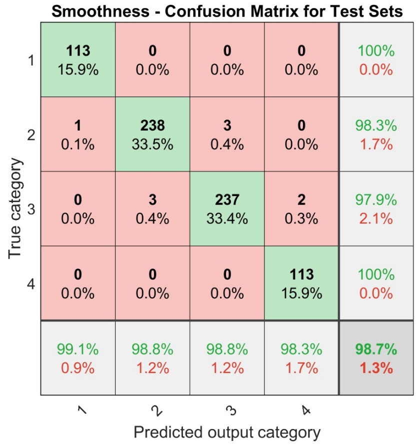

Fig. 8. Confusion matrix of operational smoothness

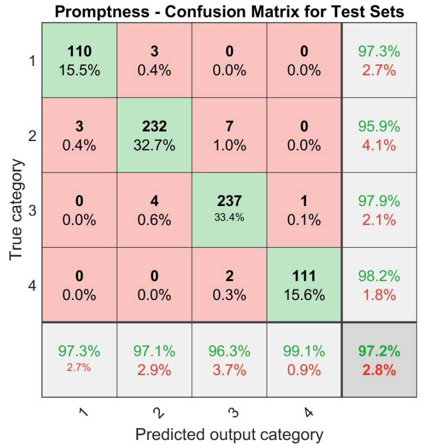

Fig. 9. Confusion matrix of operational promptness

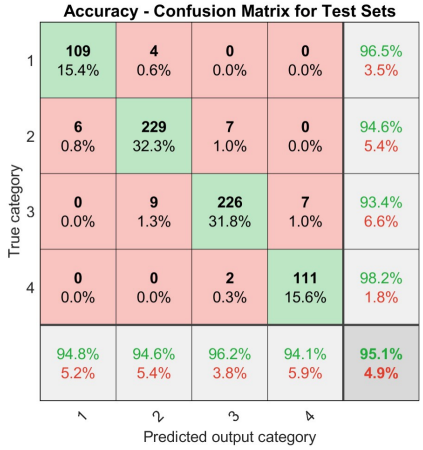

Fig. 10. Confusion matrix of operational accuracy

4.4. Comparative analyses

We compare the constructed models with XGBoost [35], QDA, CNN, CNN-self-attention, GRU-self-attention, and CNN-GRU-self-attention, and by analyzing the accuracy, recall, precision, and F1-score of different models, we can compare the fitting effect of different models.

The comparison graph of prediction accuracy is shown in Figs. 11–13. From the figures, it can be seen that in the neural network algorithm represented by CNN, the decision tree algorithm represented by XGBoost, and the discriminant analysis algorithm represented by QDA, the accuracy of CNN is higher than that of the other two basic models, which is the main reason why the CNN algorithm is chosen for optimization in this paper. After adding a self-attention mechanism, the effect of CNN is obviously improved. GRU can effectively control the information flow through the gating mechanism to overcome the problem of gradient vanishing and has a simple structure and

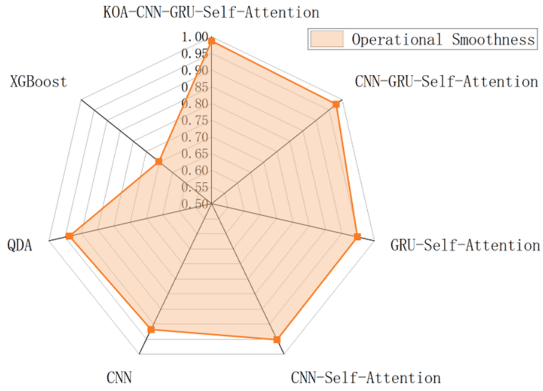

Fig. 11. Comparison of accuracy of operational smoothness

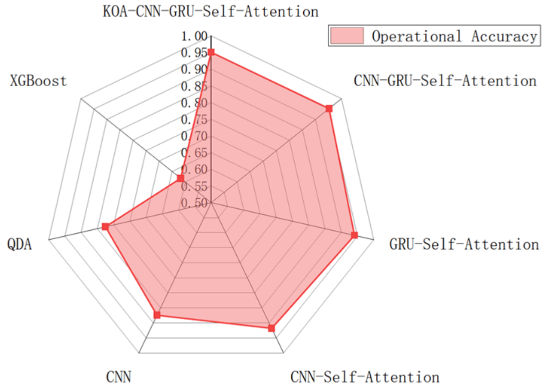

Fig. 12. Comparison of accuracy of operational accuracy

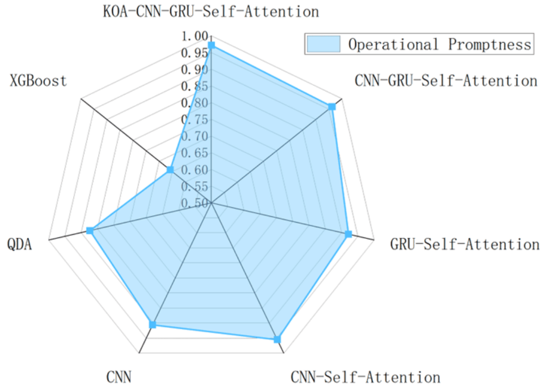

Fig. 13. Comparison of accuracy of operational promptness

high training efficiency. After replacing CNN with GRU, it is found that the effect of CNN-self-attention is comparable with GRU-self-attention. So we fused CNN with GRU, combined the advantages of CNN and GRU, and added the self-attention mechanism, and found that the accuracy was much higher than the model before fusion. To seek a better classification effect and give full play to the advantages of the CNN-GRU-self-attention model, this paper uses KOA to optimize the model hyperparameters to achieve a better combination. It can be seen that the accuracy of the CNN-GRU-self-attention model after KOA optimization has been improved to a certain extent, which is better than all other models. To be able to distinguish the classification effects of different models more comprehensively and intuitively, based on comparing the accuracy, this paper also compares and analyses the precision, recall, and F1-score of all the models, and the results are as shown in Figs. 14–16.

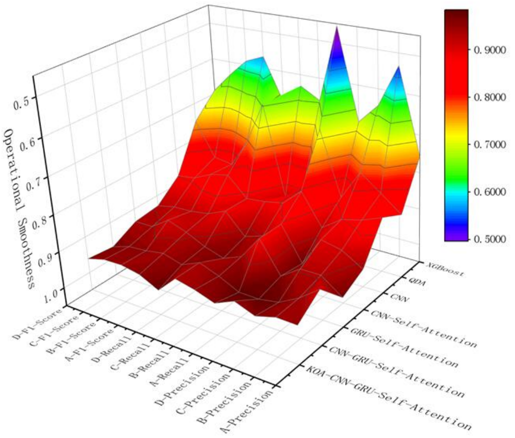

Fig. 14. Comparison of precision & recall & F1-score of operational smoothness

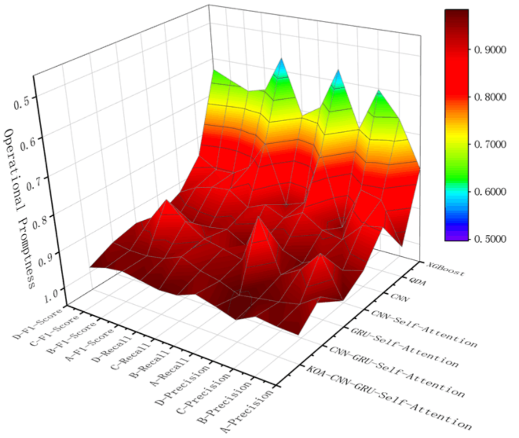

Fig. 15. Comparison of precision & recall & F1-score of operational accuracy

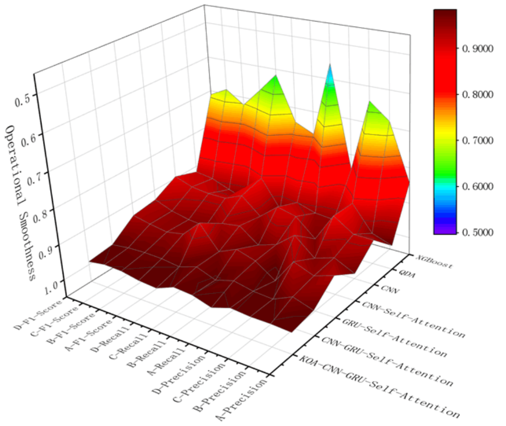

Fig. 16. Comparison of precision & recall & F1-score of operational promptness

From Figs. 14–16, it can be seen that the three base models CNN, XGBoost, and QDA are not so good for classification, but CNN is still better than the other two base models overall. And the fusion models all tend to become better. KOA-CNN-GRU-self-attention has better data in the precision, recall, and F1-score in all categories of the three metrics of operational smoothness, accuracy, and promptness. Although the metrics data of KOA-CNN-GRU-self-attention is slightly worse than CNN-GRU-self-attention in some of the categories, it is better than CNN-GRU-self-attention in general. It proves the effectiveness of the KOA optimization and the excellent effect of the KOA-CNN-GRU-self-attention classification model.

From Table 5, it can be seen that the KOA-CNN-GRU-self-attention accuracy is better than the other models, no matter the operational smoothness, operational accuracy, or operational promptness. In summary, the KOA-CNN-GRU-self-attention assessment model has good classification prediction performance for flight quality assessment and can effectively improve the efficiency of flight quality assessment.

5. CONCLUSIONS

Flight quality and flight safety are inseparable, and QAR data covers most of the important data of flight quality and engine performance status. Using QAR to classify flight quality and identify over-limit and dangerous operations is of great significance to improve the existing flight training system and enhance

Table 5
Data table of classification effects of different models

Models	Cate-gories	Operational accuracy				Operational promptness				Operational smoothness			
		Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score
KOA-CNN-GRU-self-attention	A	0.9507	0.9478	0.9646	0.9561	0.9718	0.9735	0.9735	0.9735	0.9873	0.9912	1.0000	0.9956
	B		0.9463	0.9463	0.9463		0.9707	0.9587	0.9647		0.9876	0.9835	0.9855
	C		0.9617	0.9339	0.9476		0.9634	0.9793	0.9713		0.9875	0.9793	0.9834
	D		0.9407	0.9823	0.9610		0.9911	0.9823	0.9867		0.9826	1.0000	0.9912
CNN-GRU-self-attention	A	0.9521	0.9035	0.9810	0.9406	0.9620	0.9231	0.9897	0.9552	0.9775	0.9813	0.9722	0.9767
	B		0.9436	0.9544	0.9490		0.9605	0.9681	0.9643		0.9630	0.9873	0.9750
	C		0.9726	0.9342	0.9530		0.9721	0.9569	0.9644		0.9874	0.9671	0.9771
	D		0.9820	0.9561	0.9689		0.9804	0.9346	0.9569		0.9836	0.9836	0.9836
GRU-self-attention	A	0.9408	0.9710	0.8790	0.9227	0.9211	0.9800	0.8200	0.8929	0.9479	0.9340	0.9610	0.9473
	B		0.9290	0.9440	0.9033		0.8720	0.9530	0.8452		0.9520	0.9440	0.9565
	C		0.9290	0.9650	0.9364		0.9310	0.9390	0.9107		0.9300	0.9670	0.9480
	D		0.9650	0.9480	0.9364		0.9620	0.9270	0.9419		1.0000	0.9000	0.9369
CNN-self-attention	A	0.9183	0.9407	0.8810	0.9098	0.9549	0.9450	0.9537	0.9493	0.9521	0.9737	0.9569	0.9652
	B		0.9057	0.9286	0.9170		0.9607	0.9442	0.9524		0.9461	0.9744	0.9600
	C		0.8932	0.9457	0.9187		0.9459	0.9722	0.9589		0.9639	0.9339	0.9486
	D		0.9737	0.8880	0.9289		0.9735	0.9402	0.9565		0.9151	0.9417	0.9282
CNN	A	0.8746	0.8450	0.8300	0.8374	0.9056	0.8710	0.8890	0.8799	0.9183	0.9240	0.9170	0.9205
	B		0.8660	0.8760	0.8476		0.8820	0.9190	0.8855		0.9230	0.9040	0.9200
	C		0.8790	0.8860	0.8710		0.9350	0.8950	0.9001		0.8860	0.9460	0.9134
	D		0.9160	0.8910	0.8775		0.9240	0.9170	0.9269		0.9670	0.8990	0.8949
QDA	A	0.8254	0.8764	0.6964	0.7761	0.8732	0.9659	0.7589	0.8500	0.9366	0.9817	0.9469	0.9640
	B		0.8320	0.8765	0.8537		0.8506	0.9136	0.8810		0.9615	0.9298	0.9454
	C		0.7893	0.9095	0.8451		0.8370	0.9339	0.8828		0.8897	0.9630	0.9249
	D		0.8706	0.6607	0.7513		0.9560	0.7699	0.8529		0.9519	0.8839	0.9167
XGBoost	A	0.6170	0.7568	0.4553	0.5686	0.6580	0.7903	0.5678	0.6608	0.7020	0.8421	0.5565	0.6701
	B		0.5315	0.6609	0.5892		0.6694	0.6803	0.6748		0.6842	0.7647	0.7222
	C		0.6431	0.6308	0.6369		0.6026	0.7137	0.6535		0.6392	0.7409	0.6863
	D		0.6979	0.6837	0.6907		0.7172	0.5635	0.6311		0.8421	0.6154	0.7111
PCA-PSO-SVM [4]	D	0.9000	0.9410	–	–	0.9000	0.9410	–	–	0.9000	0.9410	–	–

flight safety. In this paper, we make full use of the QAR data, propose three kinds of indicators to measure flight quality according to the existing standards, and utilize PCA to conduct a comprehensive evaluation, substitute it into the model training, and finally propose an objective full-stage flight quality evaluation model based on KOA-CNN-GRU-self-attention, which is free from the one-sidedness of the previous research, and improves the accuracy of the classification, and the method is new and novel. The KOA algorithm optimizes the parameters of CNN-GRU-self-attention, and automatically optimizes and adjusts the parameters of CNN-GRU-self-attention while successfully and reasonably evaluating flight quality, and the introduction of KOA can automatically adjust the parameters in the model for different data. This adjustment method can also adjust the parameters of other models in other deep learning algorithms, such as the number of convolutional kernels, the probability of discarding layers, etc., which is expected to achieve better results when training 3D image data. Meanwhile, with the progress of technology, it has become possible to install real-time transmission of QAR data recording systems on civil airliners in the future, and this paper is of strong practical significance to reasonably analyze and evaluate the complex QAR data to evaluate the flight quality, and it is also expected to provide a more reasonable categorization of the flight level of each pilot.

Although the KOA-CNN-GRU-self-attention model is more comprehensive for the analysis of QAR data, this study does not combine the environmental factors and other special circumstances for a comprehensive analysis, and future research can consider more comprehensive factors. Moreover, when choosing KOA hyperparameters and optimized hyperparameters in this paper, the optimum achieved may be a local optimum, and in the future, the idea of a confidence test can be borrowed to set a credibility index to measure the optimized parameters as the global optimum.

REFERENCES

[1] M. Xiong, H. Wang, Y. Diew Wong, and Z. Hou, "Enhancing aviation safety and mitigating accidents: A study on aviation safety hazard identification," *Adv. Eng. Inform.*, vol. 62, p. 102732, 2024, doi: 10.1016/j.aei.2024.102732.

[2] M. Xiong, H. Wang, C. Che, and M. Sun, "Application of text mining and coupling theory to depth cognition of aviation safety risk," *Reliab. Eng. Syst. Saf.*, vol. 245, p. 110032, 2024, doi: 10.1016/j.ress.2024.110032.

[3] "Preliminary Report on the Investigation of the Flight Accident of China Eastern Airlines MU5735 Aircraft on March 21," Civil Aviation Administration. [Online]. Available: https://www.gov.cn/xinwen/2022-04/20/content_5686311.htm [Accessed: 15. Aug. 2024].

[4] "How much damage was caused by the China Eastern Airlines plane crash?" Big Eyes World/NetEase. [Online]. Available: https://www.163.com/dy/article/H3SJE6OM0529SM26.html [Accessed: 15. Aug. 2024]

[5] F. Li, X. Xu, J. Li, H. Hu, M. Zhao, and H. Sun, "Wind Shear Operation-Based Competency Assessment Model for Civil Aviation Pilots," *Aerospace*, vol. 11, no. 5, p. 363, 2024, doi: 10.3390/aerospace11050363.

[6] Y. Gao and X-Z. Wang, "Research on Performance Assessment of Overall Aero-engine Based on QAR Data," *Sci. Technol. Eng.*, vol. 16, no. 25, pp. 322–326, 2016, doi: 10.3969/j.issn.1671-1815.2016.25.056.

[7] L. Tong, "Guiding Opinions on Comprehensively Deepening the Reform of Flight Training in Transportation." [Online]. Available: https://mp.weixin.qq.com/s?__biz=Mzg2NDg2NjUyMA==&mid=2247488806&idx=1&sn=b9614f29e824c429136e8a7b0f7c92ed&source=41#wechat_redirect [Accessed: 26.Jan.2024].

[8] L.Ch. Shunxin, "Yang.Research on a post-competency model of civil aviation flight cadets," *Int. J. Occup. Saf. Ergon.*, vol. 29, no. 4, pp. 1558–1571, 2023, doi: 10.1080/10803548.2023.2259165.

[9] W.-L. Yuan, C.-Y. Lu, W. Lu, and S. He, "Flight Quality Evaluation Based on Machine Learning," *Sci. Technol. Eng.*, vol. 21, no. 19, pp. 8262–8269, 2021, doi: 10.3969/j.issn.1671-1815.2021.19.055.

[10] W. Benchi, D. Jun, D. Chao, W. Zhentao, and Z. Shuai, "Evaluation of Flight Quality during Aircraft Takeoff Phase Based on AHP-TOPSIS Method," *Flight Mech.*, vol. 37, no. 1, pp. 80–88, 2019, doi: 10.13645/j.cnki.f.d.20181106.018.

[11] L. Zhong, Z. Qu, and F. Mora-Camino, "A model-based flight qualities evaluation approach for civil aircraft," *Concurrency Computat. Pract. Exper.*, vol. 31, no. 10, p. e4723, 2019, doi: 10.1002/cpe.4723.

[12] X. Miaoying, "Research on Flight Quality in Landing Phase Based on Time Series Analysis," MSc. thesis. Civil Aviation Flight University of China, China, 2023.

[13] F. Rizvi *et al.*, "An evolutionary KNN model for DDoS assault detection using genetic algorithm based optimization," *Multimed. Tools Appl.*, 2024, doi: 10.1007/s11042-024-18744-5.

[14] K. Fujimori, Y. Goto, Y. Liu, and M. Taniguchi, "Sparse principal component analysis for high-dimensional stationary time series," *Scand. J. Stat.*, vol. 50, no. 4, pp. 1953–1983, 2024, doi: 10.1111/sjos.12664.

[15] R. Liu, J. Shi, G. Sun, S. Lin, and F. Li, "A Short-term net load hybrid forecasting method based on VW-KA and QR-CNN-GRU," *Electr. Power Syst. Res.*, vol. 232, p. 110384, 2024, doi: 10.1016/j.epsr.2024.110384.

[16] K. Deeba *et al.*, "A disease monitoring system using multi-class capsule network for agricultural enhancement in muskmelon," *Multimed. Tools Appl.*, 2024, doi: 10.1007/s11042-024-18717-8.

[17] Ch. Wang and H. Wang, "Cascaded feature fusion with multi-level self-attention mechanism for object detection," *Pattern Recognit.*, vol. 138, p. 109377, 2023, doi: 10.1016/j.patcog.2023.109377.

[18] D.K.K. Reddy, J. Nayak, H.S. Behera, V. Shanmuganathan, W. Viriyasitavat, and G. Dhiman, "A Systematic Literature Review on Swarm Intelligence Based Intrusion Detection System: Past, Present and Future," *Arch. Comput. Meth. Eng.*, vol. 31, pp. 2717–2784, 2024, doi: 10.1007/s11831-023-10059-2.

[19] M. Abdel-Basset, R. Mohamed, S.A. Abdel Azeem, M. Jameel, and M. Abouhawwash, "Kepler optimization algorithm: A new metaheuristic algorithm inspired by Kepler's laws of planetary motion," *Knowledge-Based Syst.*, vol. 268, p. 110454, 2023, doi: 10.1016/j.knosys.2023.110454.

[20] K. Fujimori, Y. Goto, Y. Liu, and Taniguchi, "Sparse principal component analysis for high-dimensional stationary time se-

ries," *Scand. J. Stat.*, vol. 50, no. 4, pp. 1953–1983, 2024, doi: 10.1111/sjos.12664.

[21] X. Dong *et al.*, "To Raise or Not To Raise: The Autonomous Learning Rate Question," *arXiv:2106.08767*, 2021, doi: 10.48550/arXiv.2106.08767.

[22] F. Boray Tek, Ý. Çam, and D. Karlý, "Adaptive convolution kernel for artificial neural networks," *arXiv:2009.06385*, 2020, doi: 10.48550/arXiv.2009.06385.

[23] J. Noh, H.-J. Park, J. S. Kim, and S.-J. Hwang, "Gated Recurrent Unit with Genetic Algorithm for Product Demand Forecasting in Supply Chain Management," *Mathematics*, vol. 8, no. 4, p. 565, 2020, doi: 10.3390/math8040565.

[24] S. Ruitao, "Pilot Safety Performance Evaluation", M.A. thesis, Civil Aviation University of China, China, 2018.

[25] R.-S. Sun and Y.-Bi. Xiao, "Research on indicating structure for operation characteristic of civil aviation pilots based on QAR data," *J. Saf. Sci. Technol.*, vol. 8, pp. 49–54, 2012.

[26] F.B. Farahabadi, K.F. Vajargah, and R. Farnoosh, "Dimension reduction big data using recognition of data features based on copula function and principal component analysis," *Adv. Math. Phys.*, vol. 2021, no. 2, p. 9967368, 2021, doi: 10.1155/2021/9967368.

[27] Y. Lecun *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989, doi: 10.1162/neco.1989.1.4.541.

[28] A.E. Ilesanmi, T. Ilesanmi, and G.A. Gbotoso, "A systematic review of retinal fundus image segmentation and classification methods using convolutional neural networks," *Healthc. Anal.*, vol. 4, p. 100261, 2023, doi: 10.1016/j.health.2023.100261.

[29] K. Cho *et al.*, "On the Properties of Neural Machine Translation: Encoder-Decoder Approaches," *Empirical Methods in Natural Language Processing Conference(EMNLP 2014)*, Qatar, 2014.

[30] K. Zarzycki and M. Ławryńczuk, "Advanced predictive control for GRU and LSTM networks," *Inf. Sci.*, vol. 616, pp 229–254, 2022, doi: 10.1016/j.ins.2022.10.078.

[31] Vaswani A *et al.*, "Attention Is All You Need," *31st Conference on Neural Information Processing Systems (NIPS 2017)*, USA, 2017, doi: 10.48550/ARXIV.1706.03762.

[32] F. Qian *et al.*, "A model for estimating total NOx emissions of heavy-duty diesel vehicles in Kaifeng based on random forest," *Acta Sci. Circumstantiae*, vol. 43, no. 2, pp. 391–407, 2023, doi: 10.13671/j.hjkxxb.2022.0365.

[33] D.J. Hand, P. Christen, and N. Kirielle, "F*: an interpretable transformation of the f-measure," *Mach. Learn.*, vol. 110, no. 3, pp. 451–456, 2021, doi: 10.1007/s10994-021-05964-1.

[34] S. Sharma, K. Gupta, D. Gupta, S. Rani, and G. Dhiman, "An Insight Survey on Sensor Errors and Fault Detection Techniques in Smart Spaces," *CMES-Comp. Model. Eng. Sci.*, vol. 138, pp. 2029–2059, 2024.doi: 10.32604/cmes.2023.029997.

[35] N. Sehrawat *et al.*, "A power prediction approach for a solar-powered aerial vehicle enhanced by stacked machine learning technique," *Comp. Electr. Eng.*, vol. 115, p. 109128, 2024, doi: 10.1016/j.compeleceng.2024.109128.