aerospace

MDPI

Article

Interpreting Decision-Making Behavior in AI-Piloted Aircraft in Aerial Combat Scenarios: An Approach to Enhance Human-AI Trust

Zhouwei Lou [1,2,*], Weiyi Ge [2] and Ke Xie [2]

[1] Key Laboratory of Measurement and Control of Complex Systems of Engineering, Ministry of Education, School of Automation, Southeast University, Nanjing 210096, China

[2] National Key Laboratory of Information Systems Engineering, Nanjing 210007, China; geweiyi@cetc.com.cn (W.G.); xieke3@cetc.com.cn (K.X.)

* Correspondence: lou_zw@seu.edu.cn

Abstract

With the continuous advancement of artificial intelligence (AI) technology, AI algorithms have demonstrated exceptional aircraft control capabilities in highly dynamic and complex scenarios such as aerial combat. However, the inherent lack of explainability in AI algorithms poses a significant challenge to gaining sufficient trust, presenting potential safety risks that could lead to aircraft loss of control. This limitation hinders the widespread adoption of AI in practical applications. To enhance human–AI trust, improve system stability and safety, and advance the deployment of AI algorithms in practical settings, this study proposes an approach to describe and explain AI decision-making behaviors using natural language. Natural language is a straightforward medium for expressing information, which avoids the need for additional decoding or interpretation, particularly in rapidly changing battlefield environments, enabling pilots to quickly comprehend the intentions of AI algorithms and thereby fostering trust in AI systems. This study constructs a dataset of AI decision behavior description and interpretation based on adversarial temporal data in an aerial combat scenario and introduces an encoder–decoder framework that integrates an attentional mechanism. Findings from the experiments suggest that this approach effectively delineates and elucidates the AI decision-making behaviors, thereby facilitating mutual trust between humans and AI.

Keywords: human–AI trust; aerial combat; artificial intelligence; natural language; decision-making behavior

check for updates

Academic Editor: Gokhan Inalhan

Received: 22 June 2025
Revised: 8 August 2025
Accepted: 12 August 2025
Published: 13 August 2025

Citation: Lou, Z.; Ge, W.; Xie, K. Interpreting Decision-Making Behavior in AI-Piloted Aircraft in Aerial Combat Scenarios: An Approach to Enhance Human-AI Trust. *Aerospace* **2025**, *12*, 722. https://doi.org/10.3390/aerospace12080722

1. Introduction

With the rapid advancement of modern military technology, the battlefield environment is increasingly characterized by high dynamism and complexity [1]. This necessitates the integration of powerful artificial intelligence (AI) technologies into military operations. Aircraft, as pivotal assets on the modern battlefield, undertake critical missions such as combat and reconnaissance [2–4]. Effectively leveraging the advantages of AI algorithms in actual aerial combat is crucial for establishing battlefield superiority [5].

Currently, many researchers are actively exploring the application of AI algorithms in aerial combat scenarios to enable autonomous aircraft to perform combat missions [6]. Among these, AI algorithms represented by deep reinforcement learning (DRL) have emerged as the dominant approach due to their capabilities in autonomous learning and policy optimization [7,8]. DRL-based methods continuously refine their decision-making strategies through interaction with the environment, allowing agents to make autonomous decisions and respond adaptively in highly dynamic and adversarial aerial combat environments. However, most studies remain at the simulation stage. Experimental results in simulated environments indicate that AI algorithms have demonstrated capabilities surpassing human performance in autonomous aerial combat scenarios [9–11]. Meanwhile, a minority of researchers are testing the performance of AI algorithms in real flight conditions [12,13]. This requires a careful consideration of system safety, significant restrictions on AI algorithms, and the presence of a pilot to take control if the AI's decisions are not trusted [14]. Overall, deploying AI algorithms in actual aircraft still faces numerous challenges, with the most critical being the trust between humans and AI [15].

Currently, widely used AI algorithms typically rely on neural networks to represent and learn complex patterns and rules. However, the internal mechanisms of neural networks are highly complex and inherently difficult to interpret, making it challenging to intuitively understand how features are extracted from input data and how decisions are made [16]. Meanwhile, deploying AI technologies on real aircraft involves not only significant technical thresholds and costs but also potential safety risks. In addition to performance improvements, the safety and controllability of the overall system must be prioritized in practical applications [17]. These factors collectively hinder the widespread adoption of AI algorithms in actual combat scenarios, with the core issue being the uncertainty surrounding the correctness of AI-generated decisions. Therefore, establishing trust between humans and AI algorithms is of great significance. However, research in this area remains relatively limited.

AI algorithms can provide extensive decision support information, such as real-time battlefield data analysis [18], enemy tactics prediction [19,20], and threat assessment [21,22]. These methods primarily assist in analysis, requiring pilots to spend considerable time understanding the final decisions made via AI algorithms. However, in aerial combat scenarios where real-time decision-making is critical, pilots often lack the necessary time to comprehend the information provided through AI. Therefore, more efficient and direct interaction methods are required. Natural language serves as a direct means of information conveyance, facilitating a rapid grasp of the actions and intentions of AI algorithms. Natural language has been extensively employed across a multitude of tasks. For instance, natural language is used to explain how end-to-end AI systems control vehicles in autonomous driving technology [17,23]. Additionally, it can be applied to describe the informational content within remote sensing images in the domain of remote sensing image event identification [24–27]. In aerial combat scenarios, concise natural language expressions can effectively convey the decisions made via current AI algorithms and the reasons behind them, thereby quickly gaining the trust of pilots.

To enhance mutual trust between humans and AI, this study proposes an approach for describing and explaining AI decision-making behaviors using natural language. A highly dynamic and uncertain air combat scenario is selected as the experimental environment to better reflect the real-time demands of human–AI interaction. Multiple adversarial agents are generated using advanced DRL algorithms, yielding time-series data of various confrontational processes. These data are then combined with manually annotated texts to construct a corresponding natural language description dataset. On this basis, an attention-based encoder–decoder framework is designed. The encoder extracts critical information from the adversarial sequences, while the attention-based decoder effectively captures temporal context to enable more accurate language generation. Experimental results demonstrate that the proposed method can accurately describe and interpret the decision-making behaviors of AI algorithms, offering a promising approach to fostering mutual trust between humans and AI.

The main contributions and innovations of this study are summarized as follows:

- This work investigates the trust issues arising during the deployment of AI algorithms in air combat games—a topic that has received limited attention in existing research but holds significant practical importance.
- An approach is proposed to describe and interpret AI decision-making behaviors using natural language, offering a feasible approach for the practical application of AI algorithms.
- Validation results based on the constructed dataset demonstrate that the proposed encoder–decoder architecture, enhanced with an attention mechanism, can accurately describe and interpret the decision-making behaviors of AI algorithms.

The rest of this paper is structured as follows. Section 2 covers the preliminary knowledge regarding the sequence-to-sequence model used in this study. Section 3 details the construction and implementation of the aerial combat environment employed in our experiments. Section 4 elaborates on the proposed methodology, including the dataset construction and model architecture. Section 5 presents the main experimental results and validates the effectiveness and superiority of the proposed approach through multiple comparative experiments. Section 6 conducts a comprehensive discussion of the proposed approach. Finally, Section 7 concludes the paper.

2. Preliminaries

To facilitate the understanding of the proposed model, this section introduces the fundamental components employed in the approach. Specifically, a brief overview of the gated recurrent unit (GRU) [28] is presented, which serves as the core of the sequence encoder, followed by a description of the attention mechanism that enhances the decoder's ability to focus on salient parts of the input sequence during natural language generation.

2.1. GRU

The GRU is a type of recurrent neural network designed to manage long-term dependencies effectively. This model employs a duo of gating mechanisms—the reset gate and the update gate—to regulate the propagation of information. The reset gate functions to ascertain the proportion of the anterior hidden state's data that should be preserved or dismissed. At the same time, the update gate regulates to what extent the current hidden state's information is incorporated into the output or excluded. The resulting hidden state is a combination of the previous and candidate hidden states, adjusted by the update gate, capturing the relevant information at the present moment. This selective retention and obsolescence of information across various temporal intervals empower the GRU to effectively encapsulate extended temporal dependencies within sequences. As shown in Figure 1, the internal design is where the fundamental strength of the GRU lies.

At each time step, t, the GRU network first calculates two gate vectors, r_t and z_t, based on the input vector x_t and the previous hidden state vector, h_{t-1}, representing the reset gate and the update gate, respectively. Both gate vectors are computed via a sigmoid function, yielding values between 0 and 1 and representing each gate's degree of activation or deactivation.

The reset and update gate calculation equations are as follows.

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{1}$$

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{2}$$

where σ denotes the sigmoid function, W_r, W_z, U_r, and U_z denote the learnable weight matrix, and b_r and b_z denotes the learnable bias vector.

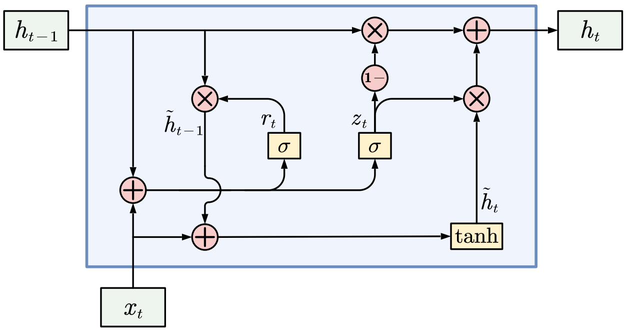


Figure 1. Structure of GRU network.

Next, the GRU network resets the hidden state vector, h_{t-1}, of the previous time step according to the reset gate vector, r_t. It obtains a new vector, $\tilde{h}_{t-1}$, representing the information retained after the reset. Then, $\tilde{h}_{t-1}$ is concatenated with the input vector x_t, and a candidate hidden state vector, $\tilde{h}_t$, is obtained through a tanh function, indicating that the current time step may need to add new information to the hidden state. The calculation formulas of the reset operation and candidate hidden state are as follows:

$$\tilde{h}_{t-1} = r_t \odot h_{t-1} \tag{3}$$

$$\tilde{h}_t = tanh(W_h x_t + U_h \tilde{h}_{t-1} + b_h) \tag{4}$$

where $\odot$ denotes the product of vector elements, W_h and U_h denote the learnable weight matrix, and b_h denotes the learnable bias vector.

The last step of the GRU network is to linearly interpolate the hidden state vector h_{t-1} and the candidate hidden state vector $\tilde{h}_t$ of the previous time step according to the update gate vector z_t, and obtain the hidden state vector h_t of the current time step, which represents the updated information. The calculation equation of the update operation is as follows:

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{5}$$

Through this internal structure, the GRU network can realize the function of long-term and short-term memory, that is, to retain or forget the past information when needed, and to add or ignore the current information.

2.2. Attention Mechanism

The attention mechanism, drawing inspiration from the human capacity for selective focus in visual and auditory processing, serves to augment the representational capacity and optimize the learning dynamics of neural architectures. Central to this mechanism is the adaptive allocation of varying degrees of significance, or attention weights, to the constituents of a given input sequence. This allocation is contingent upon the specific computational objective and the ambient context, thereby facilitating the accentuation of critical data and the concurrent downplaying of peripheral or less pertinent information.

The attention mechanism, originally utilized in computer vision for selecting image regions and detecting objects, was subsequently adopted in natural language processing to address long-range dependencies in sequence-to- sequence (Seq2Seq) models. In conventional Seq2Seq models, the encoder transforms the input sequence into a fixed-length vector, which the decoder then utilizes as its initial state to generate the output sequence. This method encounters two primary issues. First, the encoder is limited to producing a vector of constant length, impairing its capacity to capture the complete essence of the source sequence. Second, the decoder, when constructing each term, relies solely on the immediate state and the word that came before, without any direct reference to the other elements in the original sequence.

To mitigate the aforementioned challenges, the mechanism of attention presents a novel strategy. In the process of word generation through the decoder, an attentional distribution is derived, contingent upon the decoder's instantaneous state and the previously generated word. This distribution serves as an indicator of the emphasis placed on each element within the source language sequence. By leveraging this distribution, a composite of the source language words, weighted accordingly, is computed to yield a context vector. This vector is subsequently incorporated as a supplementary input into the decoder. Thus, the decoder generates each word by integrating its current state, the context vector, and the prior word. This technique allows the decoder to extract the most pertinent details from the source sequence, thereby improving the accuracy and effectiveness.

Figure 2 illustrates the structure of the attention mechanism, with s_t indicating the hidden state of the RNN at time step t, which is determined via

$$s_t = f(s_{t-1}, c_t, Y_{t-1}) \tag{6}$$

where c_t representing the contextual information, is derived from annotations $(h_1, \cdots, h_T)$ obtained via the encoder when processing the input sentence. It can be computed by

$$c_t = \sum_{j=1}^{T} a_{tj} h_j \tag{7}$$

where a_{tj} represents the weight for each annotation, h_j, and it can be calculated via

$$a_{tj} = \frac{\exp(e_{tj})}{\sum_{k=1}^{T} \exp(e_{tk})} \tag{8}$$

where $e_{tj} = a(s_{t-1}, h_j)$ represents a model for alignment, assessing the degree of correspondence between the output at the t-th instance and the input segment near index j.

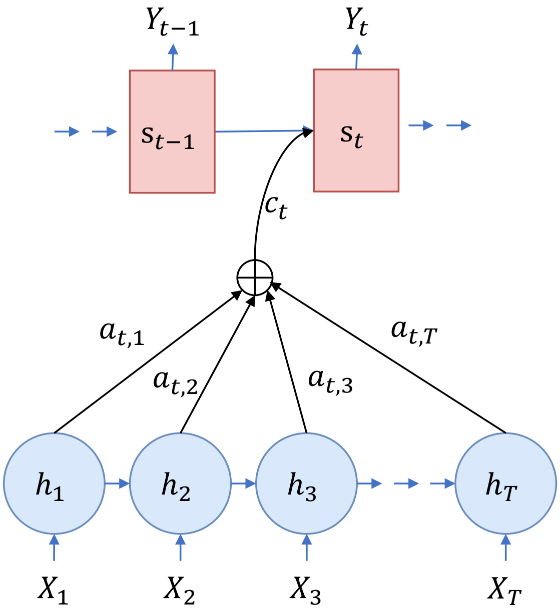


Figure 2. Structure of attention mechanism.

3. Aerial Combat Scenario Modeling

This study adopts a commonly used Dogfight scenario as the simulated aerial combat environment. In this setting, both the red and blue sides deploy a single aircraft to engage in close-range combat, with the objective of shooting down the opponent. The engagement emphasizes real-time situational awareness, action decision-making, and maneuver execution, and is characterized by high uncertainty and dynamic interactions.

Subsequent sections will provide a detailed description of the system modeling approach, the composition of states and actions, and the definition of the game objectives.

3.1. System Modeling

In order to provide a thorough depiction of the aircraft's status, the present study utilizes the terrestrial coordinate system as the fiducial framework. The aircraft's state is characterized by three-dimensional coordinates (x, y, and z), velocity (v), and the angles of pitch, yaw, and roll (γ, ψ, ϕ), as illustrated in Figure 3.

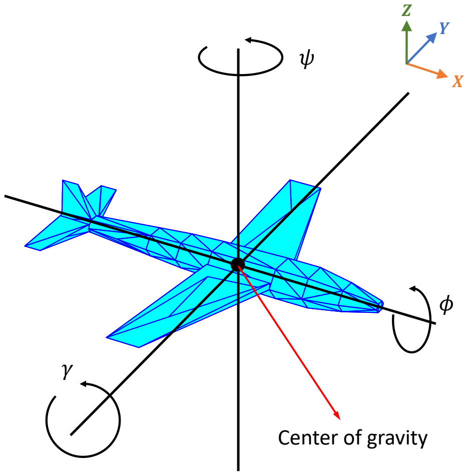


Figure 3. Schematic representation of the 3-DoF particle model for aircraft dynamics analysis.

To simplify the analysis and focus on the flight trajectories of both aircraft within a specific time frame, as well as to infer the AI algorithms' decision-making behavior and rationale, this paper omits the modeling of certain detailed moments acting on the aircraft. Instead, a simplified three-degree-of-freedom (3-DoF) point-mass model [2] of the aircraft is employed for the simulation. Utilizing this model, the temporal correlations between the aircraft's positional coordinates and velocities are established. The differential equations that govern the aircraft's movement are shown below.

$$\begin{aligned}
\frac{dv}{dt} &= g(n_x - \sin\gamma) \\
\frac{dx}{dt} &= v\cos\psi\cos\gamma \\
\frac{dy}{dt} &= v\sin\psi\cos\gamma \\
\frac{dz}{dt} &= v\sin\gamma
\end{aligned} \tag{9}$$

where g denotes gravitational acceleration, and n_x signifies the tangential overload.

3.2. Goal of the Game

The objective of both aircraft in the engagement is to achieve ultimate victory in the adversarial encounter. This study primarily focuses on the relative states of aircraft during dogfight scenarios, aiming to infer the intentions of AI algorithms and their corresponding aerial combat maneuvers. Consequently, the modeling of destructive weapons such as missiles is excluded. Instead, the outcome is determined based on metrics such as relative angles and distances between the aircraft.

To exemplify, Figure 4 depicts the positional relationship between two aircraft engaged in an offensive maneuver. The aspect angle (AA) and antenna train angle (ATA) emerge as pivotal parameters in assessing the engagement's outcome. The methodologies for computing these angles are delineated below.

$$\begin{aligned} AA &= \arccos\left(\frac{\boldsymbol{\rho}\cdot\boldsymbol{V_o}}{|\boldsymbol{\rho}||\boldsymbol{V_o}|}\right) \\ ATA &= \arccos\left(\frac{\boldsymbol{\rho}\cdot\boldsymbol{V_a}}{|\boldsymbol{\rho}||\boldsymbol{V_a}|}\right) \end{aligned} \tag{10}$$

The AA between the aggressor and target aircraft indicates the hardship of evasion, with a smaller AA indicating greater difficulty. A smaller AA indicates greater difficulty in evasion. Conversely, the ATA between the target and the aggressor aircraft indicates the precision of the attacker's aim, with a smaller ATA indicating higher accuracy. When the distance, $|\rho|$, between the two aircraft falls below a certain threshold, the target aircraft is considered to be within the attacker's engagement range. To gain a tactical edge, the attacking aircraft must meet these criteria: $|ATA| < 30^\circ$ & $|AA| < 60^\circ$ & $|\rho| < 1$ km. Maintaining this posture for an extended duration can classify the attacking aircraft as the victor in the engagement.

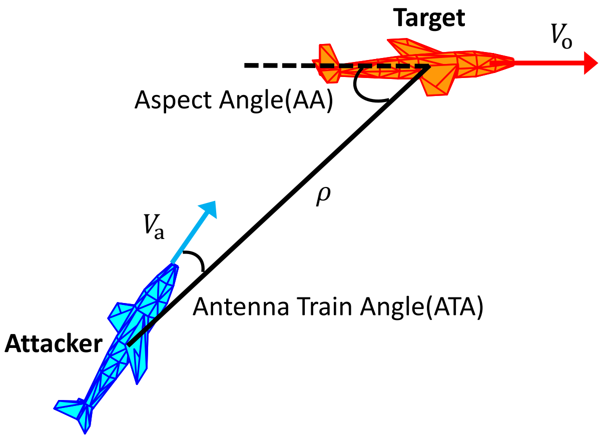


Figure 4. Spatial relationship between two aircraft during an attack.

3.3. State Definition

In this environment, the state at any given time encompasses all relevant information regarding both aircraft, and it is expressed as follows.

$$S = [x_a, x_o, y_a, y_o, z_a, z_o, v_a, v_o, \gamma_a, \gamma_o, \psi_a, \psi_o] \tag{11}$$

3.4. Action Definition

In the dogfight game scenario employed in this study, three consecutive action commands are used to operate the aircraft. The action space of the environment is depicted as follows.

$$\mathcal{A} = [\psi_{cmd}, \gamma_{cmd}, n_{xcmd}] \rightarrow [-1, 1] \tag{12}$$

where ψ_{cmd} changes the yaw angle, ψ, γ_{cmd} changes the aircraft's pitch angle, γ, and n_{xcmd} controls the speed, v.

To enhance the alignment between the aircraft model's control processes in the simulation and those of the actual aircraft, a first-order inertia component is incorporated into the three control directives. The corresponding relationship can be delineated as follows.

$$\begin{aligned} \frac{\psi}{\psi_{cmd}} &= \frac{1}{1+\tau_{\psi}s} \\ \frac{\gamma}{\gamma_{cmd}} &= \frac{1}{1+\tau_{\gamma}s} \\ \frac{n_x}{n_{xcmd}} &= \frac{1}{1+\tau_{x}s} \end{aligned} \tag{13}$$

where τ_{ψ}, τ_{γ}, τ_x are the time constants for ψ, γ, and n_x, respectively.

4. Method

4.1. Overview

This study aims to describe and interpret the decision-making behavior of current AI algorithms using natural language. Specifically, the objective is to generate a complete natural language sentence based on a sequence of confrontation data over a period of time. The confrontation sequence used in this study consists of flight data over T consecutive time steps, with each time step represented by an N-dimensional feature vector encompassing various state information of both red and blue aircraft, as outlined in Equation (11). As both the input and output of this task are sequences, it can be regarded as a typical Seq2Seq problem, with a structure similar to that of machine translation.

Typically, the Seq2Seq problem is addressed using an encoder–decoder architecture [29], which consists of an encoder and a decoder. The encoder is responsible for extracting information from the input sequence and generating a context vector. The decoder then generates the next output word based on this context vector, the previously generated word, and its own hidden state. Figure 5 illustrates the overall framework of the proposed task. The input sequence is first extracted from the confrontation data and then processed via the encoder–decoder architecture to generate a complete natural language sentence that describes and explains the decision-making behavior of the AI algorithm.

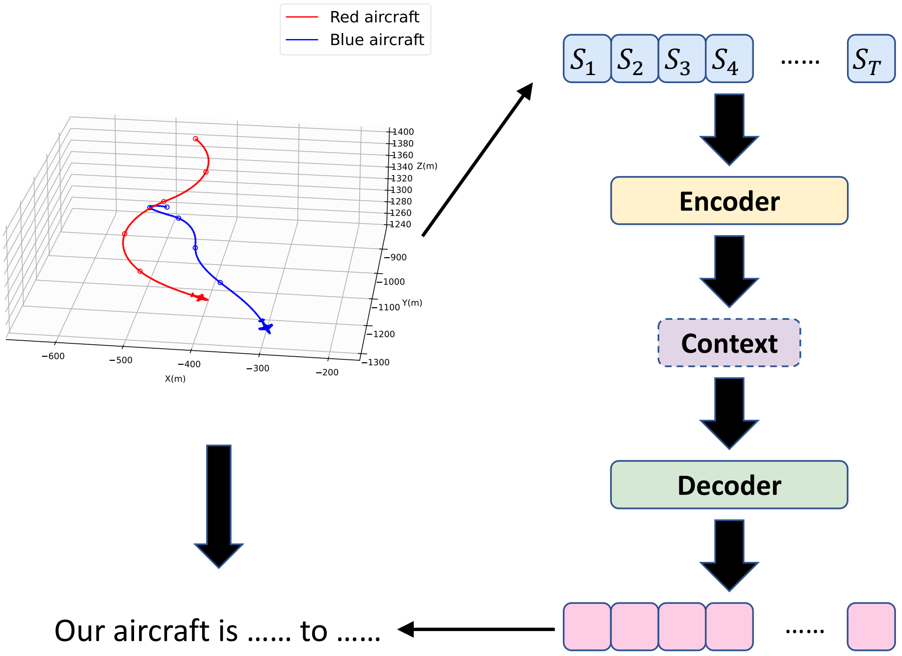


Figure 5. Overview of the issue.

4.2. Database Establishment

Given the current lack of publicly available dogfight databases, it is necessary to construct the required experimental dataset based on the dogfight environment described in Section 3.

4.2.1. Learning Dogfight Adversarial Agents

To enable an approach to describing and interpreting current AI decision-making behaviors using natural language, it is essential to construct a high-quality adversarial game dataset that encompasses diverse combat scenarios. This, in turn, requires the development of high-performance game-playing agents capable of generating the underlying adversarial data. Given the widespread adoption and strong performance potential of DRL algorithms in training intelligent agents [7,8], this study adopts the soft actor–critic (SAC) algorithm [30] for subsequent agent training. Following the scheme proposed in [2], the design of the reward function takes into account critical combat-related factors, including the relative velocity and distance between the agent and its opponent, as well as angular metrics such as AA and ATA. This design aims to guide the agent toward advantageous tactical positions while minimizing exposure to potential threats, thereby enhancing its decision-making capability in dynamic aerial combat environments.

The training of agents is divided into two stages. In the first stage, agents are trained in a one-on-one manner. Initially, multiple dogfight algorithms with diverse styles are designed based on an expert system and used as opponents to train the initial agent. Subsequently, the trained agents serve as new opponents for the further iterative training of the next generation of agents. During this process, the agents reinforce themselves by exploiting the weaknesses of their training opponents, resulting in high win rates against them. However, when facing unfamiliar opponents, their performance tends to decline markedly, revealing a lack of adaptability to more complex adversarial scenarios. Therefore, agents trained solely through one-on-one encounters in the first stage exhibit an insufficient generalization capability. To enhance overall performance, it is necessary to expose the agents to more diverse adversarial experiences [31], thereby improving their comprehensive combat effectiveness.

In the second stage of training, the agent is confronted with multiple opponents. Agents obtained from the first stage are first evaluated, and those demonstrating strong performance are selected as opponents for this phase. During training, an opponent is randomly selected from the existing pool of trained agents for each independent episode, and the interaction data are stored in a shared experience buffer. This approach allows the experience buffer to accumulate diverse combat experiences against opponents with varying strategies, thereby enabling the new agent to learn more generalized policies. Newly trained agents that exhibit strong performance are incorporated into the agent pool to participate in subsequent iterations of training.

After two stages of training, a set of high-performing agents was obtained. These agents possess independent and effective combat strategies, enabling the generation of diverse aerial combat trajectory sequences. However, since their strategies are learned through DRL algorithms, the decision-making process lacks interpretability and is difficult for humans to comprehend. Therefore, it is essential to develop methods that can enhance human–AI trust.

4.2.2. Construction of the Raw Dataset

During the confrontation, the duration of engagement between the two sides may range from a few seconds to several minutes. Since this study primarily focuses on the natural language description and explanation of AI decision-making behavior at the current moment, only the most recent engagement sequence is considered. Specifically, a 5 s segment of the engagement sequence is selected for analysis.

To construct the dataset, two agents are randomly selected from the agent pool to engage in an aerial dogfight, thereby generating a complete combat sequence. Figure 6 illustrates the full course of a representative engagement, where the red and blue curves represent the flight trajectories of the two opposing aircraft throughout the encounter. The movement directions of both sides at each time step can be used to infer their respective strategies. A continuous 5 s segment is randomly selected from the full trajectory for dataset construction. As shown in Figure 6, an example is provided in which the final 5 s are extracted, with the corresponding trajectories highlighted in darker colors, while the remaining trajectories are shown in lighter shades.

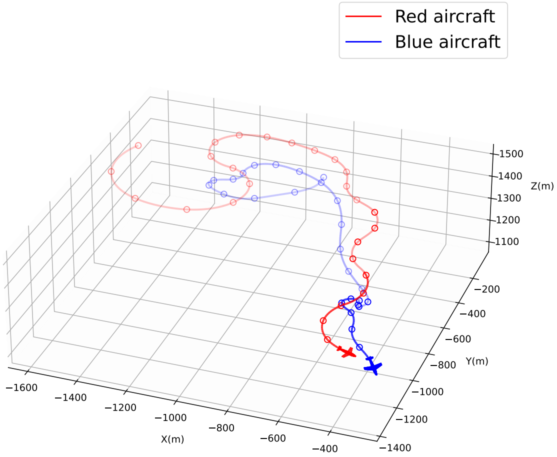


Figure 6. Example of flight trajectories of both agents during a dogfight. Dots along the trajectories indicate 1 s time intervals.

Figure 7 illustrates a 5 s sequence extracted from the example shown in Figure 6, where features such as position, velocity, and attitude angles are represented as time-varying curves. These data capture the dynamic states of both aircraft within the 5 s interval, forming the initial input format. After preprocessing, they are used as input to the encoder.

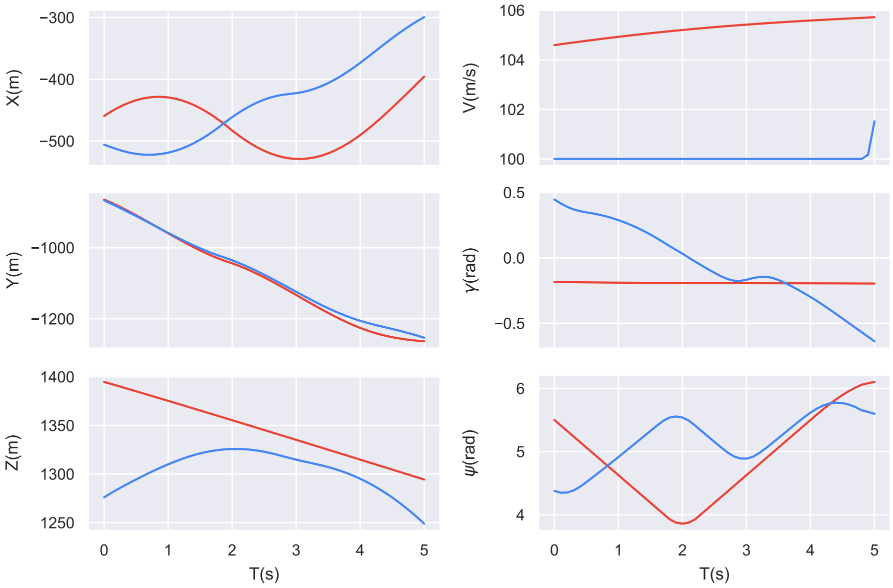


Figure 7. Extracted 5 s sequence data from the entire engagement. The red curve represents the red aircraft, and the blue curve represents the blue aircraft.

Through the above procedure, the raw data can be constructed. Subsequently, an expert system is employed to annotate the sequence data with natural language descriptions, thereby generating the corresponding data labels and building a complete dataset.

4.2.3. Dataset Augmentation

Given the challenges associated with dataset annotation, it is essential to make full use of the available data resources. When describing and interpreting the current decision-making behavior of AI using natural language, it is particularly important to learn and attend to the spatial relative positions between the red and blue aircraft in the adversarial data. In the original dogfight environment, differences in initial positions and combat strategies between the two sides lead to significant variations in their spatial distributions. Therefore, to enable the model to better capture the relative, rather than the absolute positional information of the red and blue aircraft, two data augmentation strategies are proposed based on the characteristics of the dogfight environment.

(1) Rotating the entire trajectory around the Z-axis. This method preserves the physical consistency of the trajectory data and does not alter the relative positions of the red and blue aircraft. Assuming there exists a three-dimensional coordinate point $P = (x, y, z)$ with a yaw angle ψ. When this point is rotated around the Z-axis by an angle θ, the resulting coordinates, P', and the updated yaw angle, ψ', can be computed using the following formulas.

$$\begin{aligned} x' &= x\cos\theta - y\sin\theta \\ y' &= x\sin\theta + y\cos\theta \\ z' &= z \\ \psi' &= \psi + \theta \end{aligned} \tag{14}$$

(2) Translating the entire trajectory. The overall trajectory can be translated along the x, y, and z directions. It should be noted that the z-axis data in the trajectory represents the altitude of the aircraft; therefore, the range of variation in this direction should be properly constrained during translation.

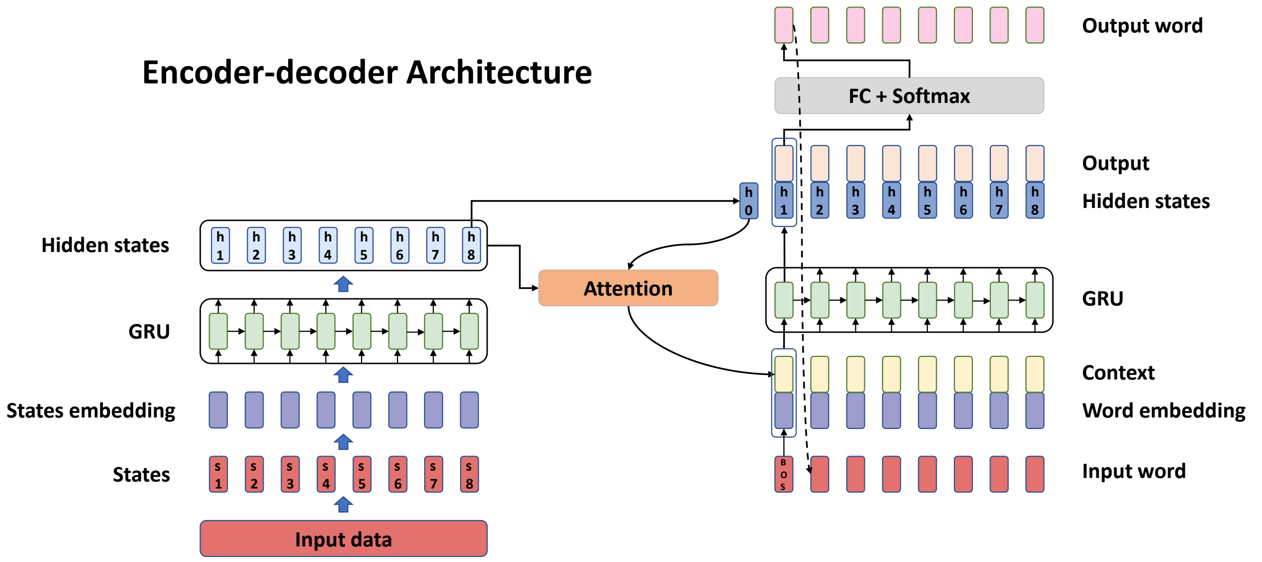
After data augmentation, a total of 10,500 samples were obtained. Each sample contains the state information of both the red and blue sides over a duration of 5 s, recorded at 0.1 s intervals. Among them, 80% were used for the training set, 10% for the validation set, and the remaining 10% for testing the model's performance.

4.3. Structure of the Model

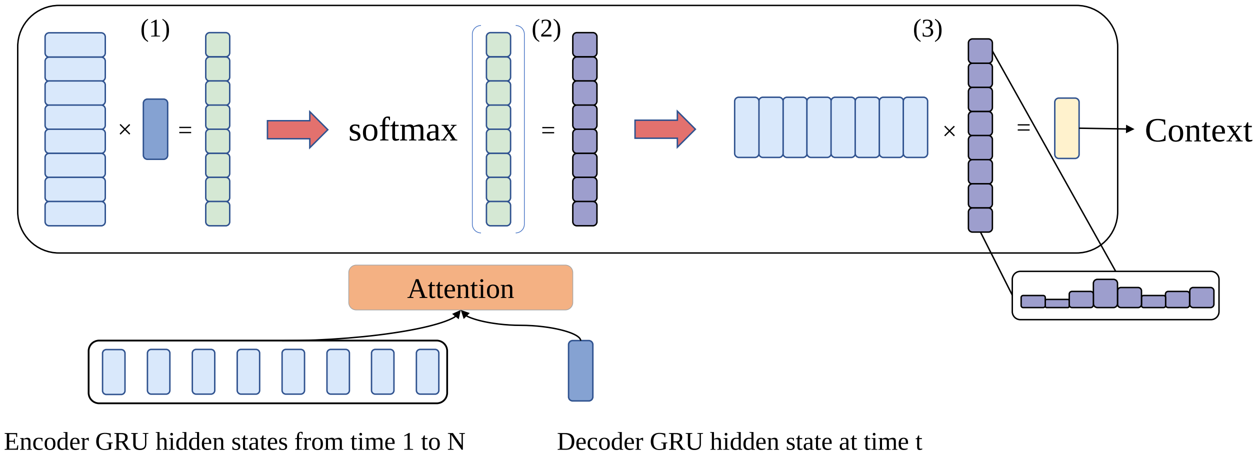
The model architecture is illustrated in Figure 8. Figure 8a presents the proposed encoder–decoder framework, with the encoder on the left and the decoder on the right. The decoder section in the figure details the process of generating the first word. Figure 8b demonstrates the computation flow of the attention mechanism. The following sections will provide a detailed explanation of the implementation and functionality of these two components.

4.3.1. Encoder

The primary function of the encoder is to extract information from the input sequence and transform it into hidden state vectors for use by the decoder, thereby completing the decoding task. The GRU network is adopted as the core structure of the encoder due to its ability to effectively capture semantic dependencies in long sequences, as well as its simplicity and computational efficiency.



(**a**)


(1)
×
=
softmax
(2)
=
(3)
×
=
Context
Attention
Encoder GRU hidden states from time 1 to N
Decoder GRU hidden state at time t

(**b**)

Figure 8. Structure of the model. (**a**) Schematic of the encoder–decoder architecture. (**b**) Computation flow of the attention mechanism.

This study uses a sequence of combat states as input to the encoder, with T time steps, and each step containing N state variables. As illustrated in Figure 8a, the left part depicts the structure of the encoder. Initially, the input state at each time step is mapped to a fixed-length vector. Subsequently, the GRU network continuously extracts temporal information, ultimately yielding a $T \times C$-dimensional hidden state vector. Here, C denotes the output feature layer dimension. The hidden state vector, which encapsulates the information extracted from the input sequence, will be utilized via the decoder.

4.3.2. Attention Decoder

In this study, an attention mechanism is employed to enhance the decoder's performance. This mechanism enables the decoder to concentrate on the key aspects of the encoder's output as it constructs each word. As a result, the improved decoder can utilize context from the full sequence instead of relying solely on the last encoder output, leading to better output accuracy and quality.

The decoder's core structure remains a GRU. The operation proceeds as follows: initially, the decoder's hidden state is initialized (usually with the last hidden state of the encoder), and a start token $< BOS >$ is input to mark the beginning of the sentence. Following this, the initial hidden state and the encoder's hidden state vectors are used to compute the attention mechanism's context vector, as depicted in Figure 8b. The context vector is combined with the embedded vector of the input word and fed into the GRU network to generate both the updated hidden state and the output vector. The updated hidden state is subsequently employed to predict the next words, with the output vector being processed through a fully connected layer and softmax activation, producing a vector with a dimension of M, where M denotes the number of words in the vocabulary. This vector indicates the probability distribution of each word in the vocabulary at the current position. The word with the highest probability is selected as the current output and used as the input for the next step. This iterative process continues until the end-of-sentence token $< EOS >$ is encountered, thus completing the sentence generation.

5. Experiment

This section provides the experimental evaluation of the proposed model. The first part introduces the evaluation metrics for performance assessment. The second part presents the main experimental results to demonstrate the effectiveness of the approach. The third part reports ablation studies conducted to further investigate the model.

5.1. Evaluation Metrics

In this study, our model produces descriptive narratives in natural language for AI's decision-making actions, utilizing adversarial dogfight data sequences as the foundation. To evaluate the accuracy of the generated sentences, appropriate evaluation metrics need to be selected.

In natural language processing evaluation, frequently employed metrics are Bilingual Evaluation Understudy (BLEU) [32], Consensus-Based Image Description Evaluation (CIDEr) [33], Recall Oriented Understudy for Gisting Evaluation-Longest Common Subsequence (ROUGE-L) [34], and Metric for Evaluation of Translation with Explicit Ordering (METEOR) [35]. Our research utilizes BLEU as one of the evaluation criteria.

The BLEU score is calculated using the following equation.

$$BLEU = BP \times \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{15}$$

where w_n is the weights for each n-gram's precision. In the field of natural language processing, an n-gram refers to a sequence consisting of n consecutive words. p_n represents the precision of n-grams, defined as

$$p_n = \frac{\text{matched n-grams}}{\text{candidate n-grams}} \tag{16}$$

BP represents the brevity penalty, defined as

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{1-r/c} & \text{if } c \leq r \end{cases} \tag{17}$$

where c is the output sentence length, and r is the reference sentence's effective length.

This study sets $N = 4$ and uses uniform weights $w_n = 1/N$.

Furthermore, due to the brevity and fixed format of the sentences designed in this study, accuracy is directly employed as the evaluation metric. A test sample is considered accurate when its output sentence matches the correct label exactly. During the evaluation process, padding positions are ignored.

5.2. Main Results

The proposed model was trained and evaluated on the experimental dataset. Figure 9 presents a selection of the experimental results. The red trajectory represents the flight path of our aircraft, while the blue trajectory indicates the flight path of the opponent aircraft. As illustrated in the figure, the produced sentences provide an accurate depiction of the AI's decision-making behavior and the motivations behind it.

With Figure 9a taken as an example, at the current moment, our aircraft holds a relative advantage. It is positioned at the enemy aircraft's side-rear, thus possessing a positional advantage, and it has been pursuing the enemy continuously for the past five seconds. By seizing the right timing to further adjust its angle, it can achieve successful targeting and engagement of the enemy. In Figure 9b, both sides are currently in a state of equilibrium, each seeking an opportunity to initiate an attack. At this stage, our aircraft is turning right to find an advantageous attack angle. Figure 9c depicts our aircraft swiftly transitioning from a balanced position within 5 s, successfully aligning its sights on the enemy. Figure 9d shows our aircraft performing a rapid upward maneuver to gain altitude advantage in the upcoming confrontation. In Figure 9e, after being pursued by the enemy aircraft for an extended period, our aircraft executes a swift downward maneuver to evade further pursuit. Finally, in Figure 9f, following the completion of turning maneuvers by both aircraft, a head-on engagement is imminent; at this point, our aircraft turns right to avoid the enemy's targeting.

5.3. Ablation Study

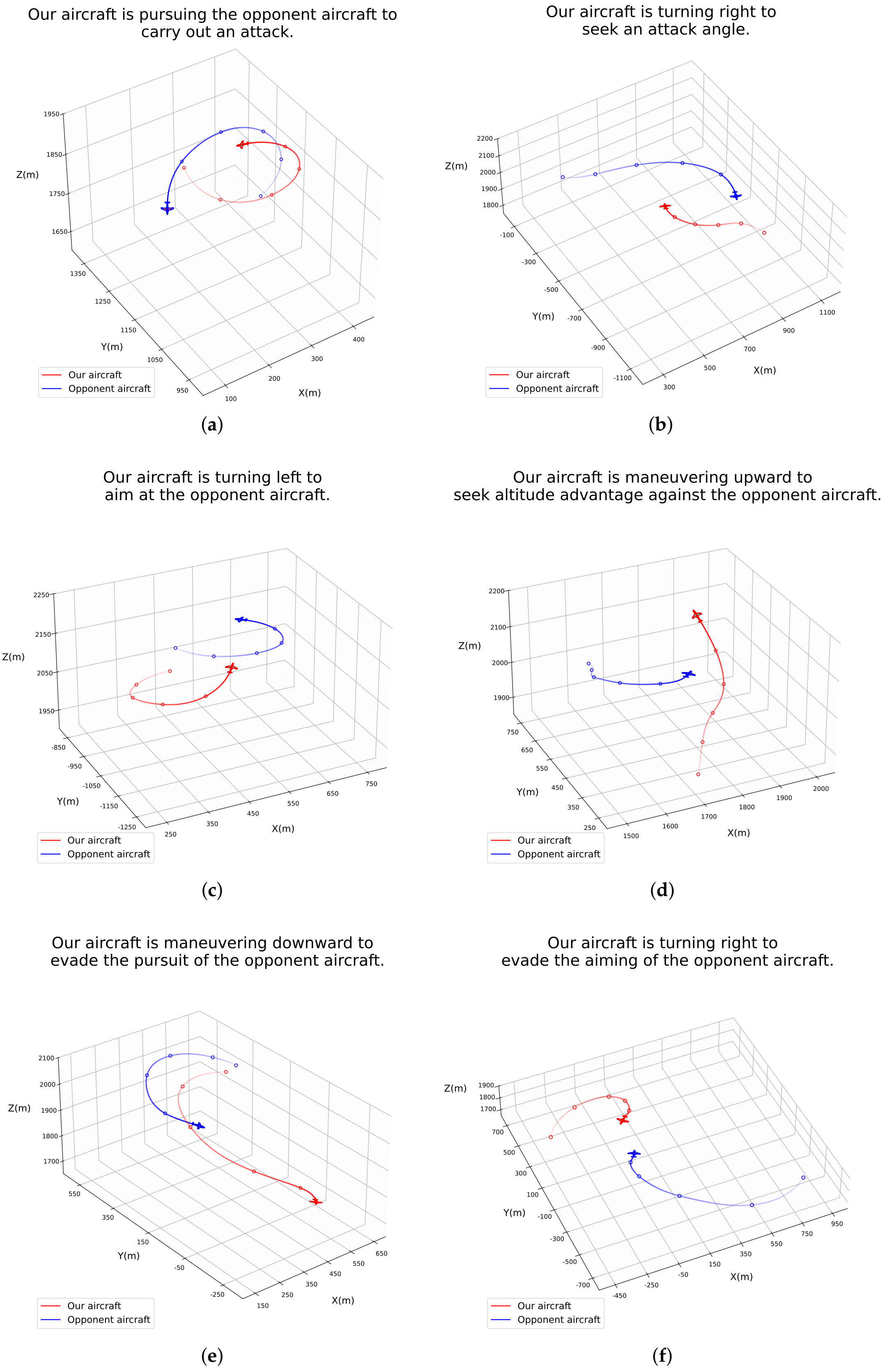
5.3.1. Comparative Evaluation

To objectively evaluate the performance of the proposed model, three different models were designed for comparison. These models consist of one employing both an encoder and a conventional decoder, one using only an attention-based decoder, and one using only a conventional decoder. For the two approaches that utilize only a decoder, the sequential data are directly input into the decoder. Each algorithm was trained five times using different random seeds. Figure 10 shows the loss curves for the different algorithms, and Table 1 presents the accuracy and BLEU4 scores for the various algorithms. The results show that, performance-wise, our proposed algorithm outperforms the other compared algorithms.

Table 1. Performance comparison of different algorithms. ± symbol signifies one standard deviation based on a series of five repetitions.

Algorithm	Accuracy (%)	BLEU4
Encoder + Attention Decoder	96.84 ± 2.05	0.9836 ± 0.0162
Encoder + Decoder	89.68 ± 6.02	0.9380 ± 0.0362
Attention Decoder only	80.28 ± 6.52	0.8762 ± 0.0391
Decoder only	75.32 ± 4.78	0.8424 ± 0.0290

By analyzing Table 1, it is evident that algorithms using an encoder–decoder architecture perform significantly better than those using only a decoder. This suggests that the application of an encoder effectively captures temporal adversarial data, thereby assisting the decoder in achieving more precise task completion. Furthermore, compared to algorithms employing conventional decoders, those adopting attention-based decoders also exhibit significant performance improvements, with accuracy gains of 8% and 6.6%, respectively, for the encoder–decoder and standalone decoder approaches.



Our aircraft is pursuing the opponent aircraft to carry out an attack.

(**a**)

Our aircraft is turning right to seek an attack angle.

(**b**)

Our aircraft is turning left to aim at the opponent aircraft.

(**c**)

Our aircraft is maneuvering upward to seek altitude advantage against the opponent aircraft.

(**d**)

Our aircraft is maneuvering downward to evade the pursuit of the opponent aircraft.

(**e**)

Our aircraft is turning right to evade the aiming of the opponent aircraft.

(**f**)

Figure 9. Results display. The sentence above each 3D scene graph is generated by the model.

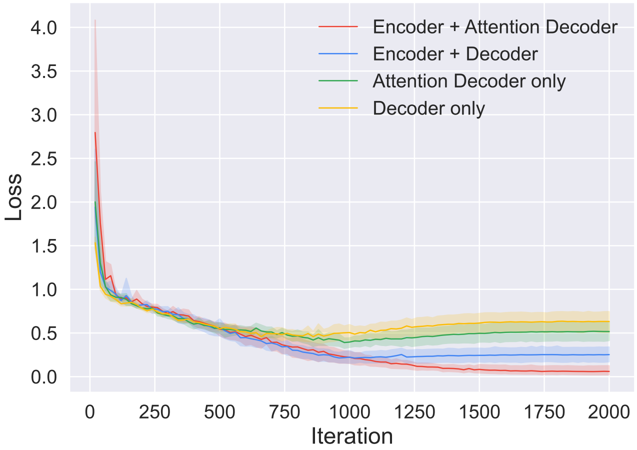


Figure 10. Loss curves for different algorithms. The solid lines indicate the average values, and the shaded areas depict the 95% confidence interval derived from five runs.

5.3.2. Impact of Data Augmentation Techniques

Due to the limited amount of original combat data and the high cost of annotation, it is crucial to make full use of the annotated data available. As discussed in Section 4.2.3, two data augmentation techniques—rotation and translation—are employed. These methods do not alter the relative spatial positions of the aircraft, thus preserving the integrity of the annotations. By enriching the spatial distribution of flight trajectories, these augmentation methods enable the model to learn the relative spatial relationships between the two aircraft, rather than focusing on their absolute positions.

Four different strategies were employed to construct the training and testing datasets: one using both rotation and translation as data augmentation techniques, one using only rotation, one using only translation, and one based on the original dataset without any augmentation. The total amount of data was kept consistent across all four strategies. For performance comparison, the testing dataset generated using both rotation and translation was uniformly adopted. Table 2 presents the performance results of the model trained on these four datasets.

Table 2. Performance comparison of different data augmentation techniques. ± symbol signifies one standard deviation based on a series of five repetitions.

Method	Accuracy(%)	BLEU4
Rotation + Translation	96.84 ± 2.05	0.9836 ± 0.0162
Rotation	72.32 ± 6.34	0.8396 ± 0.0422
Translation	38.00 ± 1.94	0.5775 ± 0.0194
Original	34.40 ± 4.80	0.5396 ± 0.0380

An analysis of Table 2 reveals that the model achieves the best performance when trained on the dataset augmented with both rotation and translation. Although these methods do not alter the relative spatial positions of the aircraft, training on the original dataset fails to accurately capture this information, resulting in an accuracy of only 34.4% on the test dataset. In contrast, the dataset augmented with rotation significantly improves accuracy, as rotation substantially diversifies the data, enriching the training set and enabling the model to learn more precise information. Overall, the same dataset can be transformed into various forms in space, and augmenting the training data with diverse yet relevant information can significantly enhance the model's learning effectiveness.

6. Discussion

With the continuous advancement of AI technology, the integration of AI algorithms in various applications is increasing. In the field of aircraft automatic control, especially in highly dynamic and complex aerial combat environments, there is a pressing need to enhance combat effectiveness through the introduction of AI algorithms. In this process, the issue of trust between humans and AI algorithms cannot be ignored. To address this issue, this paper proposes an approach that utilizes natural language to describe and explain AI decision-making behaviors. Experimental results demonstrate that the proposed approach can accurately describe and explain current AI decision-making behaviors, providing an effective solution for promoting the practical application of AI algorithms.

Previous studies have shown that using AI algorithms in aerial combat scenarios can achieve significant results, confirming that AI algorithms possess capabilities that surpass human performance in such contexts. These findings suggest that AI algorithms have the potential to be practically applied in aircraft to enhance military combat effectiveness in the future. However, applying AI algorithms to actual combat aircraft still faces numerous challenges, such as technical barriers and cost issues. In reality, if humans lack sufficient trust in AI algorithms, the performance of these algorithms will be difficult to fully realize. Currently, research on this issue is still limited, and this paper has aimed to draw academic attention to the issue of human–AI trust.

Although this paper has made some progress in addressing the issue of human–AI trust, several limitations remain. First, the proposed solution has not yet been validated in real-world scenarios. Future work needs to test and evaluate its practical feasibility in more complex environments. Second, the choice of sequence segment length significantly impacts natural language descriptions, requiring a balance based on specific circumstances to achieve optimal descriptive effects. Finally, this study requires more data and better annotations to support further improvements. Learning from more comprehensive data can further enhance the overall performance of the model. When constructing datasets, it is essential to consider the inclusion of agents with varying capabilities and ensure comprehensive coverage of the relative states of both aircraft. Despite these limitations, this study provides valuable insights and explorations into solving the issue of human–AI trust.

7. Conclusions

To advance the application of AI algorithms in aerial combat scenarios, it is imperative to address the issue of mutual trust between humans and AI. This study investigates this challenge in depth and proposes an approach for describing and explaining AI decision-making behavior through natural language. The proposed approach employs an encoder–decoder framework enhanced with an integrated attention mechanism, enabling the generation of natural language descriptions based on a sequence of combat data over a period of time. The approach is systematically evaluated in highly dynamic and uncertain aerial combat environments. The experimental results demonstrate that the proposed model achieves an accuracy of 96.84% and a BLEU4 score of 0.9836 on the test set, effectively describing and explaining AI decision-making behaviors. This offers a promising solution to enhance the trustworthiness of AI algorithms in real-world operational contexts.

Author Contributions: Conceptualization, Z.L.; methodology, Z.L. and K.X.; software, Z.L.; validation, W.G.; formal analysis, K.X.; investigation, Z.L.; data curation, Z.L.; writing—original draft preparation, Z.L.; writing—review and editing, W.G. and K.X.; visualization, Z.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by National Key Laboratory of Information Systems Engineering, No: 05202304.

Data Availability Statement: The data supporting the study findings can be provided upon request to the corresponding author.

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

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
DRL	Deep Reinforcement Learning
GRU	Gated Recurrent Unit
Seq2Seq	Sequence to Sequence
3-DoF	Three-Degree-of-Freedom
AA	Aspect Angle
ATA	Antenna Train Angle
BLEU	Bilingual Evaluation Understudy
CIDEr	Consensus-Based Image Description Evaluation
ROUGE-L	Recall Oriented Understudy for Gisting Evaluation-Longest Common Subsequence
METEOR	Metric for Evaluation of Translation with Explicit Ordering

References

1. Wang, Y.; Wang, J.; Fan, S.; Wang, Y. Quick intention identification of an enemy aerial target through information classification processing. *Aerosp. Sci. Technol.* **2023**, *132*, 108005. [CrossRef]
2. Kong, W.; Zhou, D.; Yang, Z.; Zhao, Y.; Zhang, K. UAV autonomous aerial combat maneuver strategy generation with observation error based on state-adversarial deep deterministic policy gradient and inverse reinforcement learning. *Electronics* **2020**, *9*, 1121. [CrossRef]
3. Zhang, Y.; Huang, F.; Deng, X.; Li, M.; Jiang, W. Air target intention recognition and causal effect analysis combining uncertainty information reasoning and potential outcome framework. *Chin. J. Aeronaut.* **2024**, *37*, 287–299. [CrossRef]
4. Yang, Z.; Sun, Z.X.; Piao, H.Y.; Huang, J.C.; Zhou, D.Y.; Ren, Z. Online hierarchical recognition method for target tactical intention in beyond-visual-range air combat. *Def. Technol.* **2022**, *18*, 1349–1361. [CrossRef]
5. Fang, F.; Fei, A.; He, J.; Li, Q.; Chen, J.; Feng, D. Development and consideration of key technologies for intelligent aircombat command and control in future. *Command. Inf. Syst. Technol.* **2024**, *15*, 10–18. [CrossRef]
6. Pope, A.P.; Ide, J.S.; Mićović, D.; Diaz, H.; Rosenbluth, D.; Ritholtz, L.; Twedt, J.C.; Walker, T.T.; Alcedo, K.; Javorsek, D. Hierarchical Reinforcement Learning for Air-to-Air Combat. In Proceedings of the 2021 International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, 15–18 June 2021; pp. 275–284. [CrossRef]
7. Jing, X.; Cong, F.; Huang, J.; Tian, C.; Su, Z. Autonomous Maneuvering Decision-Making Algorithm for Unmanned Aerial Vehicles Based on Node Clustering and Deep Deterministic Policy Gradient. *Aerospace* **2024**, *11*, 1055 . [CrossRef]
8. Chen, C.; Song, T.; Mo, L.; Lv, M.; Lin, D. Autonomous Dogfight Decision-Making for Air Combat Based on Reinforcement Learning with Automatic Opponent Sampling. *Aerospace* **2025**, *12*, 265. [CrossRef]
9. Sun, L.; Qiu, H.; Wang, Y.; Yan, C. Autonomous UAV maneuvering decisions by refining opponent strategies. *IEEE Trans. Aerosp. Electron. Syst.* **2024**, *60*, 3454–3467. [CrossRef]
10. Xu, J.; Zhang, J.; Yang, L.; Liu, C. Autonomous decision-making for dogfights based on a tactical pursuit point approach. *Aerosp. Sci. Technol.* **2022**, *129*, 107857. [CrossRef]
11. Hu, D.; Yang, R.; Zhang, Y.; Yue, L.; Yan, M.; Zuo, J.; Zhao, X. Aerial combat maneuvering policy learning based on confrontation demonstrations and dynamic quality replay. *Appl. Artif. Intell.* **2022**, *111*, 104767. [CrossRef]
12. Cotting, M.C.; Stephens, S.S.; Cole, J.; Barricklow, J.; Gray, W. X-62 VISTA Capabilities and Architecture. In Proceedings of the AIAA SCITECH 2023 Forum, National Harbor, MD, USA, 23–27 January 2023; p. 1744. [CrossRef]
13. Cotting, M.C.; Stephens, S.S.; Cole, J.; Gray, W.; Hartwig, L.; Caraway, D.J.; Harris, Q.; Schiavone, A.; Babala, P.; Picard, R. X-62 VISTA Simulation and Autonomy Flight Testing. In Proceedings of the AIAA SCITECH 2023 Forum, National Harbor, MD, USA, 23–27 January 2023; p. 1928. [CrossRef]
14. Schiavone, A.; Babala, P.; Hartwig, L.; Cotting, M.C. X-62 VISTA System for the Autonomous Control of the Simulation Design and Implementation. In Proceedings of the AIAA SCITECH 2023 Forum, National Harbor, MD, USA, 23–27 January 2023; p. 1926. [CrossRef]
15. Zhang, G.; Chong, L.; Kotovsky, K.; Cagan, J. Trust in an AI versus a Human teammate: The effects of teammate identity and performance on Human-AI cooperation. *Comput. Hum. Behav.* **2023**, *139*, 107536. [CrossRef]

16. Shwartz-Ziv, R.; Tishby, N. Opening the black box of deep neural networks via information. *arXiv* **2017**, arXiv:1703.00810. [CrossRef]
17. Jin, B.; Liu, X.; Zheng, Y.; Li, P.; Zhao, H.; Zhang, T.; Zheng, Y.; Zhou, G.; Liu, J. Adapt: Action-aware driving caption transformer. In Proceedings of the 2023 IEEE International Conference on Robotics and Automation (ICRA), London, UK, 29 May–2 June 2023; pp. 7554–7561. [CrossRef]
18. Wang, M.; Wang, L.; Yue, T.; Liu, H. Influence of unmanned combat aerial vehicle agility on short-range aerial combat effectiveness. *Aerosp. Sci. Technol.* **2020**, *96*, 105534. [CrossRef]
19. Zhang, A.; Zhang, B.; Bi, W.; Mao, Z. Attention based trajectory prediction method under the air combat environment. *Appl. Intell.* **2022**, *52*, 17341–17355. [CrossRef]
20. Sun, Y.; Wang, D.; Wang, W.; Xiong, L.; Yang, X. Confrontational flight trajectory prediction based on attention mechanism. In Proceedings of the 2020 International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE), Bangkok, Thailand, 30 October–1 November 2020; pp. 211–214. [CrossRef]
21. Yu, X.; Wei, S.; Fang, Y.; Sheng, J.; Zhang, L. Low-altitude Slow Small Target Threat Assessment Algorithm by Exploiting Sequential Multi-Feature with Long-Short-Term-Memory. *IEEE Sens. J.* **2023**, *23*, 21524–21533. [CrossRef]
22. Chen, C.; Quan, W.; Shao, Z. Aerial Target Threat Assessment Based on Gated Recurrent Unit and Self-Attention Mechanism. *Syst. Eng. Electron.* **2024**, *35*, 361–373. [CrossRef]
23. Kim, J.; Rohrbach, A.; Darrell, T.; Canny, J.; Akata, Z. Textual explanations for self-driving vehicles. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 563–578. [CrossRef]
24. Kumar, S.C.; Hemalatha, M.; Narayan, S.B.; Nandhini, P. Region driven remote sensing image captioning. *Procedia Comput Sci.* **2019**, *165*, 32–40. [CrossRef]
25. Bashmal, L.; Bazi, Y.; Al Rahhal, M.M.; Zuair, M.; Melgani, F. CapERA: Captioning Events in Aerial Videos. *Remote Sens.* **2023**, *15*, 2139. [CrossRef]
26. Hoxha, G.; Melgani, F. Remote sensing image captioning with SVM-based decoding. In Proceedings of the IGARSS 2020—2020 IEEE International Geoscience and Remote Sensing Symposium, Waikoloa, HI, USA, 26 September–2 October 2020; pp. 6734–6737. [CrossRef]
27. Bang, S.; Kim, H. Context-based information generation for managing UAV-acquired data using image captioning. *Autom. Constr.* **2020**, *112*, 103116. [CrossRef]
28. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv* **2014**, arXiv:1412.3555. [CrossRef]
29. Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* **2014**, arXiv:1406.1078. [CrossRef]
30. Haarnoja, T.; Zhou, A.; Abbeel, P.; Levine, S. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In Proceedings of the International Conference on Machine Learning, PMLR, Stockholm, Sweden, 10–15 July 2018; Volume 80, pp. 1861–1870. [CrossRef]
31. Lou, Z.; Wang, Y.; Shan, S.; Zhang, K.; Wei, H. Balanced prioritized experience replay in off-policy reinforcement learning. *Neural Comput. Appl.* **2024**, *36*, 15721–15737. [CrossRef]
32. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. BLEU: a method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA, USA, 7–12 July 2002; pp. 311–318. [CrossRef]
33. Vedantam, R.; Zitnick, C.L.; Parikh, D. CIDEr: Consensus-based image description evaluation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 4566–4575. [CrossRef]
34. Lin, C.Y.; Och, F.J. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04), Barcelona, Spain, 21–26 July 2004; pp. 605–612. [CrossRef]
35. Banerjee, S.; Lavie, A. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, Ann Arbor, MI, USA, 29–30 June 2005; pp. 65–72. [CrossRef]

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