YatRL SYSU CSE 2025-1 Homework4

Liar's Bar: Reinforcement Through Deception

Course Number：DCS245

Student's Name：傅祉珏

Student's Number：21307210

Advisor's Name / Title：Prof. Chen Xu

Date Due：18 January, 2026

14 January, 2026

HOMEWORK4: Liar's Bar: Reinforcement Through Deception

傅祉珏　21307210

Sun Yat-sen University, School of Computer Science and Engineering

Abstract

In the realm of imperfect information games, devising optimal strategies within survival games characterized by high stochasticity and irreversible elimination mechanisms presents a formidable research challenge. This paper focuses on the recently popular strategy game Liar's Bar, investigating how multiple agents balance the payoff of deception against the probability of survival under the lethal constraint of "Russian Roulette". Addressing the game's characteristics of long-horizon dependencies and non-stationary environments, we formalize it as a Partially Observable Stochastic Game (POSG) and propose a reinforcement learning framework that integrates a Transformer encoder with Proximal Policy Optimization (PPO). To capture opponents' historical behavioral patterns and break the strategy homogenization often seen in self-play training, we innovatively introduce a reward engineering mechanism based on "Personality Shaping". This allows for the pre-training of baseline agents with heterogeneous traits, such as aggressive bluffing, aggressive challenging, and conservative survival. Through a two-stage curriculum learning strategy, we train an unbiased Rational Agent to undergo adversarial evolution within this heterogeneous mixed-game environment. Experimental results demonstrate that the architecture incorporating attention mechanisms effectively extracts game context information. Furthermore, the Rational Agent, through dynamic Bayesian risk assessment, significantly outperforms single-personality opponents in win rate, doubt success rate, and survival rate. These findings substantiate that in survival games, a dynamic risk management strategy based on informational advantage is superior to purely aggressive gaming or conservative defense. The complete source code is available at `https://github.com/Billiefu/YatRL.git`.

Key words: Multi-Agent Reinforcement Learning; Imperfect Information Games; Liar's Bar; Transformer; Personality Shaping; Game Theory

1 Introduction

In the intersection of artificial intelligence and game theory, imperfect information games have long served as the quintessential testing ground for evaluating agent decision-making capabilities. While algorithms such as Libratus and DeepStack have achieved superhuman performance in zero-sum games like Texas Hold'em, existing research predominantly focuses on profit-maximization models based on chips or scores, often overlooking survival game scenarios characterized by "irreversible elimination mechanisms". This paper focuses on Liar's Bar, a strategy game that has recently gained global popularity by creatively integrating traditional card-bluffing mechanics with the high-stakes "Russian Roulette". Unlike traditional imperfect information games, the Nash Equilibrium in Liar's Bar is not solely determined by hand probabilities and opponent history but is dynamically constrained by the state of the revolver's cylinder. This unique mechanism imbues the

gameplay with high stochasticity and intense psychological warfare, requiring agents to not only master the arts of deception and counter-deception but also to strike a delicate balance between expected payoff and survival probability.

This gaming environment, fused with extreme risk penalties, presents novel challenges for Multi-Agent Reinforcement Learning (MARL). In Liar's Bar, the environment is non-stationary, as the strategies of participants drift drastically as the probability of a live round in the chamber increases. Furthermore, the game history exhibits long-term temporal dependencies, where an opponent's pattern of play from several rounds prior often serves as a critical cue for detecting current bluffs. Traditional policy networks based on simple fully connected layers struggle to capture these complex temporal features and often lack robustness when facing heterogeneous strategies, such as aggressive or conservative opponents. The core scientific problem this paper aims to address is how to evolve an optimal strategy in an environment that demands both aggressive deception to gain a hand advantage and extreme conservatism to avoid elimination.

To address these challenges, this paper proposes a reinforcement learning framework combining the Transformer architecture with Proximal Policy Optimization (PPO) to explore optimal gaming strategies in Liar's Bar. We first formalize the game as a Partially Observable Stochastic Game (POSG) and design a composite state space incorporating hand information, table state, and lethality probability. To effectively analyze opponent behavior under imperfect information, we introduce a Transformer Encoder within the agent architecture to extract gaming patterns from historical action sequences. Moreover, to overcome the limitation of singular strategy convergence in self-play, we innovatively introduce a reward engineering mechanism based on "Personality Shaping". By pre-training agents with distinct characteristics—such as aggressive bluffing, aggressive challenging, and conservative survival—we construct a heterogeneous mixed-game environment. Through adversarial training of a Rational Agent within this environment, we observe the emergence of advanced strategies that surpass human hard-coded logic.

The main contributions of this paper are summarized as follows:

- **Game Formalization**: We provide the first mathematical formalization of Liar's Bar as a POSG problem, explicitly defining a unique state space and reward function that incorporates survival probabilities.
- **Transformer-based MARL Architecture**: We propose an Actor-Critic architecture capable of effectively processing long-horizon game history, demonstrating the critical role of temporal memory in counter-deception tasks.
- **Personality-based Strategy Evolution**: Through experiments, we quantitatively analyze the doubt rates and survival rates of agents with different personalities, demonstrating that the Rational Agent achieves significant performance advantages in multi-agent melees through risk management strategies.

2 Related Work

The integration of Deep Reinforcement Learning (DRL) and Game Theory provides a solid theoretical foundation and algorithmic support for solving complex decision-making problems. As summarized by Dong et al. [1] and Zhang et al. [5], methods based on Policy Gradients and

Value Functions have become the core of modern DRL, while Zhao [6, 7] further elucidates the mathematical principles regarding convergence guarantees of reinforcement learning in handling stochastic processes and optimization objectives. Concurrently, Game Theory offers an analytical framework for strategic interactions among multiple agents; Xia [4] points out that in environments with information asymmetry, the design of strategic interaction and incentive mechanisms is key to understanding Nash Equilibrium. These foundational theoretical works provide the fundamental methodological support for modeling Liar's Bar as a Partially Observable Stochastic Game (POSG) and solving it using Proximal Policy Optimization (PPO) in this paper.

Significant progress has been made in academia regarding large-scale, imperfect-information complex games. In their analysis of AlphaStar, Arulkumaran et al. [8] highlighted the importance of population-based training from an evolutionary computation perspective for solving non-transitive game strategies. Similarly, Raiman et al. [12], through their research on OpenAI Five, demonstrated the critical role of long-term planning and situational awareness in multi-agent cooperation and competition. These milestone works indicate that in environments with vast state spaces and limited observations, employing complex neural network architectures (such as LSTMs or Transformers) to capture historical information is essential for policy robustness.

Focusing specifically on "Bluffing" games, early research predominantly centered on Liar's Dice. Johnson et al. [9] evaluated the application of Dynamic Programming and Game Theory in this game, while Lee et al. [10] proposed the WiLDCARD framework, demonstrating how to utilize calculative agents and reinforced decision-making to win in lying games. Regarding Liar's Bar, the specific subject of this study, Li and Miao [11] recently compared the performance of Counterfactual Regret Minimization (CFR) and Neural Fictitious Self-Play (NFSP) algorithms within this environment, exploring basic game equilibrium. However, existing academic works mostly focus on traditional zero-sum game equilibrium solutions, relatively lacking in-depth exploration of the unique "Russian Roulette" survival mechanism inherent in this game.

It is worth noting that beyond academic exploration, the popularity of Liar's Bar in streaming media and pop culture has also provided unique perspectives for AI strategy research. Lin Yi [2, 3], through a series of "AI Wars in Liar's Bar" experimental videos, intuitively demonstrated the performance of AIs driven by different algorithms in international servers. In particular, the battles between AIs with pre-set distinct "personalities" revealed the immense impact of psychological warfare and risk appetite in actual gameplay. This idea of personality-based heterogeneous gaming aligns with the concept of population diversity proposed by Arulkumaran et al. [8]. Inspired by this, and distinct from the equilibrium-focused approach of Li L's research [11], this paper focuses more on how to evolve adaptive strategies capable of risk management in an environment containing extreme death penalties through personality-based reward shaping.

3 Game Theory

3.1 Game Formalization

To systematically analyze the complex interaction mechanisms within Liar's Bar, we rigorously formalize the game as a Multi-Agent Partially Observable Stochastic Game (POSG). This game system is defined by an 8-tuple $\mathcal{G} = \langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \Omega, \mathcal{O}, \gamma \rangle$. Here, $\mathcal{N} = \{1, ..., n\}$ represents the

set of players, where $n = 4$ in a standard match. The state space $\mathcal{S}$ is a high-dimensional joint space that includes not only the private hand information of all players $H = \{h_1, ..., h_n\}$ and public table information (such as the current target card type c_{target} and the history of actions $\mathcal{H}_{hist}$), but also a latent variable—the state of the revolver's cylinder $C_{gun} \in \{0, 1\}^6$. The peculiarity of the cylinder state lies in the fact that, although its initial configuration (e.g., one bullet placed in one of six chambers) follows a uniform distribution, the state undergoes deterministic displacement or reset as the game progresses and triggers are pulled, constituting a source of fatal stochasticity in the game.

Due to the nature of imperfect information, an observation function $\Omega : \mathcal{S} \times \mathcal{N} \to \mathcal{O}$ is introduced. For any player i, the observation o_i contains only their private hand h_i and the public history, masking the hands of other players h_{-i} and the true state of the current chamber. This information asymmetry compels agents to construct a Belief State b_i, which is an estimated probability distribution of the global state s_t given the observation history $o_{i,0:t}$. The action space $\mathcal{A} = \mathcal{A}_{doubt} \times \mathcal{A}_{play}$ exhibits a hierarchical structure: a player must first decide whether to "Doubt" the previous player's action; if choosing not to doubt, they must then select a subset of their current hand to "Play". This hybrid action space results in a large branching factor for the strategy tree, increasing the computational complexity of solving for the optimal policy.

The state transition function $\mathcal{P}(s'|s, a)$ presents a unique "dual stochasticity" in this game. The first layer of stochasticity arises from standard card dealing and reshuffling; the second layer is the core of Liar's Bar—the "Russian Roulette" mechanism. When a doubt occurs, the player judged to have lost must trigger a cylinder event. Let the current chamber position be k; if $C_{gun}[k] = 1$, the state instantly transitions to an Absorbing State, meaning the player is eliminated. This mechanism causes the number of participants $|\mathcal{N}|$ to monotonically decrease over time, rendering the game process a non-stationary survival process. Therefore, the design of the reward function $\mathcal{R}$ includes not only the positive reward r_{win} for winning a single round but also a significant negative penalty r_{death} to characterize the value of survival. In this setting, the discount factor γ reflects not only the time preference for future rewards but is also mathematically equivalent to the probability weight of surviving to the next round.

3.2 Payoff Matrix & Bayesian Equilibrium

In classical game theory models, payoff matrices are typically static. However, in Liar's Bar, we must introduce the concept of "Dynamic Risk Weighting". Consider a simplified two-player subgame: a Defender chooses a playing strategy σ_{def} (Truth or Bluff), and a Challenger chooses a response strategy σ_{chal} (Pass or Doubt). The expected payoff matrix here is not a fixed constant but a function of the current cylinder state. Let the probability of a live round under the current hammer be λ_t. This probability grows non-linearly as the number of empty clicks increases (e.g., rising from 1/6 to 1/5, and eventually to 1/1). For the Challenger, the expected utility of initiating a doubt, U_{doubt}, depends not only on the prior probability of the Defender bluffing but must also subtract the cost of death risk weighted by λ_t:

$$U_{doubt}(s_t) = P(Bluff|o_t) \cdot \mathcal{V}_{success} + (1 - P(Bluff|o_t)) \cdot [\lambda_t \cdot \mathcal{V}_{death} + (1 - \lambda_t) \cdot \mathcal{V}_{survive}] \quad (1)$$

The formula above clearly indicates that as λ_t rises, even if the Challenger has high confidence

$P(Bluff|o_t)$ that the Defender is lying, the expected utility of doubting may turn negative due to the immense penalty $\mathcal{V}_{death}$.

Given that players cannot know their opponents' true hands with certainty, this game fundamentally falls into the category of Bayesian Games. Rational agents must utilize Bayes' rule to update their posterior belief $b_i(h_{-i}|\mathcal{H}_{hist})$ regarding the opponents' hand distribution based on historical behaviors (e.g., number of cards played, speed of play, past tendencies to doubt). A refined Bayesian Nash Equilibrium (BNE) strategy π^* requires each player to formulate a Best Response to the opponents' strategies π_{-i} under the given belief system. In the context of Liar's Bar, this implies that the equilibrium strategy exhibits distinct "threshold characteristics": when survival pressure is low (low λ_t), the equilibrium strategy tends towards aggressive bluffing and frequent doubting to gain information and chips; conversely, when survival pressure approaches a critical point ($\lambda_t \to 1$), the equilibrium strategy rapidly collapses into extreme Risk Aversion—playing honestly whenever possible and avoiding doubt unless absolutely certain the opponent is lying. This "Equilibrium Shift" driven by survival probability is precisely the advanced strategic form that this paper attempts to capture using reinforcement learning.

4 Methodology

4.1 Environment Design

To accurately simulate the complex gameplay of Liar's Bar within a reinforcement learning framework, we constructed a high-fidelity simulation environment based on the OpenAI Gym interface standards, formalizing it as a Multi-Agent Partially Observable Markov Decision Process (Multi-Agent POMDP). In this environment, although the underlying game logic (such as card shuffling and bullet placement) is deterministic or follows specific distributions, agents cannot access the global state $\mathcal{S}$ and are restricted to perspective-based local observations $\mathcal{O}$. To enable the neural network to effectively process unstructured game information, we mapped the discrete game states into a compact 33-dimensional continuous observation vector $\mathbf{o}_t$. This vector is not a mere stacking of raw data but a product of meticulous feature engineering, concatenated from four key feature subspaces: $\mathbf{o}_t = [\mathbf{v}_{target} \| \mathbf{v}_{identity} \| \mathbf{v}_{game} \| \mathbf{v}_{hand}]$. Here, $\mathbf{v}_{target}$ is a one-hot vector encoding the target card rank required for the current round (e.g., King, Queen, Ace, or Joker), while $\mathbf{v}_{identity}$ identifies the active player and the status of surviving players, forming the social topology of the game.

In the design of the game state vector $\mathbf{v}_{game}$, we explicitly introduced a quantification of survival risk. Unlike traditional poker games, the core dynamic of Liar's Bar stems from the "Russian Roulette" mechanism. To allow agents to perceive this lethal risk, we transformed the physical state of the revolver (i.e., the number of empty chambers n_{empty}) into a normalized risk scalar $\lambda_{risk} \in [0, 1]$. This scalar is calculated as follows:

$$\lambda_{risk} = \frac{1}{1 + n_{empty}} \tag{2}$$

This design not only reflects the hyperbolic increase in death probability as the number of empty clicks rises but also implies a dynamic adjustment signal for the future discount factor γ. Furthermore, to capture the strategic intent of opponents, $\mathbf{v}_{game}$ includes a summary of the previous round's actions—specifically, the number of cards the previous player claimed to play and whether a

doubt was initiated—providing a foundational context for subsequent time-series analysis. Finally, $\mathbf{v}_{hand}$, designed as a multi-dimensional discrete count vector, precisely describes the distribution of the agent's private hand across various ranks.

The design of the action space $\mathcal{A}$ presents similar challenges, as Liar's Bar requires players to make two distinct types of decisions within a single turn: information gaming (whether to doubt) and resource management (how to play cards). A traditional flattened action space would lead to combinatorial explosion; therefore, we designed the action space as two decoupled decision branches: a Doubt Branch and a Play Branch. The Doubt Branch is a binary action space $a_{doubt} \in \{0, 1\}$, controlling whether to challenge the previous player. The Play Branch is a multi-dimensional binary vector space $\mathbf{a}_{play} \in \{0, 1\}^5$, corresponding to the maximum of 5 cards held by the agent, where each bit indicates whether to play the card at that position. Under this composite action space, the agent's policy π is effectively a joint distribution $\pi(a_{doubt}, \mathbf{a}_{play}|\mathbf{o}_t)$. To ensure action validity, we implemented an Action Masking mechanism at the environment level. This ensures agents do not attempt to manipulate empty hand positions or make illegal decisions when a response to a doubt is required, thereby significantly reducing the invalid exploration space and enhancing training efficiency.

4.2 Network Architecture

In a quintessential imperfect information game like Liar's Bar, a single instantaneous observation is often insufficient to reveal the opponents' true intentions; the essence of the game lies in inferring private information from the historical behavioral patterns of opponents. Consequently, traditional feed-forward neural networks based on the Markov assumption perform poorly in such scenarios. To effectively capture long-horizon game dependencies and extract implicit deception patterns, we designed an asymmetric Actor-Critic architecture integrated with a Transformer Encoder. This architecture employs a dual-stream design at the feature extraction layer. The first stream processes the current static observation vector $\mathbf{o}_t$, utilizing a Multi-Layer Perceptron (MLP) coupled with residual connections to map it into a high-dimensional state embedding $\mathbf{e}_{state}$, thereby preserving precise features of the current situation. The second stream, the core of the architecture, is dedicated to processing a history action sequence of length T, denoted as $\mathcal{H}_{t-T:t} = [\mathbf{h}_{t-T}, \ldots, \mathbf{h}_t]$, where each $\mathbf{h}_\tau$ encodes the player's identity, action type, and feedback result at that moment.

To deeply mine contextual information within the historical sequence, we introduced the Multi-Head Self-Attention mechanism of the Transformer. Unlike Recurrent Neural Networks such as LSTMs or GRUs, the Transformer can attend to any position in the sequence in parallel, thus more acutely capturing subtle strategic shifts that span multiple rounds (e.g., an opponent suddenly changing their playing rhythm after three consecutive empty clicks). The core computation of the attention mechanism is formally expressed as:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \tag{3}$$

Through this mechanism, the network dynamically assigns weights to historical segments, suppressing noise and focusing on key frames with high game-theoretic value, ultimately aggregating them to generate a historical context embedding $\mathbf{e}_{hist}$. Subsequently, the state embedding and history embedding are concatenated and normalized through a Feature Fusion Layer to form a joint

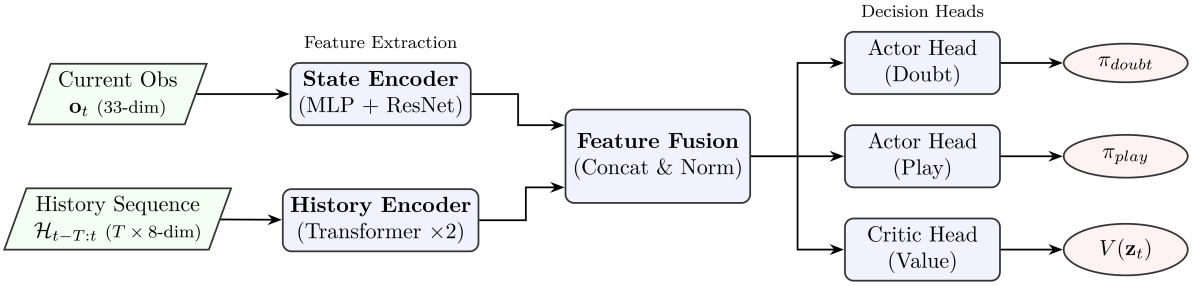

Fig 1: The Proposed Transformer-based Actor-Critic Architecture

feature vector $\mathbf{z}_t = \text{LayerNorm}(\mathbf{W}_f[\mathbf{e}_{state} \| \mathbf{e}_{hist}])$, serving as the unified representation for subsequent decision-making.

At the decision output end, the Actor network discards the traditional single output layer in favor of a decoupled Two-Head structure to accommodate the composite action space. The first output head is the "Doubt Head," which maps the joint features to binary logits, outputting action probabilities $\pi_{doubt}(a|\mathbf{z}_t)$ following a categorical distribution to determine whether to challenge the opponent. The second output head is the "Play Head," which outputs independent Bernoulli distribution parameters for each card in the hand, forming a multi-dimensional action probability $\pi_{play}(\mathbf{a}|\mathbf{z}_t)$. To facilitate training, the Critic network estimates the state value function $V(\mathbf{z}_t)$ based on the same joint feature $\mathbf{z}_t$. This design, which shares the feature extractor while separating the decision heads, not only significantly reduces the number of model parameters but also promotes the generalization ability of feature representations across different tasks, enabling the agent to precisely execute tactical actions while understanding the value of the situation.

4.3 Personality Shaping via Reward Engineering

In the self-play training of Multi-Agent Reinforcement Learning (MARL), systems are prone to falling into local optima of Nash Equilibrium, leading to strategy homogenization where all agents converge to a single, rigid behavioral pattern. However, in real Liar's Bar matches, human players exhibit vastly different styles, ranging from bluffing adventurers to cautious conservatives. To construct a highly heterogeneous and ecologically diverse gaming environment, and thereby train a Rational Agent with strong generalization capabilities, this paper proposes a reward engineering mechanism based on "Personality Shaping". We decompose the agent's total reward function R_{total} into two parts: the sparse objective environmental reward R_{env} and the dense intrinsic personality reward $R_{persona}$, formulated as $R_{total} = R_{env} + \beta \cdot R_{persona}$. Here, R_{env} is solely responsible for conveying game outcomes (+5/-5) and basic play feedback, while $R_{persona}$ utilizes carefully designed shaping terms to induce the emergence of specific behavioral preferences.

4.3.1 AggressiveBluffer: The Art of Deception

The **AggressiveBluffer** is designed to simulate aggressive players who tend to gain informational advantages and psychological dominance through high-risk deception. The core strategy of this agent lies in frequently playing cards that do not match the target rank to obfuscate opponents' judgment. To quantify this trait, we introduce a deception incentive mechanism. Let the current

Table 1: Reward Shaping Configuration for Different Personalities

Agent Persona	**Key Behavior**	**Reward Function Characteristic**
AggressiveBluffer	High Deception	$R_{persona} = +r_{bluff}$ (if lying & safe)
AggressiveChallenger	High Doubt	$R_{persona} = +r_{action}$ (on doubt) + High $r_{success}$
Conservative	Survival First	$R_{persona} = -e^{\lambda_{risk}}$ (risk penalty)
Rational	Balanced	$R_{persona} = 0$ (Pure Environment Reward)

target rank be c_{target} and the set of cards actually played be H_{play}. If the agent commits a deceptive act (i.e., $\exists c \in H_{play}, c \neq c_{target}$) and is not immediately detected (Action $a_{doubt} = 0$), a positive intrinsic reward $r_{bluff} > 0$ is granted. Conversely, if the agent chooses to play honestly, a minor regret penalty $r_{regret} < 0$ is applied to suppress its conservative tendencies. This asymmetric reward structure forces the agent to actively explore the boundaries of deception, learning to probe wildly on the edge of being doubted.

4.3.2 AggressiveChallenger: The Vigilante

The **AggressiveChallenger** plays the role of the "police" on the field, characterized by extreme sensitivity to opponents' behaviors and a high propensity to doubt. The existence of such a strategy is crucial for purifying the game environment and curbing excessive bluffing. To shape this personality, we reshape the reward surface for the doubting action. Whenever the agent chooses to initiate a challenge ($a_{doubt} = 1$), regardless of the outcome, a basic action reward r_{action} is given to encourage intervention in the situation. Simultaneously, we significantly amplify the reward weight for successful doubts (catching a liar) and markedly reduce the penalty threshold for failed doubts. This design alters the agent's risk-reward assessment, fostering a radical style of "better to mistake than to miss," thereby constituting a persistent survival deterrent to other players.

4.3.3 Conservative: The Survivalist

Distinct from the previous two, the **Conservative** agent regards "survival" as the highest, or even the sole, behavioral criterion. Under the Russian Roulette mechanism of Liar's Bar, death implies that returns instantly drop to zero or negative infinity. Therefore, the reward function for the conservative agent is designed to be strongly negatively correlated with the current survival risk. We introduce a risk penalty term based on the cylinder state, $P_{risk}(\lambda) = -e^{\lambda_{risk}}$, where λ_{risk} is the probability of death upon the current trigger pull. As the number of empty clicks increases, λ_{risk} rises, and this penalty term grows exponentially. This forces the agent to strive to avoid passive situations (being doubted) in the late stages of the game, tending to play only absolutely honest cards and adopting an extremely cautious doubting strategy against opponents, refusing to pull the trigger unless the win rate is extremely high.

4.3.4 Rational: The Unbiased Optimizer

The **Rational** agent is the target model we ultimately aim to train and serves as the "predator" within the aforementioned personality ecosystem. Unlike agents with pre-set personalities, the

intrinsic reward weight for the Rational agent is set to $\beta = 0$, meaning it is completely unaffected by artificial biases and aims purely to maximize the objective environmental reward R_{env} (i.e., final win rate and survival rate). Placing the Rational agent in a mixed game pool composed of Bluffer, Challenger, and Conservative agents forces it to learn to dynamically infer opponents' personality classes by observing their historical action sequences (encoded by the Transformer). For instance, it learns to increase doubt frequency after identifying a Bluffer, or to exploit the timidity of a Conservative to apply pressure. The evolutionary process of the Rational agent is essentially a search for the Best Response against a heterogeneous group of strategies, thereby approximating a more robust Bayesian Nash Equilibrium point.

4.4 Training Strategy

To achieve stable policy optimization within a complex multi-agent gaming environment, this paper employs Proximal Policy Optimization (PPO) as the core training algorithm. PPO is an On-Policy algorithm based on the Actor-Critic architecture that introduces a clipping mechanism to limit the update step size between the new and old policies, thereby effectively preventing performance collapse caused by drastic changes in training data distribution. The specific objective function is defined as:

$$L(\theta) = \hat{\mathbb{E}}_t \left[\min(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t) \right] \tag{4}$$

where $r_t(\theta)$ is the probability ratio of the new policy to the old policy, $\hat{A}_t$ is the estimated advantage function, and ϵ is the clipping hyperparameter. Given that Liar's Bar involves complex imperfect information game logic, training from scratch could lead agents into meaningless random exploration. Therefore, we designed a two-stage Curriculum Learning scheme aimed at guiding agents from mastering basic rules to developing advanced adversarial strategies.

The first stage is **Homogeneous Self-Play**. In this phase, we construct three independent training environments, each containing only agents of a specific personality (e.g., AggressiveBluffer). Agents compete against copies of themselves or their peers, updating parameters under the guidance of their respective personality reward functions. The primary goal of this stage is for agents to rapidly acquire the legal action space of the game (such as rules for playing cards and timing for doubts) and internalize their assigned personality traits. For instance, the Challenger agent gradually develops a conditioned reflex for high-frequency doubting, while the Conservative agent learns to avoid operations under high-risk cylinder states. This homogeneous training provides Baseline Models with distinct behavioral characteristics for the subsequent complex gaming.

The second stage is **Heterogeneous Mixed-Play**, which is the pivotal component of strategy evolution. We construct a mixed agent pool containing the pre-trained Bluffer, Challenger, and Conservative agents, freezing their network parameters to serve as Fixed Opponents within the environment. Subsequently, the unbiased **Rational** agent is placed into this environment to train from scratch. In this stage, the Rational agent faces a non-stationary and hostile heterogeneous environment where it cannot win by simply overfitting to a single strategy. Instead, it must learn to utilize the historical features extracted by the Transformer to infer the personality class of the current opponent and dynamically adjust its strategy accordingly—for example, reducing deception frequency when facing a Challenger, or leveraging the risk-aversion of a Conservative to apply pressure. This mixed training forces the Rational agent to seek the Best Response against diverse

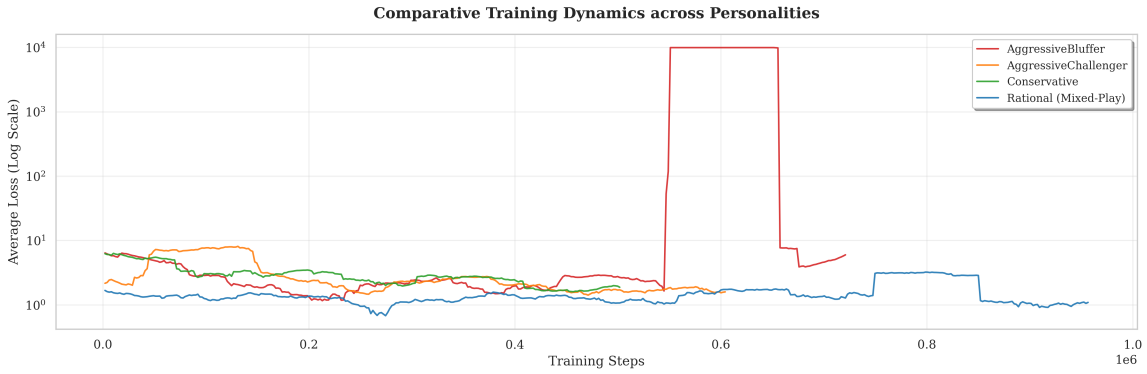

Fig 2: Comparative Training Dynamics across Personalities

opponents in a multi-party melee, thereby converging to a more robust Bayesian Nash Equilibrium point.

5 Experiments

5.1 Experimental Configuration & Training Dynamics

To comprehensively validate the effectiveness and robustness of the proposed Liar's Bar strategy learning framework, we conducted a series of experiments in a high-fidelity simulation environment built upon the PyTorch framework, with all computational tasks performed on a single-node workstation equipped with NVIDIA CUDA acceleration. Regarding the network architecture, the Actor and Critic networks share a lightweight Transformer encoder configured with a stack of 2 layers ($L = 2$), each containing 4 multi-head attention mechanisms ($H = 4$), with a hidden layer dimension set to $d_{model} = 64$. This configuration ensures that the model possesses sufficient temporal capture capabilities while avoiding the risk of overfitting under limited samples. The Adam optimizer was selected with an initial learning rate of $\alpha = 5 \times 10^{-4}$, coupled with a linear decay strategy. Considering the paramount importance of "survival" in the game, we set the discount factor to $\gamma = 0.99$ to ensure that agents, when assessing value, fully weigh current gains against long-term survival probabilities. The PPO algorithm's clipping parameter ϵ was set to 0.2, and the entropy regularization coefficient was set to 0.01 to maintain exploration vitality during the early stages of training.

The training process followed the two-stage Curriculum Learning strategy described previously, achieving not only strategy generation from scratch but also ensuring training stability. The first stage, the "Personality Solidification" phase, involved training agents with four distinct personality settings in homogenous environments for 10,000 episodes of self-play each. The second stage, the "Adversarial Evolution" phase, integrated the pre-trained AggressiveBluffer, AggressiveChallenger, and Conservative agents as fixed parts of the environment. The Rational agent then underwent reinforcement training for 20,000 episodes within this heterogeneous environment, with an update horizon set to 2048 steps.

The dynamics of the loss function during training revealed the convergence characteristics of different strategies in imperfect information games. Observing the loss curves from the first stage, the **Conservative** and **AggressiveChallenger** agents exhibited extremely fast convergence

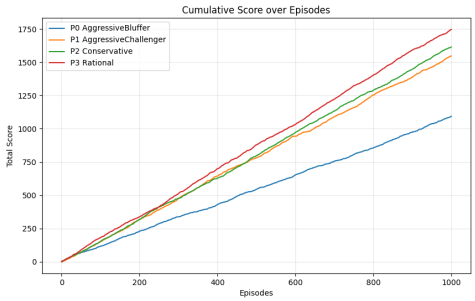

Fig 3: Cumulative Score over Episodes

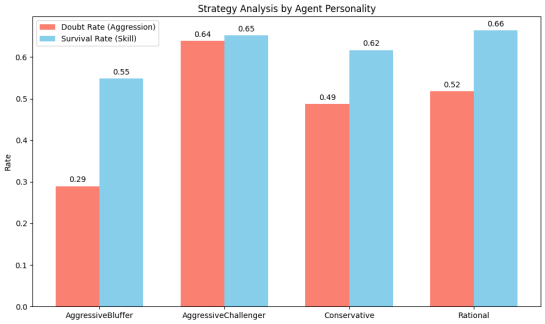

Fig 4: Strategy Analysis by Agent Personality

speeds. This is attributed to the relatively narrow and well-defined optimal policy spaces guided by their reward functions (i.e., a tendency towards honesty or doubting). However, during the early training of the **AggressiveBluffer**, a significant "Loss Spike" was observed. In-depth analysis indicates that this was due to the agent frequently triggering the extreme negative penalty ($r_{elim} = -5.0$) of the "Russian Roulette" mechanism while exploring high-frequency deception strategies, leading to a surge in the variance of the Advantage function estimation within a short period. Thanks to the Trust Region mechanism of the PPO algorithm, the model successfully suppressed malignant gradient oscillations and quickly returned to a steady state. Upon entering the second stage, despite the Rational agent facing a highly non-stationary mixed gaming environment, its average loss remained consistently at a low level ($L_{avg} < 5.0$), with no catastrophic forgetting observed. This suggests that the Rational agent not only inherited basic survival instincts from the pre-trained model but also successfully learned to find a robust Bayesian Nash Equilibrium amidst dynamically changing opponent strategy distributions.

5.2 Quantitative Evaluation

To conduct a fair and statistically significant evaluation of the trained agents' performance, we executed 1,000 independent Monte Carlo simulated matches on the test set. Before delving into the experimental results, we first define three core metrics to quantify the agents' gaming performance. The first is the **Average Score** ($\bar{S}$), which reflects the agent's survival ranking capability. In a game with $N = 4$ players, if agent i is the k-th to be eliminated (where $k \in \{0, 1, 2\}$, and the winner corresponds to $k = 3$), its score is assigned as $S_i = k$. This metric is directly correlated with the final win rate and comprehensively measures the robustness of the strategy. The second is the **Doubt Rate** (ρ_{doubt}), defined as the frequency with which an agent chooses to initiate a doubt action: $\rho_{doubt} = \frac{1}{T}\sum_{t=1}^{T} \mathbb{I}(a_{t,\text{doubt}} = 1)$. This metric characterizes the agent's aggressiveness and risk appetite. The third is the **Strategy Success Rate** (η_{succ}), used to measure the effectiveness of decisions. It is defined as the ratio of the sum of "successful bluffs (not doubted)" and "successful doubts (catching a liar)" to the total number of critical decisions. A higher value indicates more precise situational judgment by the agent.

The experimental results based on these metrics present a clear stratification in performance, strongly validating the effectiveness of the mixed-game training strategy proposed in this paper. The **Rational** agent topped the leaderboard with an average score of **1.75**, significantly outperforming other personality-based baseline models (Conservative: 1.61, Challenger: 1.55). This result indicates

Table 2: Average Scores, Doubt Rates and Survival Rates of Different Personalities

Agent Persona	Average Score	Doubt Rate	Survival Rate
AggressiveBluffer	1.09	0.29	0.55
AggressiveChallenger	1.55	**0.64**	0.65
Conservative	1.61	0.49	0.62
Rational	**1.75**	0.52	**0.66**

that, without the constraint of artificial personality bias (i.e., $\beta = 0$), the reinforcement learning algorithm can automatically evolve an equilibrium strategy that balances profit maximization with risk aversion in a complex non-stationary environment. In contrast, the **AggressiveBluffer** performed the worst, with an average score of only **1.09**. This data provides a critical game-theoretic insight: in games like Liar's Bar featuring an "irreversible elimination mechanism," survival is the prerequisite for profit. Although the AggressiveBluffer's low doubt rate ($\rho_{doubt} = 0.29$) allowed it to gain some short-term advantage by shedding cards, its overly aggressive deception provoked a high risk of being doubted. This led to a sharp increase in the expectation of death during "Russian Roulette," thereby significantly dragging down the overall utility in the long-tail distribution.

Further analyzing the microscopic behavioral statistics of the agents, we can clearly observe the significant shaping effect of reward engineering on strategy forms. The **AggressiveChallenger** exhibited the highest doubt rate (**0.64**), acting as expected as a "filter" in the gaming environment. However, this aggressive doubting style proves mathematically to be a double-edged sword, as high-frequency doubting inevitably encounters "honest counterattacks," causing backlash against its own survival rate. Most notably, the **Rational** agent's behavioral pattern stabilized at a moderate doubt rate of **0.52**, yet it achieved the highest Strategy Success Rate of **0.66**. This suggests that the Rational agent does not blindly choose neutrality or random actions but possesses precise Bayesian risk assessment capabilities. Leveraging historical context information extracted by the Transformer, it dynamically computes the expected value of actions. The Rational agent initiates a doubt decisively only when the posterior probability of the opponent bluffing, $P(Bluff|o_t)$, is sufficiently high and the potential gain outweighs the cost of death risk posed by the current cylinder state λ_{risk}. This dynamic risk management capability, based on informational advantage, is the core reason for its established dominance in the multi-party heterogeneous melee.

5.3 Qualitative Analysis: Case Studies

To investigate the microscopic mechanisms by which the Transformer architecture captures long-horizon gaming patterns, we extracted several representative game logs from the test set for qualitative review. By comparing the behavioral trajectories of different winners, we identified a significant mapping relationship between the agents' decision logic and their personality presets. In analyzing the cases won by the **AggressiveBluffer** (see Appendix A), we observed that its victories were often built upon extremely high variance. This agent persisted in high-frequency play even when its hand did not match the target (e.g., playing an 'Ace' when the target was 'Queen'). Although it was repeatedly detected and doubted by the **AggressiveChallenger**, it survived several rounds of "Russian Roulette" thanks to the low probability of a live round in the early cylinder state

($P_{death} \approx 1/6$). While this victory based on "survivorship bias" statistically dragged down its average score, it also revealed the potential of aggressive strategies to disrupt opponents' psychological defenses in short-term games. Conversely, the behavioral pattern of the **AggressiveChallenger** manifested as "indiscriminate aggression," the existence of which significantly compressed the deception space for other players, forcing the game towards an honest equilibrium, though it often suffered backlash for doubting honest players.

In contrast, the game logs won by the **Rational** agent (see Appendix D) demonstrated superior adaptive strategies and risk management capabilities. Thanks to the multi-head attention mechanism of the Transformer, the Rational agent was able to attend to opponent behavioral features spanning multiple time steps in the historical sequence. For instance, when facing the **AggressiveChallenger**, the Rational agent significantly reduced its frequency of deception, tending to play safe cards or act cautiously before that opponent moved, thereby avoiding the risk of collateral damage from high-frequency doubting. However, when facing a **Conservative** opponent, the Rational agent exhibited stronger aggression. The logs show that when the Conservative agent was forced to play an unusual number of cards (e.g., playing 3 cards at once), the Rational agent swiftly captured this anomaly signal, inferred an increased posterior probability that the opponent's resources were exhausted, $P(Resource\_Exhausted|\mathcal{H}_{hist})$, and decisively initiated a doubt to eliminate the opponent. This capability for Policy Adaptation, adjusting strategies dynamically based on opponent identity, proves that the model has not only learned the game rules but has also constructed personality modeling of other agents in the latent space, thereby establishing an informational advantage in the non-stationary multi-party melee.

6 Conclusion

This paper has deeply investigated Liar's Bar, an imperfect information game integrating deception mechanics with lethal stochasticity, by constructing a high-fidelity simulation environment and a multi-agent reinforcement learning framework. The results indicate that incorporating the Transformer architecture into the Actor-Critic network significantly enhances the agent's ability to extract features from unstructured historical information in long-horizon games. This finding aligns with the assertions made by Raiman et al. [12] regarding the importance of long-term planning and situational awareness in their OpenAI Five research. Through quantitative analysis of experimental data, we confirmed that in survival games featuring an "irreversible elimination mechanism," neither pure aggressive bluffing nor excessive conservatism constitutes the optimal solution. Conversely, the Rational agent, trained via personality-based reward shaping and adversarial evolution within a heterogeneous game pool, successfully acquired dynamic risk management strategies, thereby achieving a significant advantage in both win rate and survival rate. This result not only corroborates the theory proposed by Arulkumaran et al. [8] that population diversity is crucial for solving non-transitive game strategies but also provides supplementary evidence from a deep reinforcement learning perspective to the equilibrium studies of this game by Li and Miao [11], suggesting that the optimal strategy is not a static Nash Equilibrium but a Bayesian Best Response that drifts dynamically with the state of the cylinder.

Although the framework proposed in this paper performs exceptionally well in fixed-player simulated matches, certain limitations remain, which also illuminate directions for future research.

Firstly, the current observation space is limited to numerical game states, neglecting natural language interactions or non-verbal cues (such as hesitation time before playing) that might be involved in actual human-computer confrontation in Liar's Bar; future work could attempt to integrate multimodal inputs to enhance the realism of the gaming experience. Secondly, the current training is primarily based on offline self-play; the model's adaptability remains to be verified when facing the highly creative and psychologically inductive irrational strategies of human players, as demonstrated in the videos by Lin Yi [2, 3]. Future endeavors could explore incorporating Large Language Models (LLMs) into the gaming framework, leveraging their powerful few-shot reasoning capabilities to simulate more "human-like" opponents. Alternatively, research could focus on online adaptation algorithms, enabling agents to update their belief distributions regarding human opponents' personalities in real-time during gameplay, thereby achieving true general intelligence in broader human-computer gaming scenarios.

7 Acknowledgements

I would like to express my sincere gratitude to Prof. Chen Xu for his mentorship in the field of reinforcement learning. His profound academic insights and expert guidance have been instrumental to my journey, providing clarity and direction throughout the course of this research. I am equally deeply thankful to my fiancée, Ms. Ma Yujie, for her unwavering support and quiet encouragement. Her understanding and companionship have served as a pillar of strength, enabling me to navigate the challenges of this project with focus and determination. It has been nearly two years since I embarked on my path in reinforcement learning, and I wish to extend my appreciation to all the teachers and classmates who have offered their help and inspiration along the way. Their collective wisdom and the stimulating academic environment they fostered have been invaluable assets in my continuous growth and exploration.

References

[1] 董豪, 丁子涵, 仉尚航等. 深度强化学习: 基础、研究与应用 [M]. 第 1 版. 北京: 电子工业出版社, 2021.

[2] 林亦 LYi. AI 大战骗子酒馆! 四大顶级 AI 国际服赌命厮杀, 赢家会是? [EB/OL]. 2025[2025-03-07]. https://www.bilibili.com/video/BV1aL92YoEEe/?spm_id_from=333.1391.0.0&vd_source=80dc69d86d35a3975d8d993d67e78ad1.

[3] 林亦 LYi. AI 大战骗子酒馆 ： 八大顶级 AI 赌命血战，赢家竟是? [EB/OL]. 2025[2025-09-26]. https://www.bilibili.com/video/BV1R6nMzSESL/?spm_id_from=333.1391.0.0&vd_source=80dc69d86d35a3975d8d993d67e78ad1.

[4] 夏大慰. 博弈论：策略互动、信息与激励 [M]. 第 1 版. 北京: 机械工业出版社, 2025.

[5] 张伟楠, 沈键, 俞勇. 动手学强化学习 [M]. 第 1 版. 北京: 人民邮电出版社, 2022.

[6] 赵世钰. 强化学习的数学原理 [M]. 第 1 版. 北京: 清华大学出版社, 2025.

[7] 赵世钰. 强化学习的数学原理：英文 [M]. 第 1 版. 北京: 清华大学出版社, 2024.

[8] Arulkumaran K, Cully A, Togelius J. Alphastar: An evolutionary computation perspective[C]//Proceedings of the genetic and evolutionary computation conference companion. 2019: 314-315.

[9] Johnson T, Bangay S, Sterne P. An evaluation of how Dynamic Programming and Game Theory are applied to Liar' s Dice[J]. 2007.

[10] Lee E, Lian R, Xu B. WiLDCARD: Winning Liar' s Dice with Calculative Agents and Reinforced Decision-Making[J].

[11] Li L, Miao P. Bluff and Learn: Comparing CFR and NFSP in Liar Bar[J].

[12] Raiman J, Zhang S, Wolski F. Long-term planning and situational awareness in openai five[J]. arXiv preprint arXiv:1912.06721, 2019.

Appendix

A Victory Case of AggressiveBluffer (P0)

This game log records a typical victory process of the **AggressiveBluffer (P0)**, vividly illustrating the survival logic of a "chaos agent" in a highly stochastic environment. In the early stage, P0 adopted an extremely aggressive strategy, bluffing with an irrelevant card ('Ace') against the target ('Jack') in the very first turn. Although detected by the **AggressiveChallenger (P1)**, P0 survived the "Russian Roulette" thanks to the low probability of a live round (1/6) in the initial cylinder state. This "survivorship bias" allowed P0 to maintain a hand advantage. Finally, in the endgame, P0 employed a mixed strategy using both a genuine 'Joker' and bluff cards. This irrational mixture successfully disrupted the Bayesian inference of the **Rational (P3)** agent, causing P3 to initiate a doubt at the wrong moment and eventually be eliminated. This case demonstrates that while aggressive bluffing is not optimal in terms of long-term expectation, it possesses extremely high variance and disruptive power in single episodes, capable of breaking the Nash Equilibrium by jamming the information field.

```
=== Liar's Bar Log | 2026-01-14 22:10:55.296185 ===
File: evaluate_game/game_470.txt

--- Game Start ---
P0 (AggressiveBluffer) Hand: ['A', 'J', 'K', 'Q', 'A']
P1 (AggressiveChallenger) Hand: ['J', 'K', 'J', 'A', 'K']
P2 (Conservative) Hand: ['J', 'Q', 'K', 'Q', 'A']
P3 (Rational) Hand: ['Joker', 'Q', 'K', 'Q', 'A']
Turn P0: Played ['A']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
  > LIAR CAUGHT! P0 was lying (Played ['A'] on J).
  > CLICK. P0 survives.
Turn P2: DOUBT!
Turn P3: Played ['Joker']
Turn P0: Played ['J']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
  > TRUTH! P0 was honest. P1 plays Russian Roulette.
  > CLICK. P1 survives.
Turn P2: DOUBT!
Turn P3: DOUBT!
Turn P0: Played ['K']
Turn P1: Played ['J']
Turn P2: DOUBT!
!!! P2 doubts P1 !!!
```

```
  > TRUTH! P1 was honest. P2 plays Russian Roulette.
> BANG! P2 is eliminated.Turn P3: DOUBT!
Turn P0: Played ['Q']
Turn P1: Played ['K']
Turn P3: DOUBT!
!!! P3 doubts P1 !!!
  > LIAR CAUGHT! P1 was lying (Played ['K'] on J).
  > CLICK. P1 survives.
Turn P0: Played ['A']
Turn P1: Played ['J']
Turn P3: Played ['Q']
Turn P0: Played ['A']
Turn P1: Played ['A']
Turn P3: Played ['K']
Turn P0: DOUBT!
!!! P0 doubts P3 !!!
  > LIAR CAUGHT! P3 was lying (Played ['K'] on J).
  > CLICK. P3 survives.
Turn P1: Played ['K']
Turn P3: DOUBT!
!!! P3 doubts P1 !!!
  > LIAR CAUGHT! P1 was lying (Played ['K'] on J).
  > BANG! P1 is eliminated.
Turn P0: DOUBT!
Turn P3: DOUBT!
Turn P0: Played ['Joker']
Turn P3: DOUBT!
!!! P3 doubts P0 !!!
  > TRUTH! P0 was honest. P3 plays Russian Roulette.
  > BANG! P3 is eliminated.
-> Game Over. Winner: P0
```

B Victory Case of AggressiveChallenger (P1)

Appendix B demonstrates how the **AggressiveChallenger (P1)** reshapes the game ecology through an indiscriminate and aggressive doubting strategy. In this episode, P1 acted as a "filter" on the field, maintaining extreme sensitivity to the playing behaviors of other agents. The log shows that P1 hesitated not to initiate doubts even when opponents played honest cards (such as a 'Joker' or the true rank). While this seemingly irrational hyper-aggression forced P1 itself to face the death risk of the roulette multiple times, it also severely compressed the survival space for both the **AggressiveBluffer (P0)** and the **Rational (P3)** agents. By continuously triggering adjudications, P1 accelerated the rotation of the cylinder state, forcing opponents who attempted

to profit from deception to be eliminated quickly. This case reveals the double-edged sword effect of the aggressive challenging strategy: it cleanses deceivers from the field through high-pressure tactics, yet its ultimate victory relies heavily on the luck of surviving consecutive self-triggered risk events.

```
=== Liar's Bar Log | 2026-01-14 22:09:37.282879 ===
File: evaluate_game/game_98.txt

--- Game Start ---
P0 (AggressiveBluffer) Hand: ['K', 'Joker', 'A', 'J', 'Q']
P1 (AggressiveChallenger) Hand: ['Q', 'J', 'J', 'Joker', 'J']
P2 (Conservative) Hand: ['Q', 'A', 'A', 'Q', 'J']
P3 (Rational) Hand: ['K', 'Q', 'A', 'A', 'A']
Turn P2: Played ['A']
Turn P3: Played ['K']
Turn P0: DOUBT!
!!! P0 doubts P3 !!!
  > TRUTH! P3 was honest. P0 plays Russian Roulette.
  > CLICK. P0 survives.
Turn P1: DOUBT!
Turn P2: Played ['Q']
Turn P3: Played ['Q']
Turn P0: DOUBT!
!!! P0 doubts P3 !!!
  > LIAR CAUGHT! P3 was lying (Played ['Q'] on K).
  > CLICK. P3 survives.
Turn P1: DOUBT!
Turn P2: DOUBT!
Turn P3: DOUBT!
Turn P0: DOUBT!
Turn P1: DOUBT!
Turn P2: Played ['J']
Turn P3: DOUBT!
!!! P3 doubts P2 !!!
  > LIAR CAUGHT! P2 was lying (Played ['J'] on K).
  > CLICK. P2 survives.
Turn P0: Played ['K']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
  > TRUTH! P0 was honest. P1 plays Russian Roulette.
  > CLICK. P1 survives.
Turn P2: Played ['Q']
Turn P3: DOUBT!
```

```
!!! P3 doubts P2 !!!
> LIAR CAUGHT! P2 was lying (Played ['Q'] on K).  > BANG! P2 is
     eliminated.
Turn P0: DOUBT!
Turn P1: Played ['Q']
Turn P3: Played ['A']
Turn P0: DOUBT!
!!! P0 doubts P3 !!!
  > LIAR CAUGHT! P3 was lying (Played ['A'] on K).
  > BANG! P3 is eliminated.
Turn P1: DOUBT!
Turn P0: Played ['Joker']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
  > TRUTH! P0 was honest. P1 plays Russian Roulette.
  > CLICK. P1 survives.
Turn P0: Played ['A']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
  > LIAR CAUGHT! P0 was lying (Played ['A'] on K).
  > CLICK. P0 survives.
Turn P0: Played ['J']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
  > LIAR CAUGHT! P0 was lying (Played ['J'] on K).
  > BANG! P0 is eliminated.
-> Game Over. Winner: P1
```


C Victory Case of Conservative (P2)

This game log serves as a perfect portrayal of the **Conservative (P2)** agent's "survival first" strategy. Throughout the game, P2 maintained an extremely low operation frequency, playing only an absolutely safe 'Joker' in the early stage before shifting into a defensive posture. Meanwhile, the **AggressiveBluffer (P0)** and **AggressiveChallenger (P1)** engaged in fierce mutual attrition, where P1's continuous doubting caused the probability of a live round in the cylinder to rise rapidly. Capitalizing on this situation, P2 avoided all high-risk adjudications through an extreme conflict-avoidance strategy, waiting for the aggressive opponents to eliminate themselves. It was not until the very end, when only one opponent remained and the cylinder's lethality was critically high, that P2 delivered a fatal strike based on solid informational advantage. This outcome strongly validates the "second-mover advantage" in game theory, proving that in games with irreversible elimination mechanisms, conservative strategies that minimize variance often yield the highest survival expectation.

```
=== Liar's Bar Log | 2026-01-14 22:10:06.818229 ===
File: evaluate_game/game_237.txt
--- Game Start ---
P0 (AggressiveBluffer) Hand: ['A', 'J', 'Q', 'J', 'K']
P1 (AggressiveChallenger) Hand: ['K', 'A', 'J', 'Joker', 'A']
P2 (Conservative) Hand: ['J', 'A', 'Joker', 'J', 'Q']
P3 (Rational) Hand: ['Q', 'Q', 'A', 'K', 'J']
Turn P3: DOUBT!
Turn P0: Played ['A']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
  > TRUTH! P0 was honest. P1 plays Russian Roulette.
  > CLICK. P1 survives.
Turn P2: DOUBT!
Turn P3: DOUBT!
Turn P0: DOUBT!
Turn P1: DOUBT!
Turn P2: Played ['Joker']
Turn P3: DOUBT!
!!! P3 doubts P2 !!!
  > TRUTH! P2 was honest. P3 plays Russian Roulette.
  > BANG! P3 is eliminated.
Turn P0: Played ['J']
Turn P1: Played ['K']
Turn P2: DOUBT!
!!! P2 doubts P1 !!!
  > LIAR CAUGHT! P1 was lying (Played ['K'] on A).
  > CLICK. P1 survives.
Turn P0: Played ['Q']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
  > LIAR CAUGHT! P0 was lying (Played ['Q'] on A).
  > CLICK. P0 survives.
Turn P2: Played ['J']
Turn P0: Played ['J']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
  > LIAR CAUGHT! P0 was lying (Played ['J'] on A).
  > CLICK. P0 survives.
Turn P2: Played ['Q']
Turn P0: Played ['K']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
```

```
  > LIAR CAUGHT! P0 was lying (Played ['K'] on A).
> CLICK. P0 survives.Turn P2: DOUBT!
Turn P0: Played ['J']
Turn P1: Played ['A']
Turn P2: Played ['J']
Turn P0: Played ['Q']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
  > LIAR CAUGHT! P0 was lying (Played ['Q'] on A).
  > CLICK. P0 survives.
Turn P2: DOUBT!
Turn P0: Played ['K']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
  > LIAR CAUGHT! P0 was lying (Played ['K'] on A).
  > CLICK. P0 survives.
Turn P2: DOUBT!
Turn P0: DOUBT!
Turn P1: Played ['J']
Turn P2: Played ['A']
Turn P0: Played ['A']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
  > TRUTH! P0 was honest. P1 plays Russian Roulette.
  > CLICK. P1 survives.
Turn P2: DOUBT!
Turn P0: Played ['J']
Turn P1: Played ['Joker']
Turn P2: Played ['K']
Turn P0: Played ['A']
Turn P1: Played ['A']
Turn P2: Played ['A']
Turn P0: Played ['K']
Turn P1: Played ['Q']
Turn P2: Played ['J']
Turn P0: Played ['K']
Turn P1: DOUBT!
!!! P1 doubts P0 !!!
  > LIAR CAUGHT! P0 was lying (Played ['K'] on A).
  > BANG! P0 is eliminated.
Turn P2: DOUBT!
Turn P1: Played ['Q']
Turn P2: DOUBT!
```

```
!!! P2 doubts P1 !!!
> LIAR CAUGHT! P1 was lying (Played ['Q'] on A).  > BANG! P1 is
     eliminated.
-> Game Over. Winner: P2
```

D Victory Case of Rational (P3)

The log in Appendix D vividly illustrates how the **Rational (P3)** agent leverages historical context information extracted by the Transformer architecture to achieve advanced strategic suppression. In this episode, P3 keenly identified the behavioral pattern of the **Aggressive Bluffer (P0)**. The latter part of the log shows that, despite P0's attempts to obfuscate by mixing true and false cards, P3 demonstrated unwavering risk assessment capabilities by initiating doubts against P0 four consecutive times. Notably, even after a failed doubt (where P0 was honest), P3 did not retreat out of fear of death; instead, based on the posterior estimation of P0's historical deception frequency, it determined that the expected utility of continuing to doubt remained positive. This dynamic **Adaptive Counter-play**, targeting the specific personality flaw of the opponent, thoroughly suppressed P0's survival space. This case directly proves that the proposed architecture has not only learned the game rules but also possesses the higher-order intelligence required for opponent modeling and dynamic decision-making in non-stationary games.


```
=== Liar's Bar Log | 2026-01-14 22:11:19.820652 ===
File: evaluate_game/game_594.txt

--- Game Start ---
P0 (AggressiveBluffer) Hand: ['K', 'K', 'Q', 'A', 'J']
P1 (AggressiveChallenger) Hand: ['Q', 'A', 'J', 'K', 'A']
P2 (Conservative) Hand: ['Joker', 'Q', 'Q', 'Q', 'K']
P3 (Rational) Hand: ['J', 'A', 'K', 'Joker', 'J']
Turn P3: Played ['J']
Turn P0: Played ['K']
Turn P1: Played ['Q']
Turn P2: DOUBT!
!!! P2 doubts P1 !!!
  > LIAR CAUGHT! P1 was lying (Played ['Q'] on A).
  > BANG! P1 is eliminated.
Turn P3: Played ['A']
Turn P0: Played ['K']
Turn P2: Played ['Q']
Turn P3: Played ['K']
Turn P0: Played ['Q']
Turn P2: DOUBT!
!!! P2 doubts P0 !!!
  > LIAR CAUGHT! P0 was lying (Played ['Q'] on A).
```

```
  > CLICK. P0 survives.
Turn P3: DOUBT!Turn P0: DOUBT!
Turn P2: Played ['Q']
Turn P3: DOUBT!
!!! P3 doubts P2 !!!
  > LIAR CAUGHT! P2 was lying (Played ['Q'] on A).
  > CLICK. P2 survives.
Turn P0: Played ['A']
Turn P2: Played ['K']
Turn P3: DOUBT!
!!! P3 doubts P2 !!!
  > LIAR CAUGHT! P2 was lying (Played ['K'] on A).
  > CLICK. P2 survives.
Turn P0: Played ['J']
Turn P2: Played ['Joker']
Turn P3: DOUBT!
!!! P3 doubts P2 !!!
  > TRUTH! P2 was honest. P3 plays Russian Roulette.
  > CLICK. P3 survives.
Turn P0: DOUBT!
Turn P2: Played ['Q']
Turn P3: DOUBT!
!!! P3 doubts P2 !!!
  > LIAR CAUGHT! P2 was lying (Played ['Q'] on A).
  > CLICK. P2 survives.
Turn P0: Played ['Joker']
Turn P2: DOUBT!
!!! P2 doubts P0 !!!
  > TRUTH! P0 was honest. P2 plays Russian Roulette.
  > BANG! P2 is eliminated.
Turn P3: DOUBT!
Turn P0: DOUBT!
Turn P3: Played ['Joker']
Turn P0: DOUBT!
!!! P0 doubts P3 !!!
  > TRUTH! P3 was honest. P0 plays Russian Roulette.
  > CLICK. P0 survives.
Turn P3: Played ['J']
Turn P0: Played ['Q']
Turn P3: DOUBT!
!!! P3 doubts P0 !!!
  > LIAR CAUGHT! P0 was lying (Played ['Q'] on A).
  > CLICK. P0 survives.
```

```
Turn P0: Played ['Joker']
Turn P3: DOUBT!!!! P3 doubts P0 !!!
  > TRUTH! P0 was honest. P3 plays Russian Roulette.
  > CLICK. P3 survives.
Turn P0: Played ['K']
Turn P3: DOUBT!
!!! P3 doubts P0 !!!
  > LIAR CAUGHT! P0 was lying (Played ['K'] on A).
  > CLICK. P0 survives.
Turn P0: Played ['Q']
Turn P3: DOUBT!
!!! P3 doubts P0 !!!
  > LIAR CAUGHT! P0 was lying (Played ['Q'] on A).
  > BANG! P0 is eliminated.
-> Game Over. Winner: P3
```