

Research Article

Analysis of Single-Pilot Intention Modeling in Commercial Aviation

Lei Dong ^{1,2}, Hongbing Chen ², Changxiao Zhao ^{1,2} and Peng Wang ^{1,2}

¹Key Laboratory of Civil Aircraft Airworthiness Technology, Civil Aviation University of China, Tianjin 300300, China

²College of Safety Science and Engineering, Civil Aviation University of China, Tianjin 300300, China

Correspondence should be addressed to Changxiao Zhao; cxzhao@cauc.edu.cn

Received 20 November 2022; Revised 15 March 2023; Accepted 21 March 2023; Published 10 April 2023

Academic Editor: Binbin Yan

Copyright © 2023 Lei Dong et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the continual enhancement of the onboard avionics, the minimum flight crew has been downsized from five to two-person crew mode, and reduced crew operation has drawn extensive attention from aviation experts. Single-pilot operation (SPO) mode warrants careful account and research. This study investigated the intention modeling of commercial aviation single pilot based on the bidirectional long short-term memory (BiLSTM), mining the intention tendency of pilots' behavior through artificial intelligence technology. This was done to avoid safety hazards caused by different intents and inconsistent operations of the single pilot and the cockpit automation system. The classification task of a single pilot's behavior is the core of intention recognition. Various operation items contribute differently to the classification. To construct the interaction dataset and encode it into time series features, a single-pilot experiment is specifically performed, wherein the experience of an expert is summarized into single-pilot intent labels. The deep information in the feature vector of a single-pilot operation item is captured by the BiLSTM network, and the neural weight is adaptively assigned by the training mechanism. The operation sequence with the feature data is finally loaded into the softmax layer for intention classification. The proposed method is evaluated against long short-term memory (LSTM), term frequency-inverse document frequency (TF-IDF), convolutional neural network (CNN), Naive Bayesian (NB), and distributed representation's intention modeling techniques. Because the proposed methods have higher F1 scores, the model can effectively share real-time information about the single-pilot intention with the cockpit automation system.

1. Introduction

Single-pilot operation (SPO) is a key research concept for next-generation aviation technology [1]. For the SPO mode to be as effective and of high quality as the dual-pilot operating mode, the capabilities of the avionics system must be enhanced [2]. With the continuous development of the commercial SPO concept, the collaborative flight organization structure with automation at its core inevitably increases the overall complexity [3], which increases safety risks.

The National Aeronautics and Space Administration (NASA) is studying the relevant aspects of SPO [4], which relies on developing and integrating a higher level of automation than the two-person crew mode of the past. Recently, a decline in crew numbers in the cockpit of commercial aircraft and an increase in air traffic density have

decreased aviation accident rates [5]. Automation is the main factor addressing these challenges [6]. Despite all stakeholders agreeing on the effectiveness of high automation on safety enhancement, they are also concerned that extreme automation may cause profound changes in human-machine cooperation or introduce new human errors [7]. Automation has particularly shown many forms of bad behavior, like automation bias [8], automation surprise, mode confusion [9], unexplained decisions, and decision errors [10], that have intensified stakeholders' concerns. Since SPO will necessitate increasingly autonomous systems to carry out missions previously performed by the copilot [11], aircraft automation is defined as a cooperator that would act with intention. A key discovery is that conflicting intentions may arise from insufficient support of partnerships between the pilot crew and aircraft automation

[12]. The transfer and execution of control rights between the single pilot and the cockpit automation systems will inevitably conflict when their intended goals diverge [13]. Consequently, automation must be user-centered [14] and adapt to the single-pilot intentions, which must establish passive or implicit perception capabilities to encourage pilot-automation partnership.

Human factor research has shown that because of one flight control operator under the SPO mode, the piloting skills and physical condition of the sole pilot directly affect the safety of the flight. With the evolution of SPO mode technology, automation pushes single pilots into the role of a supervisor rather than detaching them out of control. Generally, human pilots are required to monitor the automation partner to ensure its efficient assistance during airplane flights. However, bidirectional monitoring forms a crucial constituent of team collaboration, which enables the pilot and automation partner to examine each other, similar to interpersonal cooperation. In particular, SPO human-machine interaction can be facilitated when real-time monitoring objects are dynamically migrated between a single pilot and an automation partner. The overall objective of the European commission-funded project A-PiMod (applying pilot models for safer aircraft) [15] is to develop a pilot model that enables the advanced cockpit assistance system to learn about the single-pilot cognitive state. The critical function performed by the automation system in charge of supervision behaviors continues to provide feedback regarding pilot cognition or intention. The automation system must find the implicit intention data from the behavior characteristics of a single pilot with time series changes, which enables active recognition and complete understanding of the single-pilot intention. This will support free flight oriented toward the single-pilot intention.

The intention recognition process directly drives the generation of situation awareness. Situational unawareness may be the result of poor intention recognition. Statistics show that human error accounts for about 70% of aviation accidents [16]. How to avoid the potential threat caused by action errors and perceive the unsafe decision intention or manipulation behavior of a single pilot has hence become one of the research goals of developing SPO mode technology [17].

For the pilot intention modeling, the disquisition difference of relevant technical basis primarily exists in the selection of data type and study approach. Mccruer [18] examined the pilot transfer function model based on classical and modern control theories. The model was used to describe actual patterns of pilot behavior. The pilot intent and error recognition (PIER) module of the crew assistant military aircraft (CAMA) is another method described in [19]. The method was based on fuzzy logic algorithms, and intent recognition was first defined as classification problems. Hayashi [20] used instrument scanning to simulate the aircraft pilot's attention switching behavior. To analyze the pilots' eye-movement statistical data, the researcher employed a hidden Markov model (HMM). However, our focus was on the physical interactions rather than the scanning behavior. In [21], NASA launched a human performance modeling pro-

ject, analyzing pilot performance using five cognitive modeling tools. The project supports the prediction of pilot errors and behavior. The cognitive architecture for safety critical task simulation (CASCaS) was proposed in [22]. CASCaS was defined as an operator model that perceives associated intentions through discrete action sequences rather than classifying intentions with probability. The A-PiMod project designed a pilot model based on the adaptive automation concept in [15]. The A-PiMod assessed the pilot state through multimodal interactions between pilots and the cockpit. The literature suggests that to model single-pilot behavior, a fully adaptive automated approach is required. The HMM-based intent recognition module described in [12] could be viewed as a knowledge engineering method based on rules. Annotated interaction data were used to learn the HMM parameters. The module only inferred eight basic flight tasks, including "set flaps," "set approach," and "change heading," owing to the model's performance limitations.

According to a domestic and foreign research foundation, the current research on pilot intention modeling primarily focuses on system modeling and analysis verification based on classical algorithms, while there are few studies on pilot intent recognition based on intelligent system. The single-pilot intention is implemented through a series of pilot manipulation behaviors that exhibit the typical characteristics of the time sequence change. Nevertheless, the majority of the above approaches are employed to recognize static characteristics; consequently, there is no apparent superiority in handling process events with timing features like aircraft piloting. With the rapid advancement of deep learning, this study proposes an intention modeling method for commercial aviation single pilots based on bidirectional long short-term memory (BiLSTM), which treats intention recognition as a timing classification problem. The proposed methods, whether from the evaluation of human-machine interaction design or the model capability, can handle the increasingly complex challenges of aircraft pilot intent recognition.

This study is structured as follows: Section 2 describes the relevant theoretical basis; Section 3 provides an overview of a single-pilot intention model; Section 4 defines the intention space and characteristics and designs the SPO test; Section 5 details the single-pilot intention modeling method based on the BiLSTM network; a case study is conducted in Section 6 to demonstrate the interaction dataset and perform model evaluation and comparative analysis; and finally, discussion and summary are made.

2. Preliminaries

The classification task of time series information is the core of single-pilot intent recognition. Different data preprocessing and feature capture techniques contribute differently to the classification. Learning a vector representation for an operation item is crucial. The one-hot vectors created using one-hot encoding are the most used representation. Deep learning's BiLSTM is used to better capture the sequence information because of the vector's sparse feature.

2.1. Long Short-Term Memory (LSTM). The traditional neural network model is inefficient in processing sequence learning because it cannot explain the correlation between the front and back of the sequence [23]. Recurrent neural networks (RNNs) are a type of sequence learning model that can dynamically learn sequence features. However, owing to the gradient disappearance or the explosion problem [24], traditional RNNs are difficult to train. In 1997, Hochreiter and Schmidhuber proposed LSTM [25], which introduced a gating structure and solved the aforementioned problems. The LSTM unit diagram is shown in Figure 1.

The cell state at the time step t contains the data that the LSTM unit learned from the previous time steps, and the hidden state at the time step t contains the output of the LSTM unit for the time step. The gate structure, which does not provide information but is used to limit the amount of information, must control the information update of the cell state. The gate structure is a type of multilevel feature selection technique.

The input gate i_t regulates the frequency of cell state update. The forget gate f_t regulates the frequency of cell state forget. The output gate o_t manages the frequency of adding cell state to the hidden state. The cell candidate g_t is responsible for adding information to the cell state. The formulas are as follows:

$$\begin{aligned} i_t &= \sigma_g(W_i x_t + R_i h_{t-1} + b_i), \\ f_t &= \sigma_g(W_f x_t + R_f h_{t-1} + b_f), \\ o_t &= \sigma_g(W_o x_t + R_o h_{t-1} + b_o), \\ g_t &= \sigma_c(W_g x_t + R_g h_{t-1} + b_g). \end{aligned} \quad (1)$$

In above formulas, x_t is the input of the time step t . W_i , W_f , W_o , and W_g are the components of the input weights W . R_i , R_f , R_o , and R_g are the components of the recurrent weights R . b_i , b_f , b_o , and b_g are the components of the bias b . The σ_g denotes the sigmoid function, and σ_c denotes the hyperbolic tangent function.

At the time step t , the cell state c_t and the hidden state h_t are given by

$$\begin{aligned} c_t &= \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \\ \mathbf{h}_t &= \mathbf{o}_t \odot \sigma_c(\mathbf{c}_t), \end{aligned} \quad (2)$$

where \odot denotes the Hadamard product.

2.2. Bidirectional Long Short-Term Memory (BiLSTM). In the traditional RNN and LSTM model, information can only be propagated forward, causing the state of the time step t to depend only on the data before the time step t [23]. The BiLSTM network is proposed to make each instant include the data of the upper and lower time steps.

The BiLSTM network uses both forward and backward observations to enhance performance. Thus, the BiLSTM network concatenates the forward and backward hidden states before passing them to the next network layer [26]. The following equation defines the corresponding calcula-

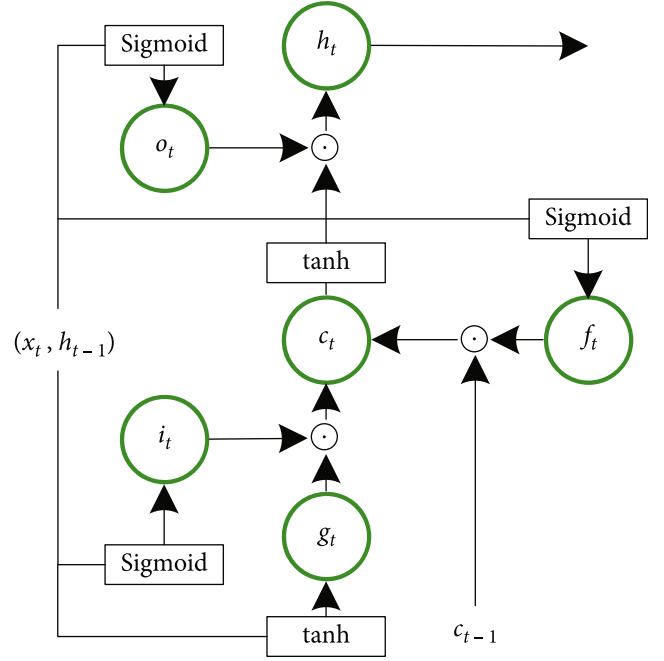


FIGURE 1: The diagram of LSTM unit structure.

tion formulas [27] and represents the forward and backward processes, respectively.

$$\begin{aligned} \vec{\mathbf{i}}_t &= \sigma_g(\vec{\mathbf{W}}_i \mathbf{x}_t + \vec{\mathbf{R}}_i \mathbf{h}_{t-1} + \vec{\mathbf{b}}_i), \\ \vec{\mathbf{f}}_t &= \sigma_g(\vec{\mathbf{W}}_f \mathbf{x}_t + \vec{\mathbf{R}}_f \mathbf{h}_{t-1} + \vec{\mathbf{b}}_f), \\ \vec{\mathbf{g}}_t &= \sigma_c(\vec{\mathbf{W}}_g \mathbf{x}_t + \vec{\mathbf{R}}_g \mathbf{h}_{t-1} + \vec{\mathbf{b}}_g), \\ \vec{\mathbf{o}}_t &= \sigma_g(\vec{\mathbf{W}}_o \mathbf{x}_t + \vec{\mathbf{R}}_o \mathbf{h}_{t-1} + \vec{\mathbf{b}}_o), \\ \vec{\mathbf{c}}_t &= \vec{\mathbf{f}}_t \odot \vec{\mathbf{c}}_{t-1} + \vec{\mathbf{i}}_t \odot \vec{\mathbf{g}}_t, \\ \vec{\mathbf{h}}_t &= \vec{\mathbf{o}}_t \odot \sigma_c(\vec{\mathbf{c}}_t), \\ \overleftarrow{\mathbf{i}}_t &= \sigma_g(\overleftarrow{\mathbf{W}}_i \mathbf{x}_t + \overleftarrow{\mathbf{R}}_i \mathbf{h}_{t+1} + \overleftarrow{\mathbf{b}}_i), \\ \overleftarrow{\mathbf{f}}_t &= \sigma_g(\overleftarrow{\mathbf{W}}_f \mathbf{x}_t + \overleftarrow{\mathbf{R}}_f \mathbf{h}_{t+1} + \overleftarrow{\mathbf{b}}_f), \\ \overleftarrow{\mathbf{g}}_t &= \sigma_c(\overleftarrow{\mathbf{W}}_g \mathbf{x}_t + \overleftarrow{\mathbf{R}}_g \mathbf{h}_{t+1} + \overleftarrow{\mathbf{b}}_g), \\ \overleftarrow{\mathbf{o}}_t &= \sigma_g(\overleftarrow{\mathbf{W}}_o \mathbf{x}_t + \overleftarrow{\mathbf{R}}_o \mathbf{h}_{t+1} + \overleftarrow{\mathbf{b}}_o), \\ \overleftarrow{\mathbf{c}}_t &= \overleftarrow{\mathbf{f}}_t \odot \overleftarrow{\mathbf{c}}_{t+1} + \overleftarrow{\mathbf{i}}_t \odot \overleftarrow{\mathbf{g}}_t, \\ \overleftarrow{\mathbf{h}}_t &= \overleftarrow{\mathbf{o}}_t \odot \sigma_c(\overleftarrow{\mathbf{c}}_t). \end{aligned} \quad (3)$$

When the input is $[\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]$, the output of the forward hidden state of the BiLSTM network is defined as $[\vec{\mathbf{h}}_1, \vec{\mathbf{h}}_2, \dots, \vec{\mathbf{h}}_n]$, and the output of the backward hidden state of

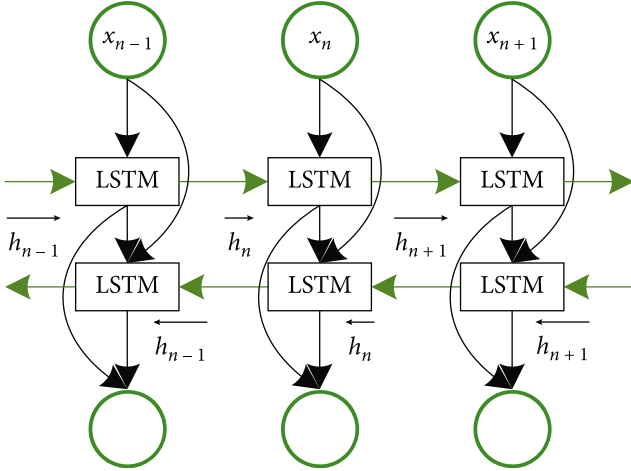


FIGURE 2: The net structure of BiLSTM.

the BiLSTM network is defined as $[\overleftarrow{\mathbf{h}}_1, \overleftarrow{\mathbf{h}}_2, \dots, \overleftarrow{\mathbf{h}}_n]$; then, the formula of the hidden state matrix H is

$$\mathbf{H} = [\overrightarrow{\mathbf{h}}_1, \overrightarrow{\mathbf{h}}_2, \dots, \overrightarrow{\mathbf{h}}_n; \overleftarrow{\mathbf{h}}_1, \overleftarrow{\mathbf{h}}_2, \dots, \overleftarrow{\mathbf{h}}_n]. \quad (4)$$

A graphical illustration of the BiLSTM network is shown in Figure 2.

2.3. One-Hot Encoding. One-hot encoding [28] encodes N states using N -digit status register, and the result is an N -dimension binary vector.

Assuming that the state number of the status register is k , the positive integer is $j \in \{1, \dots, k\}$, and the one-hot vector $\chi(j)$ is defined as a k -dimensional real column vector; there is

$$\chi(j)_l = \begin{cases} 1, & l = j, \\ 0, & l \neq j, \end{cases} \quad (5)$$

where $\chi(j)_l$ is the l -th element of the vector $\chi(j)$.

3. Single-Pilot Intention Model

The theoretical foundation for addressing thinking deviation and intention inconsistency in the collaborative process between a single pilot and cockpit automation system is the effective modeling of single-pilot intent. To enhance the single-pilot-automation partnership, this model was developed with reference to the European project A-PiMod. Figure 3 shows the technical scheme block diagram for single-pilot intention model. The human-machine multimodal interface (HMMI) collects real-time data on interactions with the test cockpit systems. The intention inference module provides the cockpit automation system with an adequate understanding of the flight intents that the single pilot is performing. The situation awareness assessment can detect unsafe operation actions and develop explanatory mechanisms for observed and perceived single-pilot behavior. The cockpit risk assessment continuously provides information about the risk for all currently single-pilot behavior. The

results of these two assessments will impact the use of a proactive risk or safety tool, supporting many functions such as cross-check of safety critical tasks, decision assistance, and dynamic assignment of flight tasks. By applying the intention recognition results to the assistance tool of airplane driving, the cockpit automation system could properly alert the single pilot in a risk status in the light of a certain flight intent to assure that the aviator is able to precisely maneuver the airplane.

4. Single-Pilot Operation Experiment

4.1. Airworthiness Certification Requirements. According to the in-depth analysis of the airworthiness requirements of FAR 25.1523 “minimum flight crew,” when approving the minimum flight crew, the airworthiness authority shall consider the operational complexity and corresponding workload level of a single pilot under the architecture in Figure 3. FAA TC-13/44 (flight deck controls and displays), CAAP 5.59-1(0) (single-pilot human factors), and CAP 737 (flight crew human factors) consider the problem of human factors in the cockpit from the aspects of human-machine interface design and multimodal input equipment. According to AC25.1523, the airworthiness authority must pay special attention when one pilot is incapacitated to work for some reason, while the other pilot must continue to safely fly and land the plane. The situation has implicitly adapted to the SPO. Additionally, it is known from the flight manual of a particular commercial aircraft that the authority and manufacturer have established standard operating for various flight tasks in the two-person crew operation. Flight monitoring, inspection, navigation, communication, and other functions related to the copilot in the standard operating procedures are appropriately simplified and assigned to the main pilot for execution, and a flight simulation system is used to conduct the SPO test. According to AC25.1523, although the task allocation method of the SPO test will increase the workload of the single pilot, it can still ensure safety.

4.2. Single-Pilot Operation Test. To gather the required data to train and validate the single-pilot intention model, we conducted SPO tests in the digital experimental cockpit (DECO) of the Civil Aviation University of China in Tianjin. The DECO is a modular flight simulation system that can simulate the most advanced cockpits that aircraft manufacturers are currently producing. Figure 4 shows an image of the DECO. DECO can fulfill the collection needs of the SPO test interactive data stream after secondary development. We invited three cadet pilots to fly simulations in the DECO. All these participants were experienced in flying a B737-800. A series of experimental flight scenarios have been chosen to evaluate the single-pilot intention modeling concept. The scenarios integrated a “theatre approach” and incorporated a variety of task elements. Each participant flew multiple times in each flight scenario and completed the given flight task. Each scenario lasted approximately one hour, commencing with the cockpit preparation phase and ending with the landing phase. Before each SPO test, the participant was randomly assigned a flight plan and required to perform several mission events (take-off, climb, descent, etc.) in order. During the test, the

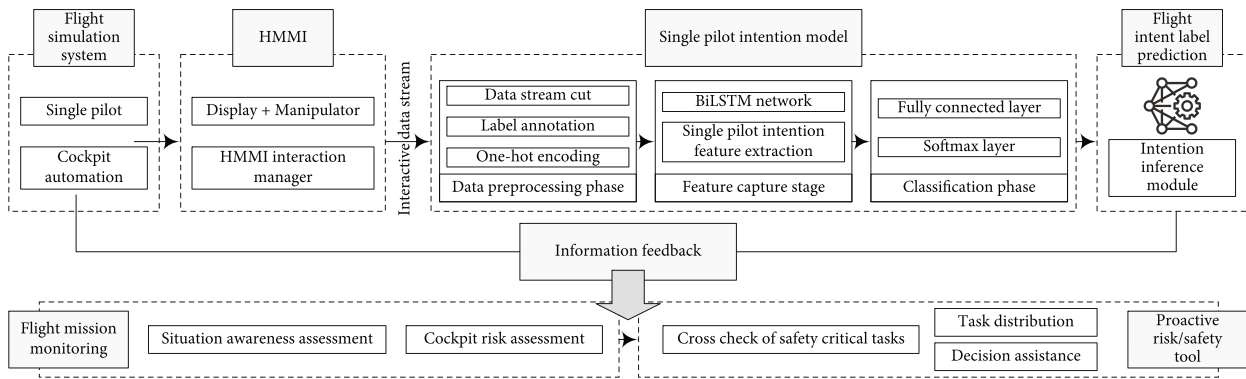


FIGURE 3: The technical scheme block diagram for single-pilot intention model.



FIGURE 4: DECO flight simulator used for dynamic interaction data acquisition.

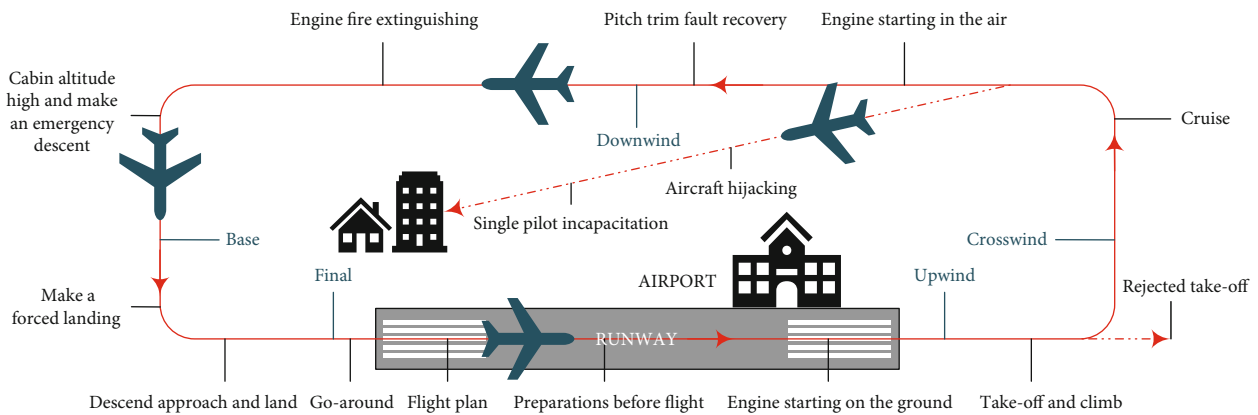


FIGURE 5: Flight diagram of commercial aircraft traffic pattern.

participant received random emergency alerts and was expected to execute emergency operating procedures in response. Both the flight path and each mission event of the single-pilot test are indicated in Figure 5.

4.3. Single-Pilot Intention Space. The single-pilot intention space varies with application scenarios and mission forms. It needs to be defined together with a unique safety requirement of the SPO mode. For the normal or emergency operation process of the SPO mode, this study established a single-pilot intention space including 11 types of flight intent labels. As shown in Figure 5, these labels include engine starting on the ground, take-off and climb, descend approach and land, go-around, cabin altitude high and make an emergency descent, make a forced landing, engine fire extinguishing, engine starting in the air, pitch trim fault

recovery, aircraft hijacking, and single-pilot incapacitation. The issue of inferring the single-pilot intentions must be modeled and resolved in conjunction with the aforementioned flight intent labels. The aforementioned flight intent labels can be modified to fit a particular situation, and their quantity can be increased to meet specific requirements. Given that in the current modeling, even if the pilot operates according to an intention unlisted here, the model will estimate either of the 11 flight intents, which is incorrect. To enable a single-pilot intention model to reject examples from unknown intentions while performing N -class classification tasks [29], class $(N + 1)$ is defined as the unknown intent label. Despite the lack of prior knowledge about the unknown intention, we can still use the known flight intent labels to detect the unknown [30]. The uncertainty threshold is defined based on the probability distribution of the

softmax layer output, which further simplifies the issue of unknown intention recognition.

4.4. Single-Pilot Intention Recognition Feature. Reviewing a single-pilot intention necessitates using a function that enables automation to continuously perform surveillance actions with low costs but high benefits. This includes whether the information source is accessible or difficult to locate, whether the information provided is easy or hard to understand, and whether the sampled information source is current or not [31]. DECO's main goal is to provide data collectors with maximum flexibility, to meet the aforementioned specifications.

To control the flight of the aircraft, the sole pilot must display a specific behavior, and the cockpit must support the operation of the flight crew. A single pilot must interact with the flight simulation system in certain ways, such as "start right engine," "autothrottle off," "select course," "press TOGA," and "landing gear DN." These interactions are called operation items. This study focused on 92 of these interactions because they are essential for triggering flight mission events and are therefore closely related to task intent data. The operation items collected during the SPO test will form an operation sequence that can accurately reflect the single pilot's decision-making intent as $\mathbf{Y} = [y_{i1} y_{i2} \cdots y_{is}]$, where s is the length of the sequence.

5. Single-Pilot Intention Modeling Method Based on BiLSTM

5.1. Vectorization of Single-Pilot Operation Item. Vectorizing the single-pilot operation items, that is, preprocessing these data into a format that the BiLSTM layer can directly accept and understand, is the first step of single-pilot intention modeling. One-hot encoding is a classical vectorization technique.

Let $b(y_i)$ be a positive integer index of the operation item y_i with $b(y_i) = i$, $i \in \{1, \dots, 92\}$. Suppose there is a one-hot vector $\chi(b(y_i))$, according to formula (5), there is

$$\chi(b(y_i))_l = \begin{cases} 1, & l = b(y_i), \\ 0, & l \neq b(y_i). \end{cases} \quad (6)$$

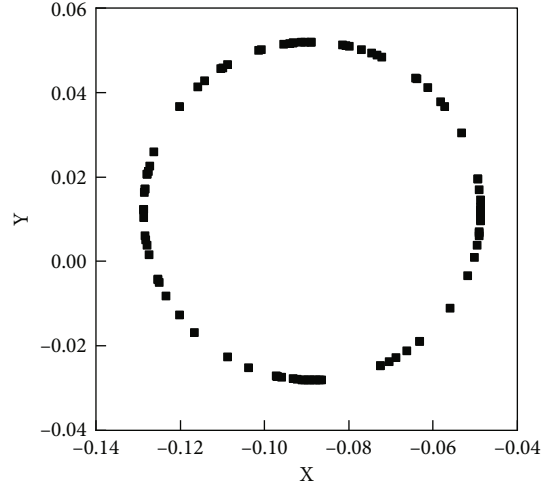
According to the order of operation item y_i in the operation sequence, combined $\chi(b(y_i))$ to generate the one-hot matrix M , there is

$$\mathbf{M} = [\chi(b(y_{i1})) \chi(b(y_{i2})) \cdots \chi(b(y_{is}))]_{92 \times s}. \quad (7)$$

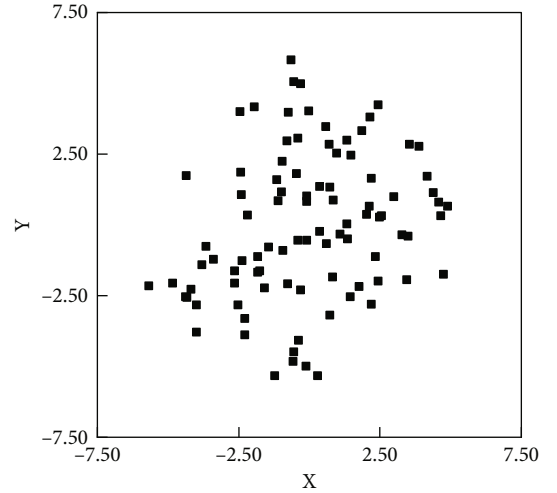
Natural language processing (NLP) theory states that the correlation increases as the Euclidean distance of a word vector decreases and decreases as the Euclidean distance increases. Therefore, the one-hot vector of the single-pilot operation item also includes an implicit semantic correlation. Given that the Euclidean distances between two different one-hot vectors are always equal, there is

$$\|\chi(b(y_p)) - \chi(b(y_q))\|_2 = \|\chi(b(y_m)) - \chi(b(y_n))\|_2, \quad (8)$$

where $p \neq q, m \neq n$.



(a)



(b)

FIGURE 6: Spatial visualization of single-pilot operation item vectors. (a) Based on the one-hot encoding method. (b) Based on distributed representation method.

The semantic correlation between one-hot encoded single-pilot operation items is the same. This means that the one-hot encoding actively ignores the semantic data while primarily preserving the time series relationship. Compared with the distributed representation method [32], the one-hot encoding results in a more concise expression of the operation item and effectively creates a one-hot matrix with high representativeness with the operation sequence, which is more conducive to feature capture. Figure 6 shows the visualization outcomes of 92 single-pilot operation item vectors in low-dimensional space using T-distributed stochastic neighbor embedding (T-SNE). Based on the one-hot encoding technique, the distance from the circle's center to each single-pilot operation item is the same, specifically as shown in Figure 6(a).

5.2. Single-Pilot Intention Model Based on BiLSTM. As a crucial way in understanding a single pilot's behavior, intention recognition can be abstracted as intention classification. This

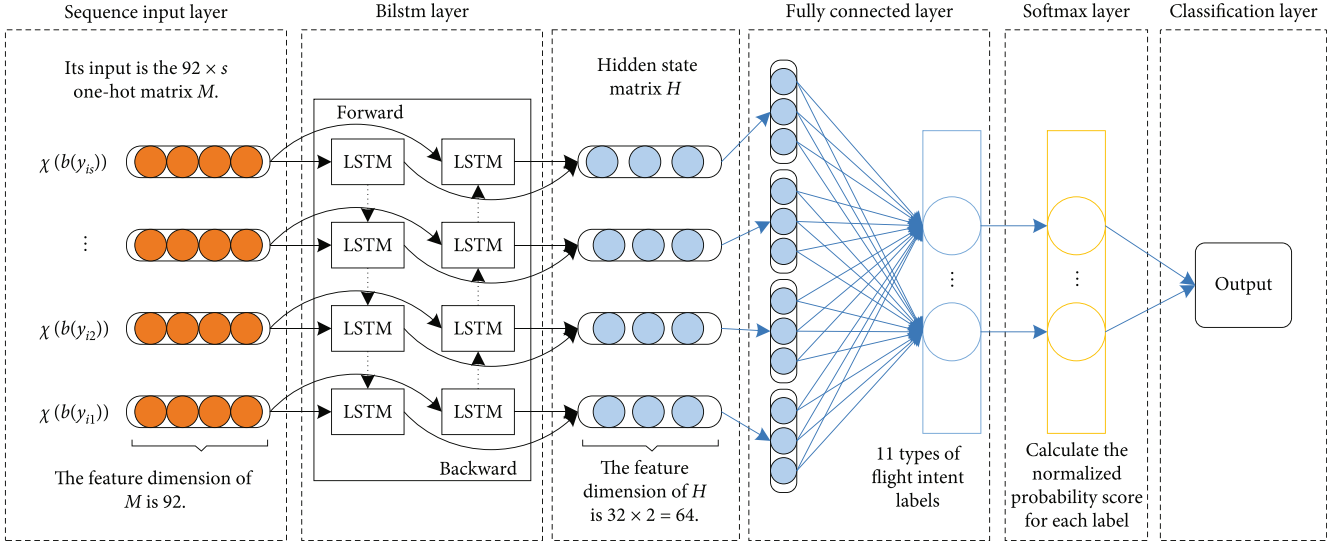


FIGURE 7: BiLSTM network architecture diagram of the single-pilot intention model.

TABLE 1: Analysis of the network architecture of the single-pilot intention model.

S/N	Name	Type	Activations	Learnables	Description
1	sequenceInputLayer (sequence input with 92 dimensions)	Sequence input	92	—	Input sequence data into the network.
2	bilstmLayer (BiLSTM with 32 hidden units)	BiLSTM	64	InputWeights 256×92 RecurrentWeights 256×32 Bias 256×1	Learn bidirectional long-term dependencies between operation sequence time steps.
3	fullyConnectedLayer	Fully connected	11	Weights 11×64 Bias 11×1	Create a fully connected layer with 11 neurons for the 11 flight intent labels.
4	softmaxLayer	Softmax	11	—	Applies the softmax function to calculate the probability scores for each flight intent label.
5	classificationLayer	Classification output	11	—	Output the classification outcome according to the probability scores.

study uses the sequence input layer, the BiLSTM layer, the fully connected layer, the softmax layer, and the classification output layer to build the BiLSTM network architecture as a classifier.

The application flow of the BiLSTM network architecture is as follows: first input the one-hot matrix M using the sequence input layer; the BiLSTM layer is then used to capture the one-hot matrix's timing properties in both directions and obtain a hidden state matrix H with bidirectional long-term dependencies between operation sequence time steps; map H to all 11 neurons in the fully connected layer, with each neuron corresponding to a flight intent label; the softmax layer applies the normalized exponential to calculate the normalized probability scores for the 11 flight intent labels; finally, the classification output layer outputs the classification outcome of the operation sequence's flight intent label. Figure 7 shows the single-pilot intention model's BiLSTM network architecture diagram, and Table 1 presents an analysis of the network architecture.

The model training algorithm is shown in Algorithm 1.

6. Case Analysis

6.1. Simulation Environment. Intel Xeon Silver 4214R processor (2.4 GHz) and 64 G of RAM serve as the experimental hardware platform. Windows 10 operating system is the experimental software platform, while the MATLAB R2022a programming language is the development environment.

6.2. Dataset of Single-Pilot Intention Modeling. A data sample is obtained by the DECO interactive sensor for every action and entered into the database. Specific time steps can effectively present the trend of the intentions underlying the pilot's driving behaviors. Therefore, for every U time step, one operation sequence sample is annotated from the acquired time series data based on the experience of experts. The following three principles that guided data annotation are discussed: (1) interviewing field experts and noting operation sequences that might trigger flight intents. (2) To determine the flight intent label of the operation sequence, the field experts select the standard operating procedures

```

Input: Dataset: dataset
Output: BiLSTM Model: bilstm_model
flight_intent_labels, operation_sequences ← load dataset
intent_dictionary ← operation_sequences
for operation_sequence in operation_sequences : do
  X ← wordEmbedding(intent_dictionary, operation_sequence)
  y ← categorical(flight_intent_label)
end for
bilstm_model ← build_bilstm(MiniBatchSize, NumHiddenUnits, layer_size)
loss, out ← build_output(bilstm_output, in_size, out_size)
solver ← AdamOptimizer(loss, LearnRate)
for epoch in epochs: do
  for MiniBatch_X, MiniBatch_y in get_MiniBatch(X, y, MiniBatchSize)
  do
    bilstm_model.run(loss, feed=input: MiniBatch_X, target: MiniBatch_y)
    bilstm_model.run(solver, feed=input: MiniBatch_X, target: MiniBatch_y)
  end for
end for
return bilstm_model

```

ALGORITHM 1: The training of the single-pilot intention model based on BiLSTM.

written by CAAC or the manufacturer. (3) Multiexpert cross-check is used to reduce the impact of errors.

To implicitly establish the mapping relationship between the single-pilot operation sequence and the single-pilot intention space, this study trains the BiLSTM-based single-pilot intention model using the interactive dataset that DECO recorded. First, the intent label of the operation sequence is calibrated using the knowledge engineering method to obtain the entire dataset. The preprocessed dataset is then input into the BiLSTM network architecture for training to obtain the mapping relationship. In real time, the sensor integrates one sequence sample from the gathered single-pilot operation items every continuous T time step, encoding the collected sequence into the trained single-pilot intention model. The outcome of the single-pilot intention recognition is finally obtained.

The dataset contains 173 operation sequences, of which 61 are training sets, 94 are test sets, and 18 are validation sets. The training set was used to fit the data features, the test set was used to assess the model performance, and the validation set was used to assess overfitting [33]. An example of the dataset is shown in Table 2.

6.3. Evaluation Indicators of Single-Pilot Intent Model Performance. It is essential to define the evaluation indicators to assess the generalization performance of this model. Precision, recall, and F1 are the three selected evaluation indicators.

$$\begin{aligned}
 \text{precision} &= \frac{TP}{TP + FP}, \\
 \text{recall} &= \frac{TP}{TP + FN}, \\
 \text{F1} &= \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}.
 \end{aligned} \tag{9}$$

TP presents the quantity of true positive examples, FP presents the quantity of false positive examples, and FN presents the quantity of false negative examples.

6.4. Hyperparameter Setting of Single-Pilot Intent Model. By controlling variables, the model's hyperparameter setting obtains the most appropriate parameter values to obtain the best effect of a single-pilot intent classification: MaxEpochs, learning rate, MiniBatchSize, NumHiddenUnits, optimization function, and loss function are the hyperparameters that must be set.

6.4.1. MaxEpochs. MaxEpochs denotes the maximum number of training epochs. The generalization performance of the model gradually improves, as the epochs increase. However, if the number of epochs is too large, it may cause an overfitting issue. Figure 8 shows the corresponding F1 at various MaxEpochs. According to the figure, the classification performance F1 gradually improves as MaxEpochs increases. F1 is relatively stable when MaxEpochs is above 25 and reaches its maximum value when MaxEpochs equals 35. Hence, the MaxEpochs is set to 35.

6.4.2. Learning Rate. The learning rate is a key parameter that affects the speed at which the training algorithm updates the weight. If the learning rate is too high, training will only produce suboptimal results or diverge; if it is too low, training time will be long. Figure 9 shows the corresponding F1 at various learning rates. According to this figure, 0.002 is the best learning rate.

6.4.3. MiniBatchSize. MiniBatchSize represents the mini-batch size. Setting MiniBatchSize determines how the sequence data will be padded and how the F1 will be impacted. Figure 10 shows the corresponding F1 at various MiniBatchSizes. Figure 10 demonstrates that eight is the best MiniBatchSize.

TABLE 2: Dataset of operation sequences and flight intent labels.

S/N	Operation sequences	Flight intent labels
1	Thrust levers to idle (y_{13}), engine fuel control switch to RUN position (y_3), start right engine (y_{10}), monitor right engine parameters (y_8), start left engine (y_6), monitor left engine parameters (y_7), flap/slat lever to detent 2 (y_4), autobrake switch to RTO (y_1), set bleed air (y_{11}), probe heat switch to on (y_9), window heat switch to high position (y_{14}), yaw damper on (y_{15}), generator switch to on (y_5), shut down APU (y_{12}), check flight control (y_2)	Engine starting on the ground
2	Thrust levers to 40% N1 and keep stable (y_{23}), thrust levers to 60% N1 and keep stable (y_{24}), press TOGA and verify setting of take-off thrust (y_{20}), perform rotation to 15 degrees (y_{21}), climb with FD command (y_{18}), landing gear up (y_{19}), flap/slat lever to detent 0 (y_{17}), autopilot on (y_{16}), adjust standard air pressure at transition altitude (y_{22})	Take-off and climb
3	Complete descent preparation on CDU (y_{29}), set DA (y_{36}), select course (y_{34}), check landing elevation (y_{27}), set autobrake (y_{35}), complete approach briefing (y_{28}), safety belt switch to on (y_{33}), adjust standard air pressure at transition altitude (y_{22}), set exterior lights (y_{37}), landing gear DN (y_{31}), flap/slat lever to landing configuration (y_{30}), set exterior lights (y_{37}), autopilot off (y_{25}), autothrottle off (y_{26}), reverse thrust on (y_{32}), thrust levers to idle (y_{13})	Descend approach and land
4	Press TOGA (y_{39}), lift flap/slat lever upward for 1 detent (y_{38}), verify thrust increase (y_{40}), climb with FD command (y_{18}), landing gear UP (y_{19})	Go-around
5	Put on an oxygen mask and set 100% (y_{47}), select mask in boom/mask switch (y_{49}), establish crew communication (y_{51}), PA announces emergency order (y_{45}), safety belt switch to on (y_{33}), descend to preselected altitude FL100 (y_{48}), establish FLC mode (y_{41}), establish HDG mode (y_{42}), set preselected heading (y_{50}), maintain proper speed (y_{43}), thrust levers to idle (y_{13}), speed brake lever as required (y_{52}), transponder 7700 (y_{53}), notify ATC (y_{44}), passenger oxygen switch to on (y_{46})	Cabin altitude high and make an emergency descent
6	PA announces emergency order (y_{45}), transponder 7700 (y_{53}), notify ATC (y_{44}), TAWS switch to off (y_{65}), ELT control switch to on (y_{56}), safety belt switch to on (y_{33}), no smoking switch to on (y_{62}), set landing data (y_{64}), cockpit pressure control switch to off (y_{54}), turn man rate knob right to stop position (y_{66}), set BARO REF (y_{63}), landing gear as required (y_{60}), landing configuration as required (y_{59}), PA announces emergency order (y_{45}), thrust levers to idle (y_{13}), engine fuel control switch to off (y_{57}), apply parking brake (y_{38}), lift extinguishing handle and turn it to the end (y_{61}), execute emergency evacuation (y_{55})	Make a forced landing
7	Engine fuel control switch to off (y_{57}), engine ignitor switch to auto (y_{69}), engine bleed switch to off (y_{68}), check bleed air pressure (y_{67}), start affected engine (y_{70}), stop starting affected engine (y_{71}), notify ATC (y_{44})	Engine fire extinguishing
8	Autothrottle off (y_{26}), engine bleed switch to off (y_{68}), thrust levers to idle (y_{13}), engine fuel control switch to run position (y_3), engine ignitor switch to auto (y_{69}), start APU (y_{79}), engine bleed switch to off (y_{68}), APU bleed switch to on (y_{74}), bleed isolation valve switch to open (y_{76}), start affected engine (y_{70}), APU bleed switch to off (y_{73}), engine bleed switch to auto (y_{77}), air-conditioning packs switch to auto (y_{78}), bleed isolation valve switch to auto (y_{75}), anti-icing air source switch to normal position (y_{81}), APU as required (y_{72}), set to TA/RA in TCAS (y_{80})	Engine starting in the air
9	Airspeed is limited below 278 kt/0.7 M (y_{84}), reset horizontal stabilizer trim disconnect 1 (y_{85}), reset horizontal stabilizer trim disconnect 2 (y_{86}), operate horizontal stabilizer trim actuator (y_{82}), autopilot as required (y_{83})	Pitch trim fault recovery
10	Disconnect ATC (y_{88}), disconnect TCAS (y_{90}), disconnect GPS (y_{89}), disconnect ACP (y_{87}), TAWS switch to OFF (y_{65}), landing gear UP (y_{19}), thrust levers to 100% N1 and keep stable (y_{91})	Aircraft hijacking
11	No input (y_{92}), no input (y_{92}), no input (y_{92}), no input (y_{92}), no input (y_{92}), no input (y_{92})	Single-pilot incapacitation

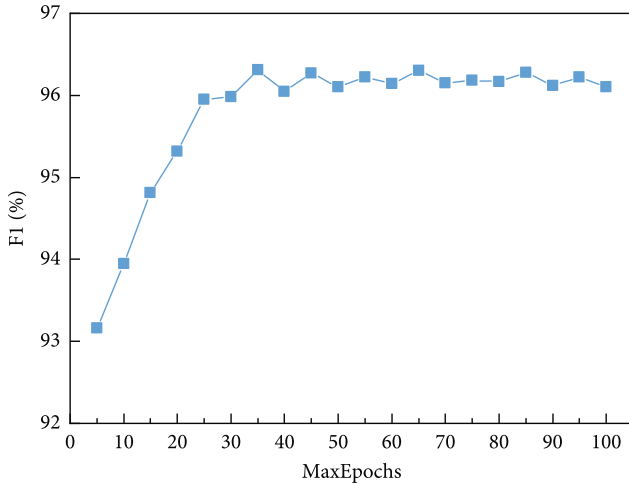


FIGURE 8: Relationship between MaxEpochs and F1.

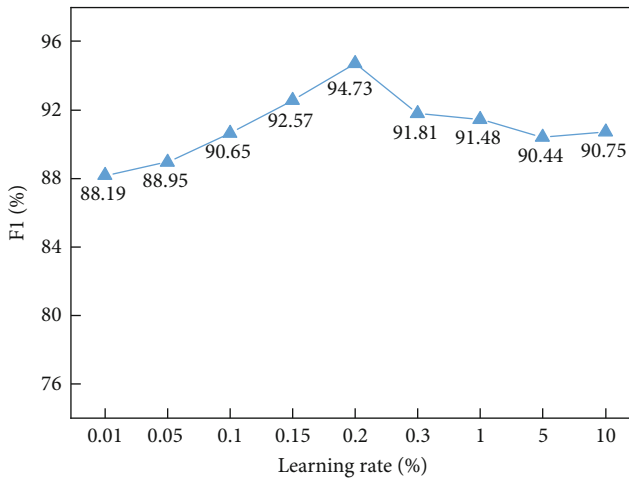


FIGURE 9: Relationship between learning rate and F1.

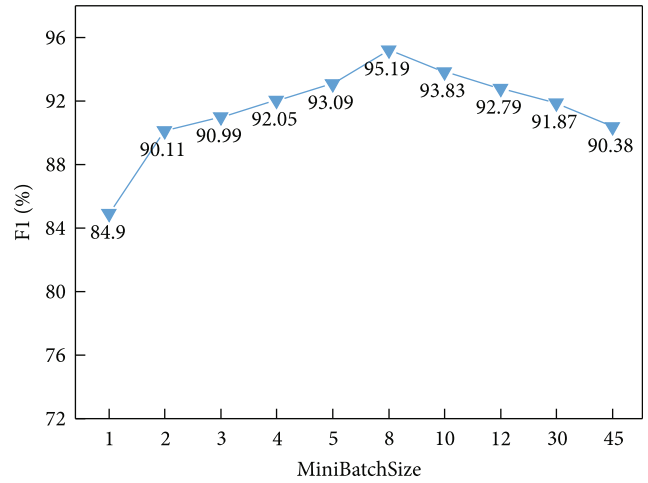


FIGURE 10: Relationship between MiniBatchSize and F1.

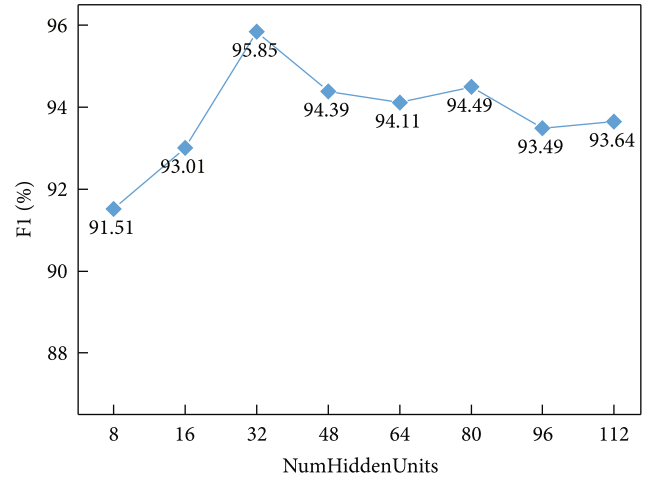


FIGURE 11: Relationship between NumHiddenUnits and F1.

6.4.4. *NumHiddenUnits*. NumHiddenUnits represent the number of hidden units. NumHiddenUnits correspond to the amount of data remembered from the sequence and directly affect the classification performance F1. If NumHiddenUnits are too small, the network’s learning ability will be impaired; if NumHiddenUnits are too large, the layer is prone to overfitting the training data. The corresponding F1 at various NumHiddenUnits is shown in Figure 11. From the figure, 32 is the ideal number of NumHiddenUnits.

Cross entropy is chosen as the loss function in this model. Cross entropy, a common loss function for assessing classification performance, lowers the risk of gradient disappearance during stochastic gradient descent [34], which is often better than the classification error rate or the mean square error. To optimize the loss function, the Adam optimizer is also selected in this model.

The optimal hyperparameter combinations are shown in Table 3.

6.5. *Simulation Results and Model Evaluation*. Generally, maneuvering-behavior changes occur in real time and are discrete during the operation of commercial aircraft. The

TABLE 3: Optimal hyperparameter combination.

S/N	Hyperparameter	Value
1	MaxEpochs	35
2	Learning rate	0.002
3	MiniBatchSize	8
4	NumHiddenUnits	32
5	Loss function	Cross entropy
6	Optimization function	Adam

one-hot encoding technique is used to preprocess the operation sequences before probabilistic classification, which is a prerequisite for single-pilot intention prediction. Figure 12 presents the visualization result of one-hot encoding for the second operation sequence listed in Table 2.

The BiLSTM-based single-pilot intention model is trained for 35 epochs, and we obtain an accuracy of 96.5% on the training sets and 94.4% on the validation sets. The training progress diagram is shown in Figure 13. According to the simulation results, the F1 score of the proposed model in this study is 95.60% on the test set. A confusion matrix

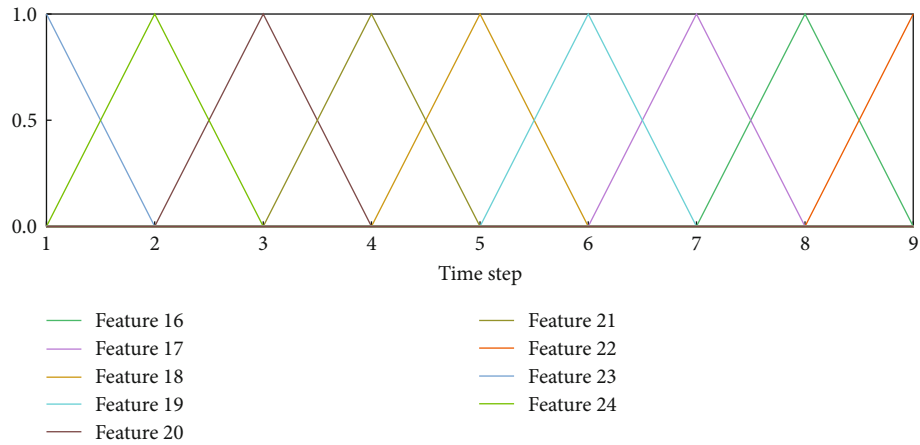


FIGURE 12: Time series visualization result.

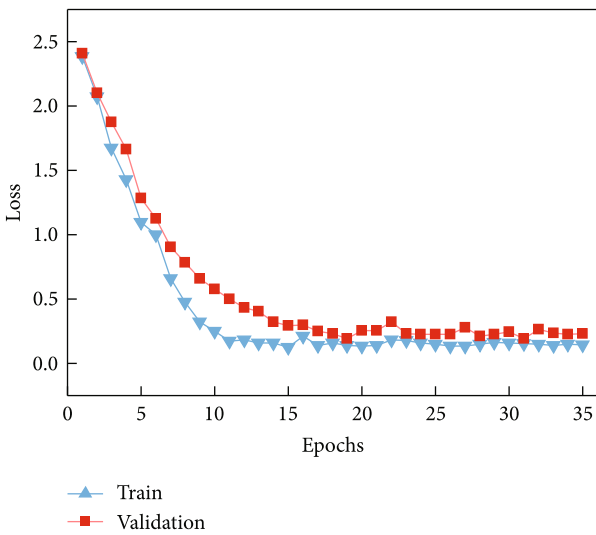
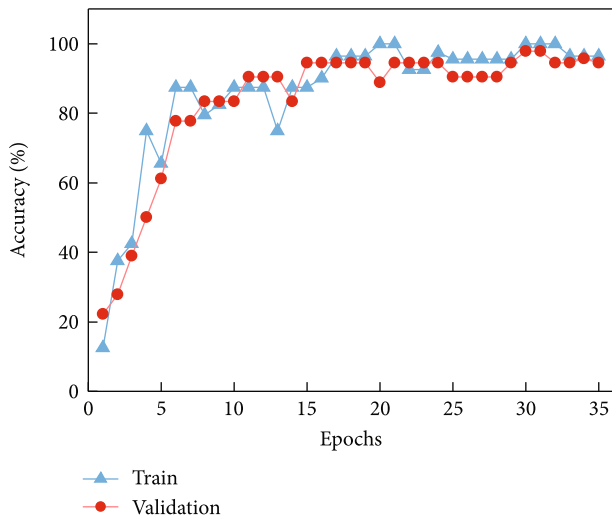


FIGURE 13: Accuracy and loss during training.

with the diagonal line representing the number of correctly identified samples was made to further observe the relationship between various intentions, as shown in Figure 14.

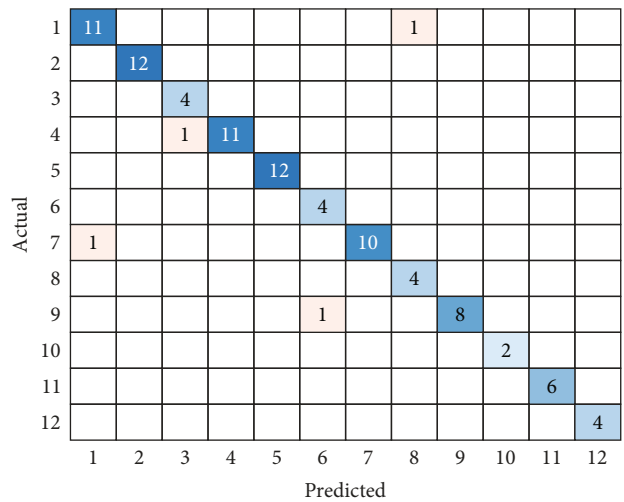


FIGURE 14: Confusion matrix of labels predicted by BiLSTM model and actual labels as marked in the data.

Based on the proposed single-pilot intent model's network architecture, deep learning training is conducted in a sequence-to-sequence manner. The flight intent labels of each time step of the interaction data are determined using the newly trained model. All inputs are required in advance to determine the flight intents at each time step because BiLSTM also uses the input from the future to the past. To avoid the real-time application issue, the interaction data are forcibly cut; specifically, each T ($T=5$) operation item sampled constitutes an entire operation sequence and is identified, which significantly reduces the waiting time for sampling subsequent operation items. For comparison, a stairstep graph is drawn, as shown in Figure 15. Calculations show that the model has a good classification effect because the classification performance F1 is 90.79%.

Figures 14 and 15 demonstrate that the model effectively expresses the data, owing to the high relevance between the current flight intent and interaction data. This corresponds with the real situation. Figure 16 shows the label meanings of the numbers in Figures 14 and 15.

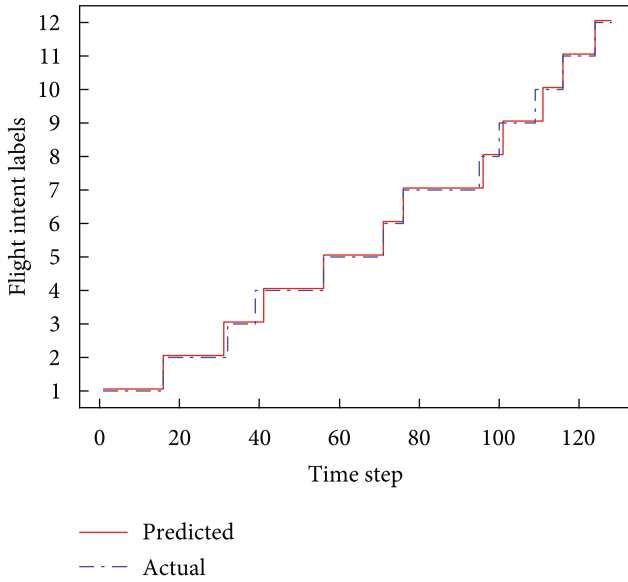


FIGURE 15: Flight intent labels detected by BiLSTM model and actual labels as marked in the data.

- 1-cabin altitude high and make an emergency descent
- 2-descend approach and land
- 3-engine fire extinguishing
- 4-engine starting in the air
- 5-engine starting on the ground
- 6-go-around
- 7-make a forced landing
- 8-pitch trim fault recovery
- 9-take-off and climb
- 10-aircraft hijacking
- 11-single pilot incapacitation
- 12-unknown intention

FIGURE 16: Label meanings of the numbers in Figures 14 and 15.

6.6. Analysis and Comparison of the Different Modeling Methods on Single-Pilot Intent. The long short-term memory (LSTM), term frequency-inverse document frequency (TF-IDF), convolutional neural network (CNN), Naive Bayesian (NB), and distributed representation intent modeling techniques were compared with the proposed modeling technique to demonstrate its superiority. The hyperparameters of LSTM are shown in Table 3. TF-IDF refers to the weighted distributed word vectors with TF-IDF, which embodies the contribution of various single-pilot operation items to the classification task [23]. The CNN technique employs five-sized three sliding convolution filters. Based on known prior probabilities, Naive Bayesian methods use Bayesian formulas to calculate posterior probabilities. The VEC refers to the distributed vector representation created by word2vec, which converts a single-pilot operation item into a 9-dimensional real vector.

Figure 17 shows the F1 scores from 10 repeated simulations using various modeling techniques. The F1 scores of each method are relatively stable in multiple simulations. The CNN, Naive Bayesian, and VEC have poor classification performance, with F1 scoring below 65%. The F1 score

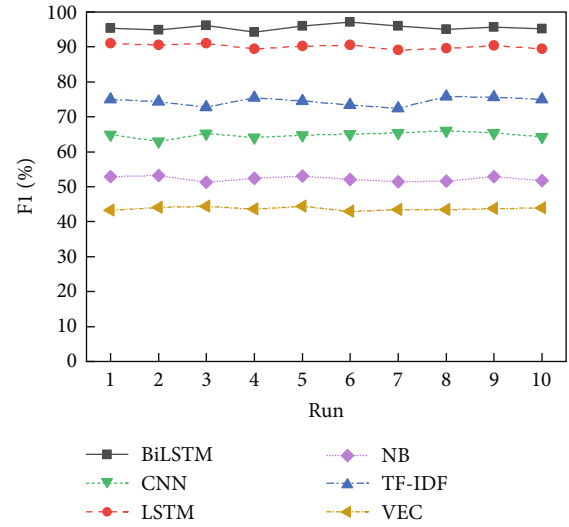


FIGURE 17: F1 scores of ten simulations for various intention modeling methods.

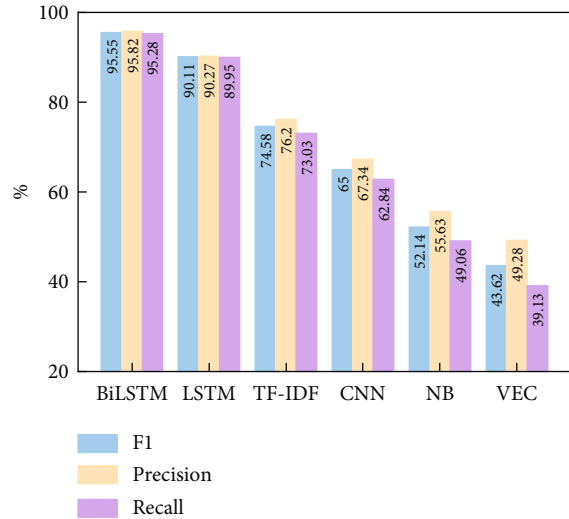


FIGURE 18: Comparison of simulation results between the proposed method and other intention modeling methods.

increases above 90%, under BiLSTM or LSTM, which is suitable for sequence modeling.

Figure 18 shows the average results of precision, recall, and F1 of 10 repeated simulations with various modeling techniques. The proposed method has a precision of 95.82%, a recall of 95.28%, and a F1 of 95.55%. Additionally, the F1 scores of BiLSTM and LSTM are superior to those of TF-IDF, Naive Bayesian, and CNN. VEC showed the worst classification performance. A specific analysis is as follows: (1) The LSTM model, which is marginally worse than the BiLSTM model, can only transmit information from front to back and cannot capture bidirectional information of the operation sequence. (2) TF-IDF performs worse than the LSTM model because it can only learn the features of term frequency and inverse document frequency in the operation sequences but has no ability to learn the time series features. (3) The one-hot matrix is a two-dimensional matrix

stacked by 0 and 1, and CNN can be used to automatically extract features from it. The classification performance of CNN is lower than LSTM because CNN can only extract local information, which cannot meet the requirements of learning presequence and postsequence association information. (4) The Bayes formula is used in Naive Bayesian to calculate the posterior probability based on the known prior probability of the research object. The method's classification performance is poor because the dataset does not adhere to the premise of the independence hypothesis. (5) Word2vec converts an operation item into a low-dimensional real vector. Although the VEC can avoid the dimensionality curse caused by the input of a high-dimensional one-hot vector, it primarily contains the semantic information of operation items and cannot contain the behavior information. Therefore, this method is completely unsuitable for modeling flight intent.

7. Discussion of Single-Pilot Intention Modeling

The proposed intention modeling concept can support a single-pilot operation. A fully adaptive approach of human-machine coordination requires robust modeling and detailed analysis of a single-pilot intention. The single-pilot cognitive status confirmation may require integration between the flight simulation system DECO and crew monitoring. This model's simulation output enables a thorough examination of single-pilot task intents. Analysis of task intents demonstrates differences in task complexity and automation assistance requirements under different scenarios. Thus, the results of intention recognition could serve as the basis for a cockpit automation system to optimize human-machine coordination and enhance cooperation based on the pilot's intention, particularly when a single pilot's behavior or intention is abnormal. However, the real scenario of single-pilot intent modeling is often more complex than the ideal one, and the impact of real-time model performance and multimodal input on system development remains to be discussed. The real-time dynamic recognition of the single pilot's intent is instructive for aviator error action prediction and calibration. The cockpit automation system must stand ready to offer additional safety recommendations with reference to risks/hazards and effective action plan. Because of the time delay of entire sequence sampling, the real-time intention perception of the BiLSTM model lags slightly behind the single-pilot action/sequence change. Nonetheless, even in certain emergency operating procedures or under difficult operating conditions, a minor delay in intention information does not typically cause severe consequences. Thus, the proposed model can still be applied to rapidly check and refresh the potential intents, thereby satisfying the real-time and effective requirements of intent recognition. Further, careful attention must be given to the techniques and measures applicable to identifying interaction. This involves assessing the crew cognitive status via multimodal input, such as touch, gesture, voice, and eye-tracking. The action of gazing can directly reflect the pilot's attention allocation and plays a vital role in optimizing the intention inference capacity. In the future, we plan to append eye-tracker data

as operation sequence input, which may contribute to a classification rate increase and will require extra development.

8. Conclusion

This study proposes a method for modeling a single pilot's intention in commercial aviation based on BiLSTM. To identify the single-pilot intentions, we design a quantity of experiments to extract the single-pilot operation sequence from the flight simulation system DECO records. More unsafe decision intentions or manipulation behaviors could be discovered through the mining and analysis of these single-pilot behavior data, which is significant for enhancing the pilot-automation partnership. Additionally, the BiLSTM model considers the time-dependent data and can determine the classification of intentions using a feedforward neural network and softmax linear variation. The above research findings expose the correlation mechanism of a single pilot's behavior and intention coupling to a limited degree. Compared with the other five single-pilot intention modeling methods, the proposed model achieves a better effect of single-pilot intention identification. However, theoretically, the flight intents can also proceed in a manner with alternating actions, which involve the single pilot interrupting one flight intent, executing another, and next resuming the previously performed flight intent. When the intent within the operation sequence migrates, it will not only affect the reasoning logic of the model itself but also cause identification confusion. These will be the upcoming significant works.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This paper was sponsored by the National Key Research and Development Program (2021YFB1600600), the Tianjin Education Commission Scientific Research Project (2022KJ058), and the Fundamental Research Funds for the Central Universities (3122022044).

References

- [1] M. Wang, G. Xiao, and G. Q. Wang, "Single pilot operation mode technology," *Acta Aeronautica et Astronautica Sinica*, vol. 41, article 323541, 2020.
- [2] Y. Chen, Y. Luo, M. Wang, K. Zhong, G. Xiao, and G. Wang, "DFCluster: an efficient algorithm to mine maximal differential biclusters for single pilot operations task synthesis safety analysis," *Chinese Journal of Aeronautics*, vol. 35, no. 5, pp. 400–418, 2022.
- [3] S. M. Neis, U. Klingauf, and J. Schiefele, "Classification and review of conceptual frameworks for commercial single pilot

- operations,” in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, pp. 1–8, London, UK, 2018.
- [4] M. L. Cummings, A. Stimpson, and M. Clamann, “Functional requirements for onboard intelligent automation in single pilot operations,” in *AIAA Infotech @ Aerospace*, p. 1652, San Diego, California, USA, 2016.
 - [5] R. O’Connor, Z. Roberts, J. Ziccardi et al., *Pre-Study Walk-through with a Commercial Pilot for a Preliminary Single Pilot Operations Experiment*, Springer, 2013.
 - [6] P. C. Schutte, “How to make the most of your human: design considerations for single pilot operations,” in *Proceedings of the International Conference on Engineering Psychology and Cognitive Ergonomics*, Los Angeles, CA, USA, 2015.
 - [7] N. G. Leveson, *Engineering a Safer World: Systems Thinking Applied to Safety*, The MIT Press, 2016.
 - [8] K. L. Mosier, L. J. Skitka, S. Heers, and M. Burdick, “Automation bias: decision making and performance in high-tech cockpits,” *Decision Making in Aviation*, vol. 8, pp. 271–288, 2017.
 - [9] D. D. Woods and N. B. Sarter, “Learning from automation surprises and going sour accidents,” in *Cognitive Engineering in the Aviation Domain*, pp. 327–353, 2000.
 - [10] J. Orasanu, L. Martin, J. Davison, and C. H. Null, “Errors in aviation decision making: bad decisions or bad luck?,” in *Proceedings of the Fourth Conference on Naturalistic Decision Making*, Warrenton, VA, 1998.
 - [11] G. Tokadli, M. Dorneich, and M. Matessa, “Evaluation of play-book delegation approach in human-autonomy teaming for single pilot operations,” *International Journal of Human-Computer Interaction*, vol. 37, no. 7, pp. 703–716, 2021.
 - [12] S. Suck and F. Fortmann, “Aircraft pilot intention recognition for advanced cockpit assistance systems,” in *Proceedings, Part II, of the 10th International Conference on Foundations of Augmented Cognition: Neuroergonomics and Operational Neuroscience*, pp. 231–240, Toronto, ON, Canada, 2016.
 - [13] D. Comerford, S. L. Brandt, J. B. Lachter et al., “NASA’s single-pilot operations technical interchange meeting: proceedings and findings,” NASA/CP—2013–216513, 2013.
 - [14] C. E. Billings, *Human-Centered Aircraft Automation: A Concept and Guidelines*, National Aeronautics and Space Administration, Ames Research Center, 1991.
 - [15] J. Cahill, T. C. Callari, F. Fortmann, D. Javaux, and A. Hasselberg, “A-PiMod: a new approach to solving human factors problems with automation,” in *13th International Conference on Engineering Psychology and Cognitive Ergonomics*, pp. 269–279, Toronto, ON, Canada, 2016.
 - [16] A. Isaac, S. T. Shorrock, R. Kennedy, B. Kirwan, H. B. Andersen, and T. Bove, “Technical review of human performance models and taxonomies of human error in ATM (HERA),” HRS/HSP-002-REP-01, 2003.
 - [17] R. E. Bailey, L. J. Kramer, K. D. Kennedy, C. L. Stephens, and T. J. Etherington, “An assessment of reduced crew and single pilot operations in commercial transport aircraft operations,” in *Proceedings of the 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*, St. Petersburg, FL, USA, 2017.
 - [18] D. Mcruer, “Human dynamics in man-machine systems,” *Automatica*, vol. 16, no. 3, pp. 237–253, 1980.
 - [19] M. Strohal and R. Onken, “Intent and error recognition as part of a knowledge-based cockpit assistant,” *Applications and Science of Computational Intelligence*, vol. 3390, pp. 287–299, 1998.
 - [20] M. Hayashi, *Hidden Markov Models for Analysis of Pilot Instrument Scanning and Attention Switching*, [Ph.D. thesis], Massachusetts Institute of Technology, 2014.
 - [21] D. C. Foyle and B. L. Hooley, *Human Performance Modeling in Aviation*, CRC press, Boca Raton, USA, 2007.
 - [22] A. Lüdtke, L. Weber, J. P. Osterloh, and B. Wortelen, “Modeling pilot and driver behavior for human error simulation,” in *International Conference on Digital Human Modeling*, pp. 403–412, San Diego, CA, USA, 2009.
 - [23] G. Xu, Y. Meng, X. Qiu, Z. Yu, and X. Wu, “Sentiment analysis of comment texts based on BiLSTM,” *IEEE Access*, vol. 7, pp. 51522–51532, 2019.
 - [24] J. Wang, B. Peng, and X. Zhang, “Using a stacked residual LSTM model for sentiment intensity prediction,” *Neurocomputing*, vol. 322, pp. 93–101, 2018.
 - [25] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
 - [26] K. S. Tai, R. Socher, and C. D. Manning, “Improved semantic representations from tree-structured long short-term memory networks,” in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp. 1556–1566, Beijing, China, 2015.
 - [27] R. Zhao, R. Yan, J. Wang, and K. Mao, “Learning to monitor machine health with convolutional bi-directional LSTM networks,” *Sensors*, vol. 17, no. 2, p. 273, 2017.
 - [28] J. Buckman, A. Roy, C. Raffel, and I. Goodfellow, “Thermometer encoding: one hot way to resist adversarial examples,” in *ICLR 2018: International Conference on Learning Representations 2018*, Vancouver, BC, Canada, 2018.
 - [29] G. Fei and B. Liu, “Breaking the closed world assumption in text classification,” in *Proceedings of the Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, San Diego, California, 2016.
 - [30] T. E. Lin and H. Xu, *Deep unknown intent detection with margin loss*, ACL, 2019.
 - [31] F. Fortmann, S. Suck, D. Javaux, J. Cahill, T. C. Callari, and A. Hasselberg, “Developing a feedback system to augment monitoring performance of aircraft pilots,” in *Proceedings of the 2016 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA)*, San Diego, CA, USA, 2016.
 - [32] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” *Advances in Neural Information Processing Systems*, vol. 26, pp. 3111–3119, 2013.
 - [33] D. Liu, Z. Zhao, and L. D. Gan, “Intention detection based on Bert-Bilstm in taskoriented dialogue system,” in *Proceedings of the 2019 16th International Computer Conference on Wavelet Active Media Technology and Information Processing*, Chengdu, China, 2019.
 - [34] G. Liu and J. Guo, “Bidirectional LSTM with attention mechanism and convolutional layer for text classification,” *Neurocomputing*, vol. 337, pp. 325–338, 2019.