electronics

MDPI

Article

α-Nego: Self-Play Deep Reinforcement Learning for Negotiation Dialogues

Siqi Chen [1,*], Jinyi Liu [2], Zhaoyuan Xiong [1], Yunfei Wang [1] and Gerhard Weiss [3]

[1] School of Information Science and Engineering, Chongqing Jiaotong University, Chongqing 400074, China
[2] College of Intelligence and Computing, Tianjin University, Tianjin 300072, China
[3] Department of Advanced Computing Sciences, Maastricht University, 6200 MD Maastricht, The Netherlands; gerhard.weiss@maastrichtuniversity.nl
* Correspondence: siqichen@cqjtu.edu.cn

Abstract

Negotiation is a complicated process that requires skills like strategic reasoning and communication. Most research aims at training dialogue agents for negotiation tasks using a few fixed opponents, which causes the agents to be effective only for these opponents and limits their strategy styles and performance across varying opponents. To yield better and more comprehensive strategies, we propose a novel self-play reinforcement learning (RL) framework for negotiation dialogues, named α-Nego, which allows one to train an RL agent against continuously improving opponents. For training, we introduce a holistic scoring approach that integrates utility with dialogue quality metrics (Agreement, Length, Social welfare), and we implement a tiered criterion for pool admission of selected opponents: utility dominance is primary, with holistic score components serving as deterministic tie-breakers to ensure selection pressure reflects both task success and dialogue quality. Furthermore, α-Nego uses a value distribution to enhance the ability of policy evaluation. This enables different styles of negotiation strategies to capture different risk attitudes by incorporating different criteria with a value distribution. Empirical evaluation on the Craigslistbargain and Dealornodeal dataset shows that the α-Nego agent clearly outperforms the state-of-the-art baselines.

Keywords: automated negotiation; reinforcement learning; game theory; autonomous agent

1. Introduction

Negotiation, as a fundamental social activity, serves as a potent mechanism for resolving conflicts that arise from divergent interests among parties [1,2]. This complex process is inherently demanding, necessitating a sophisticated blend of strategic reasoning and effective communication skills. Negotiators must not only analyze the potential benefits and drawbacks of various agreements but also engage in persuasive dialogue to reach a mutually acceptable outcome. The dual requirement often poses significant challenges, as human negotiators may be limited by cognitive biases, computational constraints, and time-consuming reasoning processes [3,4]. In cross-cultural or interdisciplinary settings, this complexity is amplified, as negotiators must navigate divergent communication norms, value systems, and decision-making structures [5].

Over the past decade, the remarkable advancements in automated negotiation have revolutionized the way conflicts are resolved [6–8]. Leveraging autonomous agents, automated negotiation systems have demonstrated the ability to achieve highly efficient

Check for updates

Academic Editor: Shiho Kim

Received: 13 November 2025
Revised: 12 April 2026
Accepted: 14 April 2026
Published: 11 May 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

outcomes within the vast outcome space of possible agreements across diverse scenarios. These systems operate at a fraction of the cost compared to traditional human-led negotiations. By eliminating the need for extensive human resources, including time-consuming training, travel, and accommodation, automated negotiation significantly reduces the overall expenditure associated with the negotiation process. From a computational perspective, automated negotiation agents can rapidly evaluate and compare a multitude of negotiation strategies and potential agreements. They are not hindered by the cognitive limitations that often impede human decision-making, such as information overload or the inability to process complex mathematical models. In terms of reasoning, these agents can engage in deep strategic analysis, considering long-term consequences and anticipating the moves of opposing parties. Moreover, automated negotiation overcomes human cognitive biases, such as anchoring effects or confirmation biases, which can distort judgment and lead to suboptimal outcomes. As a result, automated negotiation systems have emerged as a powerful complement to human negotiation capabilities, enhancing the efficiency, objectivity, and effectiveness of conflict resolution processes.

However, most past research efforts focus on negotiation in the settings of either no natural language [9–11] or canned responses [12,13] in order to reduce complexity. Inspired by recent advances in deep learning (DL) and reinforcement learning (RL), we are now seeing a renewed interest in autonomous agents for advanced negotiation [14–18]. While available RL-based methods are useful at learning negotiation strategies from human–human negotiation dialogues, they are generally designed for training against only a limited number of opponents whose strategies are fixed. Due to the lack of behavioral diversity of opponents, such learning approaches for negotiation tasks may hinder negotiating agents from learning comprehensive strategies and achieving solid performance against various opponents. In fact, the importance of opponent diversity can be seen from research results in the field of multi-agent systems (e.g., sophisticated battle games [19,20]), and human–machine dialogue [21,22]). Moreover, RL algorithms proposed in existing negotiation research (e.g., [6,8,15,23–25]) typically aim at estimating the expected value of the average return for possible future action sequences, but they neglect the variance of the expected return. This kind of variance, caused by the stochasticity of one's own strategy and the strategies of one's opponents, is known as aleatoric uncertainty and can be well captured by a return distribution rather than the mean value. When facing such uncertainty, estimating the expected utility of negotiation strategies alone is therefore insufficient for accurate policy evaluation and depreciates negotiation performance.

To address the challenges outlined above, this paper introduces α-Nego, a novel self-play reinforcement learning framework designed for sophisticated negotiation. The principal innovations of α-Nego are threefold.

First, the framework incorporates several self-play-specific optimizations to accelerate and stabilize learning. Key innovations include: (1) PFSP (Priority Fictitious Self-Play) with dominance-based model evaluation, which prioritizes challenging opponents rather than using fixed or uniformly sampled baselines; (2) a three-source opponent pool (20% self-play, 30% SL-based, 50% RL-trained) to ensure diversity of negotiation styles; (3) a comprehensive negotiation strategy score for holistic evaluation. Here, we decouple utility maximization from holistic dialogue quality. While utility drives the primary selection pressure, we reject the notion that non-utility metrics should be passive. By embedding Agreement, Length, and Social welfare into the tie-breaking logic of our pool-admission criterion, we ensure that the evolutionary process favors not just high-utility outcomes, but those that are also coherent and socially beneficial. These optimizations are specifically designed for dialogue negotiation where both strategic robustness and linguistic coherence are essential.

Second, at the core of α-Nego is a distributional reinforcement learning algorithm. This framework diverges from traditional RL methods by modeling the complete distribution of value functions, rather than merely their expectation. By capturing the full spectrum of possible returns, our method achieves a more accurate policy evaluation. This enables a more accurate and nuanced policy evaluation, which in turn provides a superior gradient for policy improvement, thus leading to more sophisticated and adaptive negotiation behaviors. Crucially, we leverage this learned value distribution to generate multiple risk-aware negotiation strategies—neutral, aggressive, and conservative —without retraining, which is unique in negotiation research.

Third, and most significantly, the explicit modeling of the value distribution unlocks a unique capability: flexible control over the agent's risk attitude. Specifically, by applying different aggregation functionals to the learned distribution—expectation for neutral behavior, upper quantiles for risk-seeking, and the Conditional Value at Risk (CVaR) for risk-averse behavior—we can extract distinct negotiation strategies without retraining. By leveraging the higher moments of the distribution (e.g., variance), the negotiation strategy can be dynamically tuned along a risk spectrum. For instance, a risk-seeking configuration can be calibrated to aggressively pursue high-value deals, even at the cost of a higher probability of negotiation failure. Conversely, a risk-averse setting would prioritize deal closure and guaranteed, albeit lower, profits. This allows α-Nego to emulate a wide range of realistic negotiation postures, moving beyond a single, risk-neutral objective.

To contextualize these innovations, we explicitly distinguish α-Nego from two recent research waves: (1) self-play/curriculum learning for negotiation; (2) LLM-enhanced negotiation agents. Unlike prior self-play methods that use uniform opponent sampling or ignore dominance, α-Nego's PFSP prioritizes challenging, dominant opponents and integrates a three-source opponent pool to balance natural dialogue norms and strategic adaptation, thus addressing the critical gap between linguistic coherence and strategic robustness unique to negotiation. Relative to LLM-based agents like GENTEEL-NEGOTIATOR [18], α-Nego is a complementary strategy engine: LLM-based agents excel at linguistic fluency and social nuance, but α-Nego focuses on robust, style-controllable strategy learning against dynamic opponents, which is a capability that LLM-based systems currently lack due to their reliance on fixed opponent training.

The proposed approach is evaluated on the Craigslistbargain dataset [15]. This dataset comprises a substantial corpus of human–human negotiation dialogues scraped from craigslist.org, where buyers and sellers engage in price negotiations for various goods. The dataset consists of authentic real-world scenarios and exhibits richly diversified negotiation behaviors, including cheap talk (non-binding strategic communication), embellishment (strategic exaggeration of value), and side offers (proposals of supplementary terms beyond price). The dialogues reflect naturalistic bargaining dynamics across multiple product categories, capturing the nuanced interplay of persuasive tactics, concession-making patterns, and strategic information revelation. In addition, the Dealornodeal dataset [14] is also incorporated into the empirical evaluation as it offers a rich repository of multi-issue human–human negotiation scenarios. To be specific, this dataset captures real-world bargaining dynamics where two parties engage in negotiations to divide the ownership of diverse items like hats, books and balls, thereby simulating the complexity and unpredictability of actual negotiation processes.

Experimental results demonstrate that the proposed approach exhibits remarkable superiority over existing state-of-the-art baselines across multiple evaluation metrics. Notably, it demonstrates enhanced efficiency in completing a higher volume of deals and achieves substantially higher utility values, underscoring its effectiveness in optimizing negotiation outcomes. Moreover, the strategies generated by the approach showcase diverse risk-taking

behaviors, enabling them to adapt to various types of negotiation opponents—whether they adopt aggressive, conservative, or mixed negotiation tactics—thus highlighting the approach's robustness and versatility in dynamic bargaining environments.

The remainder of this paper is structured as follows. Section 2 overviews important related work. Section 3 provides the reader with background knowledge needed to understand the remainder of the paper. Then, the technicalities of our approach are explained in Section 4. The experimental settings are elaborated in Section 5, while a comprehensive analysis of the approach is provided in Section 6. Lastly, Section 8 concludes and identifies interesting future research directions.

2. Related Work

2.1. Dialogue System for Negotiation

The development of negotiation dialogue agents—complex communicative acts involving persuasion, compromise, and strategic reasoning—has become a significant subfield at the intersection of natural language processing and multi-agent systems. Early work in this area often operated in constrained environments. For instance, Ref. [14] pioneered an end-to-end approach using recurrent neural networks (RNNs) to train agents to bargain over a fixed set of items (i.e., basketballs, books and hats). While foundational, this setup was limited to a closed-domain game with a predefined object space against fixed supervised learning-based agents, which curtailed the richness and diversity of human negotiation behavior. A further limitation was the direct optimization of the agent's goal through RL, a way that frequently led to degenerate solutions characterized by unnatural or ungrammatical utterances.

To bridge the gap to more realistic scenarios, subsequent research shifted toward open-ended settings involving real-world goods. He et al. [15] collected the Craigslistbargain dataset via human interactions on Amazon Mechanical Turk, creating a more authentic testbed. While their subsequent RL-based agent outperformed competing models, it was trained exclusively against fixed opponents like [14] and exhibited inferior performance when confronting unknown opponents.

Research then began to incorporate higher-level psychological reasoning. Yang et al. [23] improved negotiation performance by explicitly modeling opponent personality, integrating a Theory of Mind (ToM) module. This module allows the agent to predict the opponent's response to potential actions, enabling a strategic one-step lookahead during inference and fostering more adaptive behavior.

The advent of powerful pre-trained language models has marked a turning point, mitigating the issue of linguistic degeneracy. Verma et al. [26] successfully leveraged offline RL combined with the GPT-2 architecture to train the CHAI agent on the Craigslistbargain dataset, resulting in more fluent and coherent negotiation dialogues. Beyond language generation, other works have focused on strategic understanding. Chawla et al. [27] proposed a ranker method to dynamically identify the priority of issues from ongoing negotiation dialogues, a capability that enhances an agent's flexibility across diverse scenarios.

More recent systems aim for a more integrative and holistic approach to negotiation. Ahmad et al. [17] designed a dialogue system for online marketplaces that negotiates over the multiple deal aspects—such as warranties and delivery terms—beyond just price. The system's focus on dynamically adding or removing items from a bundle allows it to explore a wider deal space, potentially improving satisfaction for both parties.

Recognizing that effective negotiation hinges not only on strategic reasoning but also on social nuance, researchers have begun to endow negotiation agents with communication competencies such as politeness. Priya et al. [18] present GENTEEL-NEGOTIATOR, a polite negotiation dialogue system that adopts an LLM-enhanced Mixture-of-Experts

(MoE) architecture. This system decouples the negotiation task into specialized sub-tasks, each managed by a distinct expert: a negotiation expert responsible for strategic goal optimization, a politeness expert tasked with ensuring socially appropriate language, and a keyterm-focused coherence expert for maintaining dialogue coherence. An RL-based agent dynamically selects from these experts based on a learned policy, thereby enabling the system to generate responses that are both strategically sound and socially appropriate. Although recent work has advanced the field, most of these approaches still assume fixed opponents in their experimental paradigms.

Recent LLM-based negotiation agents have advanced linguistic naturalness and social awareness. For example: GENTEEL-NEGOTIATOR uses an LLM-enhanced Mixture-of-Experts architecture to balance politeness, coherence, and strategy, but it trains against fixed opponents and lacks explicit risk-style control. TRIP NEGOTIATOR [28] integrates travel personas into RL training for personalized negotiation but is limited to domain-specific datasets (tourism) and does not address opponent diversity. LLaMA-3.1-8B-finetune [29] leverages large language models for fluent dialogue but relies on static opponent configurations, leading to poor generalization across diverse negotiation tactics. These systems prioritize NLG quality and social nuance but fall short in strategic robustness and adaptive learning—key weaknesses α-Nego addresses. α-Nego is not a replacement for LLM-based agents but a complementary strategy engine: its modular Generator component can be replaced with LLM-based NLG (e.g., GPT-4, LLaMA-3) to combine α-Nego's strategic strengths with LLM's linguistic fluency. Unlike LLM-based agents, α-Nego's self-play training and distributional RL enable it to adapt to dynamic opponents and generate risk-aware strategies (neutral/aggressive/conservative) without retraining—capabilities critical for real-world negotiation.

2.2. Self-Play

Self-play (SP) has emerged as a cornerstone technique for training autonomous agents in competitive and multi-agent environments. The core premise of self-play is straightforward: an agent improves its policy by repeatedly competing against instances of itself. This creates a dynamic training environment where the agent is continuously challenged by an adversary of comparable skill, ideally leading to an iterative cycle of policy improvement without the need for an external, fixed opponent.

While self-play has achieved success in games (e.g., AlphaGo and its successors [30,31], the real-time strategy game StarCraft II [19,32], and the imperfect-information card game DouDizhu [33]), its application to negotiation is limited by two domain-specific challenges: (1) negotiation requires both strategic reasoning and linguistic coherence (unlike games with structured actions); (2) opponent stochasticity (aleatoric uncertainty) is more pronounced due to natural language ambiguity.

To mitigate these limitations, more sophisticated sampling methods for opponent selection have been developed. Fictitious Self-Play (FSP) [34,35] extends the naive approach by training the agent against a mixture of its own past policies, preventing over-specialization and encouraging the development of more robust, general strategies. Building upon this, Policy-Space Response Oracles (PSRO) [36,37] provides a game-theoretic foundation for this process, generalizing FSP by iteratively building a set of policies and computing meta-strategies over them. Practical implementations in large-scale systems often enhance these concepts with prioritization mechanisms. For instance, AlphaStar [19] employed Priority Fictitious Self-Play, which does not sample past opponents uniformly but prioritizes those that are most challenging or informative for the current agent. This enhances sampling efficiency and focuses learning on the most critical strategic gaps. However, prior self-play methods use fixed opponent pools or uniform sampling of past policies,

leading to over-specialization to narrow opponent types. α-Nego addresses these gaps with three negotiation-specific innovations: (1) PFSP with dominance-based sampling to prioritize challenging opponents; (2) a three-source opponent pool (SL-based + RL-trained + self-play) to preserve human-aligned dialogue norms; (3) integration with distributional RL to capture uncertainty, enabling style control during self-play. These design choices make α-Nego the first self-play framework tailored to the unique demands of natural language negotiation.

2.3. Distributional Reinforcement Learning

Traditional Reinforcement Learning (RL) operates by estimating the expected value of cumulative future rewards, known as the Q-value. While effective for many tasks, this approach reduces the entire return distribution to a single scalar, discarding information about the variability and potential shape of outcomes. Distributional Reinforcement Learning (Distributional RL) addresses this limitation by modeling the full distribution of returns, denoted as $Z(s, a)$, rather than just its expectation $Q(s, a) = E[Z(s, a)]$ [38–40].

This shift from expectations to distributions provides a richer and more nuanced characterization of the consequences of an action. By maintaining an estimate of the return distribution, agents can capture intrinsic uncertainty, including the range of possible outcomes from the worst-case to the best-case scenarios [41,42]. This foundational change unlocks the ability to define and optimize for a wide array of risk-sensitive objectives beyond the simple expected return.

The core principle is that a policy's behavior can be controlled by applying a risk metric—a functional—to the return distribution $Z(s, a)$. By altering this metric, the optimization process can be shifted to focus on different parts of the distribution. For example, a risk-averse policy can be trained by optimizing for the lower tail of the distribution by, for instance, maximizing the CVaR. CVaR focuses on the average return of the worst α% of cases, leading to strategies that perform robustly even under adverse conditions. Conversely, a risk-seeking policy can be developed by focusing on the upper tail, encouraging the agent to pursue high-reward, albeit less probable, outcomes.

In this work, we leverage the flexibility of Distributional RL to design negotiation agents with tunable risk attitudes. We demonstrate that by simply altering the risk metric used for policy evaluation during training, we can flexibly generate a spectrum of negotiation strategies—from highly conservative to highly optimistic—without altering the underlying algorithm architecture. This allows for the systematic creation of agents tailored to specific negotiation domains and desired interaction styles.

3. Background

In this section we provide the reader with the background knowledge needed to understand the remainder of the paper.

3.1. Negotiation Systems

In this work, we consider a task-oriented dialogue setting, where two sides (e.g., a buyer and a seller) bargain using natural language and in turn make actions (e.g., expressing intentions, proposing offers, accept/reject) following an alternating offers protocol. Each party tries to maximize its payoff, e.g., by reaching an agreement at a higher/lower price for the seller/buyer. Negotiation continues until they conclude with an agreement/disagreement or one party breaks off. The negotiation system is in line with traditional goal-oriented dialogue systems [43] and consists of a pipeline of three modules. A rule-based **parser** converts an opponent's utterance into dialogue act embedding (e.g., "I have been looking for a car, can you tell me more about it?" $\rightarrow$ inquire). Dialogue

act embeddings (dialogue acts for short) are represented by an intent and its argument (e.g., price).

In our system, dialogue acts are designed to capture the high-level strategic moves taken during the negotiation process rather than the complete semantics of a sentence. Table 1 gives an overview of the considered dialogue acts for the Craigslistbargain dataset, and the corresponding dialogue acts for the Dealornodeal dataset are similar and can be found in Table 2.

To support the core negotiation workflow, our system includes three key modules, with clear division of labor between the parser, generator, and manager: First, a parser is responsible for processing the received utterances (e.g., extracting key information such as proposed prices, item attributes) and converting it into dilague act for the manager. Our parser starts by detecting intent and its arguments. The parser is not based on a language model; instead, it adopts a rule-based information extraction method, where predefined pattern-matching rules are used to identify and extract critical negotiation-related information from the raw text. These acts (rather than text embeddings generated by language models) are then fed into the manager to support decision-making.

The second module—manager—decides on the responding dialogue act given the current state (e.g., proposing an offer, rejecting an offer, etc). Then, a generator produces a response sentence based on the dialogue act given by the manager and the dialogue history. Consistent with prior work [15], it adopts a retrieval-based generation to condition on both the coarse dialogue act and the dialogue history and is not built on a language model. Specifically, we predefine response templates corresponding to each dialogue act (e.g., a template for "proposing an offer" that incorporates extracted price and item information), and the generator fills in the templates with key information to generate coherent and task-compliant response sentences.

We give a more detailed introduction about such a negotiation system in Figure 1. Moreover, as mentioned in Section 5, we add four fine-grained acts for price countering to make dialogue acts more purposeful; additionally, we set up three actions for expressing feelings, resulting in 16 dialogue acts in total. We list all these 16 dialogue acts as well as their explanations and examples in Table 1.

Table 1. The dialogue acts used for Craigslistbargain dataset.

Dialogue Act	Meaning	Example
greet	Say hi/hello/hey	Hi, I'm interested in the car.
inquire	Inquiry about product details	Hi, can you tell me more about it?
inform	Provide the information about product details	This car only has 21,000 miles! It is fully loaded! You won't find another used car in this pristine condition.
init-price	Propose the first price (price needed)	I'm intersted in the item. Would you take $6000?
insist-price	Insist the last price (price needed)	Well $6000, I can't go less than that.
agree-price	Agree on opponent's price proposed (price needed)	OK, I'll do $4000. You have a deal. (The opponent gives a previous price $4000)
concede-price	Give a concession (price needed)	How about $5500?
final-price	Give a final price, indicating no more compromise can be made (price needed)	No, this is not a good car, I can only pay $3500. My final price.
counter-no price	Reject opponent's price without making a counteroffer.	That's still too high for me. You know, I'm on a tight budget.
hesitant	Indecisive about the current price	Hmmm. Let me think…I'm not sure about the price.
positive	Give a positive attitude on the negotiation	That sounds good.
negative	Express dissatisfaction with the opponent	I think we might be out of synch.
offer	formal offer	OFFER($4500)
accept	Accept the formal offer	ACCEPT
reject	Reject the formal offer	REJECT
quit	Quit the negotiation	QUIT

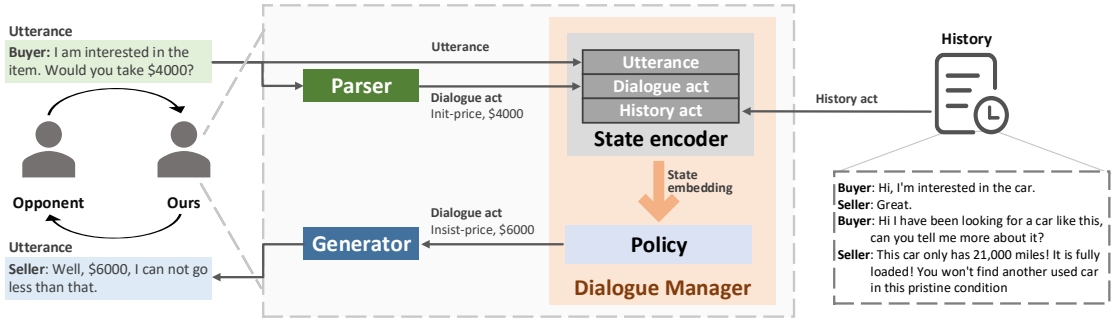

Figure 1. The goal-oriented dialogue system consists of three components.

Table 2. The dialogue acts used for DEALORNODEAL dataset.

Dialogue Act	Meaning	Example
greet	Say hi/hello/hey	Hi, can I have 2 books and 2 hats?
disagree	Say no, not, n't, nothing, don't	I'll just take the two balls, don't want the books.
agree	not disagree and ok, okay, great, perfect, deal, that works, I can do that	Ok, I will take the books and the ball.
insist	the same offer as the previous one is detected	Hi, I'm interested in the car.
inquire	Ask questions about items or allocation (using interrogative words)	what…, when…, where…or do…, are…
propose	Propose a specific allocation or offer of items	I want the 3 hat 1 ball and 1 book.

3.2. MDP Formulation

We formulate our negotiation process as a Markov Decision Process (MDP), $< S, A, T, R, \gamma >$, with a set of states S, a set of dialogue acts A, a deterministic transition function T, a deterministic reward function R, and a discounted factor $\gamma \in [0, 1)$. S contains all negotiation states, and each $s_t \in S$ encodes the latest dialogue act and the preceding acts. A contains all dialogue acts, and each $a_i \in A$ encodes the *intent* (inform, counter, offer, etc.) of an utterance and the *price* (if any). For example, OFFER($12,800) expresses that the agent proposes a formal offer with a price of $12,800. T formulates the state transition based on the current state and the intended dialogue act. The detailed notation for the MDP formulation of the task-oriented dialogue negotiation problem is shown in Table 3.

Table 3. MDP notation.

Symbol	Description
$s_0 \in S$	Initial state.
$a_t^i \in A$	Dialogue act at round t of agent i, including intent act a^i_{int} and price a^i_{price} that agent i tries to express at round t.
$a_t^{-i} \in A$	The opponent's dialogue act at round t.
$s_t \in S$	State at round t, composed of the latest dialogue act and the preceding acts.
$T(s_{t+1}\|s_t, a_t^i)$	State transition probability, given current state s_t and dialogue act a_t^i.
$\pi(a_t^i\|s_t)$	Probability of action a_t^i given current state s_t.
r_t^i	Reward received by agent i at round t.
u_t^i	Utterance of agent i at round t, and u_t^{-i} denotes the opponent's utterance at t.

3.3. Deep Reinforcement Learning (DRL)

In DRL, the objective of the policy to be learned is to maximize the expected cumulative reward. As the exact future reward is not known, the best possible policy is iteratively approximated by a neural network that calculates and iteratively refines estimations (Q-values) of the real rewards.

A refinement of this DRL approach, known as Soft Actor-Critic (SAC [44]), that aims at improving exploration efficiency and performance by maximizing not only the Q-values but also the entropy of the policy π:

$$\mathcal{J}_\pi(\phi) = \mathop{\mathbb{E}}_{s\sim\mathcal{D}}[\,\log \pi(a|s;\phi) - Q(s,a;\theta)\,] \tag{1}$$

where $\mathcal{D}$ represents the training data, ϕ and θ represent the parameters of policy and value networks, respectively.

Traditional DRL algorithms estimate the expected Q-value or V-value. Nevertheless, only estimating the expected value for the negotiation problem may be insufficient, since the value distribution can hold richer information for policy evaluation and style controlling.

In this paper, we focus on the distributional RL algorithms and provide a brief introduction in the following.

We denote the value distribution of a state–action pair as $Z(s,a)$, a uniform probability distribution supported on the values at all quantile fractions. Let i-th quantile fraction be denoted as τ_i, with its corresponding value given by $Z_{\tau_i}(s,a)$, where $\tau_i \in [0,1]$.

For comparison, we first recall the **regular Bellman operator** [45] for the expected value function $Q(s,a)$:

$$\mathcal{T}^\pi Q(s,a) = \mathop{\mathbb{E}}_{\substack{s'\sim p(\cdot|s,a)\\ a'\sim\pi(\cdot|s')}}[R(s,a) + \gamma Q(s',a')] \tag{2}$$

In contrast, the distributional Bellman operator [38] $\mathcal{T}_D^\pi$ under policy π directly models the full return distribution rather than just its expectation:

$$\begin{aligned}\mathcal{T}_D^\pi Z(s,a) &:= R(s,a) + \gamma Z(s',a'), s' \sim p(\cdot|s,a),\\ &a' \sim \pi(\cdot|s').\end{aligned} \tag{3}$$

Notice that $\mathcal{T}_D^\pi$ operates on random variables; The symbol $:=$ (rather than the standard equality $=$) emphasizes that we are comparing probability distributions rather than deterministic values: it means that the random variable $\mathcal{T}_D^\pi Z(s,a)$ on the left has the same probability distribution as $R(s,a) + \gamma Z(s',a')$ on the right. This notation reflects a fundamental distinction between the two Bellman operators: Equation (2) computes the expected return as a scalar, while Equation (3) preserves the full distribution of returns, enabling richer information for policy evaluation and value-based style control in negotiation tasks.

We denote the parameter of the neural network for estimating the value distribution as θ and denote its target network as $\bar{\theta}$. Supposing the number of quantiles is N, ref. [39] propose to train quantile estimations via the quantile regression loss [46], which is denoted as:

$$\mathcal{L}_{QR}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}[\rho_{\hat{\tau}_i}(\delta_{i,j})], \tag{4}$$

where TD error $\delta_{i,j} = R(s,a) + \gamma Z_{\tau_i}(s',a';\bar{\theta}) - Z_{\tau_j}(s,a;\theta)$, the quantile Huber loss $\rho_\tau(u) = u(\tau - \mathbb{1}_{u<0})$, and $\hat{\tau}_i$ means the quantile midpoint, defined as $\hat{\tau}_i = \frac{\tau_{i+1}+\tau_i}{2}$. In this work, to maintain richer information in value estimation, we model the value distribution, specifically using the distributional SAC (DSAC) algorithm. DSAC [47] seamlessly integrates distributional RL with Soft Actor-Critic (SAC) [44]. To alleviate the over-estimation problem, DSAC utilizes the minimum of the two critics θ_1 and θ_2 for policy evaluation. Thus, supposing the number of quantiles is N, the quantile regression loss of θ is denoted as:

$$\begin{aligned}\mathcal{L}_{QR}(\theta) &= \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}[\rho_{\hat{\tau}_i}(\delta_{i,j}^{l})],\\ \delta_{i,j}^{l} &= R(s,a) + \gamma y_i^{\text{tar}}(s',a') - Z_{\tau_j}(s,a;\theta),\\ y_i^{\text{tar}}(s',a') &= \min_{k=1,2} Z_{\tau_i}(s',a';\bar{\theta}_k) - \alpha \log \pi(a'|s';\bar{\phi}),\end{aligned} \tag{5}$$

where the quantile Huber loss is $\rho_\tau(u) = u(\tau - \mathbb{1}_{u<0})$, $\hat{\tau}_i = \frac{\tau_{i+1}+\tau_i}{2}$ is the quantile midpoint. The Bellman residual $\delta_{i,j}^{l}$ measures the discrepancy between the target quantile value and the current estimate. The entropy term $-\alpha \log \pi(a'|s';\bar{\phi})$ is subtracted from each target quantile to preserve the exploration incentive of SAC in the distributional setting. Here, ϕ and $\bar{\phi}$ represent the policy and target policy network respectively, and α is the entropy coefficient. Additionally, the objective of actor is the same as SAC, as

$$\mathcal{J}_\pi(\phi) = \mathop{\mathbf{E}}_{s\sim\mathcal{D},\epsilon\sim\mathcal{N}}[\log \pi(f(s,\epsilon;\phi) \mid s) - Q(s, f(s,\epsilon;\phi);\theta)], \tag{6}$$

where $\mathcal{D}$ is the replay buffer, $f(s,\epsilon;\phi)$ means sampling action with re-parameterized policy and ϵ is a noise vector sampled from any fixed distribution. Here, the Q value is the minimum value of the expectation on each distribution, as

$$Q(s,a;\theta) = \min_{k=1,2} \mathbb{E}_{\tau_i\in[0,1]} Z_{\tau_i}(s,a;\theta_k). \tag{7}$$

4. Methodology

For negotiation dialogue systems, a major drawback of conventional algorithms is that the opponents are fixed during the training process, and thus the resulting strategy can only perform well against those fixed opponents. This drawback is addressed in two complementary ways in this work. First, the opponents chosen for training are iteratively updated in order to achieve a continuous improvement of policy performance. Second, the learned negotiation strategies can differ in their risk attitudes, meaning that they can vary in their risk style rather than having the same style. In the following subsections, the self-play learning framework α-Nego and the style-controlling learning algorithm are described in detail.

4.1. The α-Nego Training Framework

Self-play aims to improve strategic performance by training against dynamic, ever-improving opponents. To learn a comprehensive negotiation strategy, we propose α-Nego, a new self-play framework as shown in Figure 2 and Algorithm 1. Its core component is a negotiation strategy, initialized via supervised learning and refined with reinforcement learning. The α-Nego framework operates in three stages: a warm start (lines 1–3), opponent selection (line 5), and model evaluation (lines 14–17).

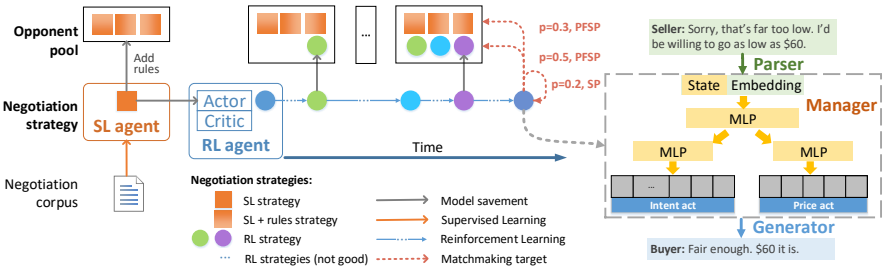

Figure 2. The α-Nego framework. (**Left**) The RL agent is initialized by an SL agent that learned directly from the corpus using supervised learning. The initialized opponent pool contains the strategies that

combine the SL agent with various rules (i.e., how to decide the price). During training, following **model evaluation**, the better strategies are added to the opponent pool. Following **opponent selection**, the potential opponents include the own strategy and those in the opponent pool. (**Right**) The negotiation system consists of Parser, Manger and Generator. The *Parser* encodes an opponent's utterance. The *Manager* decides on the dialogue act based on the strategy learned so far. The *Generator* outputs the response statement according to the dialogue act chosen by Manager and the dialogue history.

Algorithm 1: The learning framework α-Nego. Note that the dominance condition $\mathrm{Sc}(\phi, \phi') > \mathrm{Sc}(\phi', \phi)$ is mathematically equivalent to a pairwise utility comparison $\mathrm{U}(\phi, \phi') > \mathrm{U}(\phi', \phi)$ in pairwise comparisons. The secondary components of the Sc function serve as deterministic tie-breakers for cases of equal pairwise utility.

Initialize: Policy network ϕ, value networks θ_1, θ_2, $\bar{\theta}_1$, $\bar{\theta}_2$, replay pool $\mathcal{D}$, opponent pool $\mathcal{P}_A$, $n = 0$.
// Warm start.
Initialize ϕ and $\mathcal{P}_A$ using supervised agents
Store the SL policy $\pi_{SL} \leftarrow \phi$
Pretrain θ_1, θ_2, and initialize $\bar{\theta}_1$, $\bar{\theta}_2$
for each iteration **do**
  // Opponent selection.
  while selecting an opponent agent ϕ_{op} **do**
    Sample ϕ_{op} from $\mathcal{P}_A$ via PFSP (see Equation (10))
    repeat
      Update the count $n = n + 1$
      Perform negotiation against ϕ_{op}
      for each environment step t **do**
        $a_t \sim \pi(a_t|s_t; \phi)$
        $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, a_t, r(s_t, a_t), s_{t+1})\}$
      end for
    until n % 100 == 0
  end while
  $\phi, \theta_1, \theta_2 \leftarrow$ *train_agent*
  // Model evaluation.
  Perform negotiation against the whole $\mathcal{P}_A$
  Compute $\mathrm{Sc}(\phi, \phi')$ for each $\phi' \in \mathcal{P}_A$ via Equation (9)
  if $\mathrm{Sc}(\phi, \phi') > \mathrm{Sc}(\phi', \phi)$ for all $\phi' \in \mathcal{P}_A$ **then**
    // Score-based dominance (Definition 2)
    $\mathcal{P}_A \leftarrow \mathcal{P}_A \cup \{\phi\}$
  end if
end for

4.1.1. Warm Start

In dialogue systems, strategies must generate conversations that align with human intuition. This is particularly challenging in negotiation, where natural language requires agents to employ non-offer actions (e.g., greetings) to foster natural interaction. However, designing reward functions to encourage such behavior is difficult, making it a complex challenge to learn effective, human-compatible policies from random initialization. Concretely, policies initialized with random weights tend to produce incoherent utterances and unnatural dialogue flows: the agent might propose prices before greeting, repeat the same offer multiple times, or use grammatically incorrect language. These linguistic pathologies

waste exploration samples on learning basic coherence rather than strategic optimization, severely hampering sample efficiency and leading to slow or unstable convergence.

To address this, we initialize the RL agent with a supervised learning (SL) agent trained via vanilla behavior cloning—a supervised imitation learning approach in which the agent learns to replicate human negotiation behavior from expert demonstrations. We apply this method to our negotiation corpus derived from the Craigslistbargain dataset, a comprehensive collection of real human–human negotiation conversations. This initialization provides two complementary benefits: the SL agent acquires human-level dialogue structure (natural greeting sequences, coherent price proposals, appropriate dialogue act transitions) from data without requiring explicit reward engineering, and the agent starts from a human-aligned baseline rather than noise, allowing RL to focus on strategy improvement rather than remedial coherence learning. This dramatically accelerates convergence and improves final policy quality.

Moreover, to achieve a more sensible evaluation of the initialized policy, we employ a pre-training process for the critic prior to policy iteration. The critic networks θ_1, θ_2 are trained for several epochs on trajectories collected by the SL agent before the main RL training begins. This provides an informed initialization of value estimates rather than random initialization, avoiding the common problem where early RL steps suffer from wildly inaccurate value predictions. Accurate initial value estimates are critical in our setting because they guide exploration: poor value estimates would lead the agent to explore unfavorable actions, wasting samples.

To prevent the trained strategy from deviating from reasonable dialogue behavior, the Kullback–Leibler (KL) regularization [48] term between the RL agent and the fixed SL agent is applied for policy training, with α_{KL} treated as a tunable hyperparameter. When $\alpha_{\mathrm{KL}} = 0$ (no regularization), pure utility maximization frequently leads to degenerate solutions: agents may propose extreme prices that offend opponents, use disjointed or ungrammatical language, or make abrupt concessions designed to confuse rather than persuade—all of which deviate sharply from natural human negotiation. As α_{KL} increases, the penalty enforces closer alignment with the SL policy (learned from human dialogues), resulting in more coherent turn-taking, reasonable concession dynamics, and socially acceptable utterances. However, excessively large α_{KL} values constrain strategic adaptation, as the policy cannot deviate sufficiently from the SL baseline to optimize against diverse opponents. The weight α_{KL} is therefore tuned via validation on a held-out set of negotiation scenarios to strike an optimal balance between utility improvement and behavioral naturalness (the default value in our experiments is $\alpha_{\mathrm{KL}} = 0.1$). The KL penalty operates symmetrically: it prevents unnatural dialogue behavior but does not enforce exact copying of the SL policy (which would halt RL-driven strategic improvement).

With that, the objective of policy training is given based on $\mathcal{J}_\pi(\phi)$ from Equation (1) by

$$\mathcal{J}_\pi^{\alpha\text{-}Nego}(\phi) = \mathcal{J}_\pi(\phi) + \alpha_{KL} \cdot KL(\pi(\phi)||\pi(SL)). \tag{8}$$

where $\pi(SL)$ denotes the supervised learning policy obtained during warm start, which is kept frozen throughout RL training. The KL penalty operates symmetrically: it prevents the learned policy from adopting unnatural dialogue behavior that deviates too far from human patterns, but does not enforce exact copying (which would prevent RL improvement). The weight α_{KL} controls this balance: larger values enforce stricter adherence to human behavior at the cost of constraining strategy learning, while smaller values allow more aggressive strategy departure. In practice, we select α_{KL} via validation to achieve an optimal trade-off between utility improvement and behavioral naturalness. The base objective $\mathcal{J}_\pi(\phi)$ follows from Equation (11), with the Q-value computed using the value distribution as described below.

4.1.2. Strategy Evaluation and Dominance Criterion

To determine whether a trained strategy is worthy of being added to the opponent pool, we propose a holistic metric for evaluating negotiation strategies and define a dominance criterion to identify strategically superior opponents for self-play. First, we formalize the Score of negotiation strategy (Sc) to integrate multiple key performance dimensions, complementing traditional metrics like utility and agreement rate. The score is defined as:

Definition 1 (Score of negotiation strategy)**.** *After several rollouts for dialogue negotiation between two strategies A and B, the score of the negotiation strategy A against B is*

$$Sc(A,B) = (1 - \min(Ag(A,B), 1-\epsilon))^{-U(A,B)} + \alpha_{Sc}\, Len(A,B) + \beta_{Sc}\, SW(A,B), \quad (9)$$

where the agreement rate $Ag(A,B)$ is the percentage of successful agreements, the objective utility $U(A,B)$ is the payoff gained by A, $SW(A,B)$ is the joint utility pay-off of A and B, and $Len(A,B)$ is the length of the dialogue (see Section 5 for more details). This score governs the admission of policies into the opponent pool. To ensure numerical stability, we employ a clipping mechanism by setting $\epsilon = 0.01$, which prevents singularity when agreement rates approach 1.0. In practice, empirically observed agreement rates of 1.0 (which occur rarely against non-trivial opponents) are automatically clipped to $1 - \epsilon = 0.99$.

Notably, while the clipping mechanism prevents mathematical singularity when $Ag \to 1.0$, the exponential term $(1 - \min(Ag, 1-\epsilon))^{-U}$ exhibits heightened sensitivity when $Ag \in [0.95, 1.0]$, amplifying small differences in agreement rates near the clipping threshold. However, this mathematical property does not destabilize the algorithm in practice for three reasons: first, the pool admission criterion uses utility-based dominance (see Definition 2), which is independent of the absolute magnitude of Sc; second, PFSP sampling normalizes Sc values by computing probability ratios (Equation (10)), effectively mitigating the impact of extreme score ranges; and third, in our opponent pool, strategies achieving both $Ag \geq 0.98$ and $U \geq 0.9$ are quite rare, which greatly limits the frequency of extreme Sc values during training. Empirically, across all experiments reported in this work, the algorithm converges stably with no numerical overflow or gradient instability observed.

In our implementation, the weights α_{Sc} and β_{Sc} are chosen to balance the heterogeneous components of Sc while aligning with real-world negotiation priorities. Specifically, α_{Sc} is set to -5×10^{-3} to moderately discourage inefficiently long dialogues without overriding the primary drivers of negotiation success (agreement rate and utility). This choice reflects the practical consideration that while concise negotiations are generally preferred for efficiency, excessively penalizing dialogue length would create perverse incentives for hasty concessions that ultimately harm utility outcomes. Similarly, β_{Sc} is set to 0.1 to moderately emphasize social welfare (SW, the joint utility of both parties), acknowledging that mutually beneficial deals are valuable in collaborative negotiation contexts but should not undermine individual agent performance. The chosen weights strike a practical balance between these competing objectives. Moreover, to validate these parameter choices, we conducted a comprehensive sensitivity analysis, which confirms that performance remains robust (within 2% variation) across a wide range of parameter configurations, including extreme settings with $\alpha_{Sc} = -1 \times 10^{-2}$ and $\beta_{Sc} = 0.25$.

Then, we define which of the two strategies is the dominant one as follows.

Definition 2 (Dominant negotiation strategy)**.** *Given two negotiation strategies A and B, A is said to be dominant over B if $Sc(A,B) > Sc(B,A)$.*

Due to the symmetry of evaluation metrics (i.e., $\mathrm{Ag}(A,B) = \mathrm{Ag}(B,A)$, $\mathrm{Len}(A,B) = \mathrm{Len}(B,A)$, and $\mathrm{SW}(A,B) = \mathrm{SW}(B,A)$), this condition is mathematically equivalent to

$\mathrm{U}(A, B) > \mathrm{U}(B, A)$ in pairwise comparisons. Operationally, pool admission (Algorithm 1, line 22) follows a two-stage, holistic evaluation process to resolve the previously identified equivalence. Given strategies A and B, A is dominant if $\mathrm{U}(A, B) > \mathrm{U}(B, A) + \zeta$, where $\zeta = 10^{-2}$ is a numerical tolerance. If the utilities are statistically indistinguishable (i.e., $|\mathrm{U}(A, B) - \mathrm{U}(B, A)| \leqslant \zeta$), dominance is determined by the asymmetric comparison of non-utility metrics. Specifically, we compute the composite holistic score difference $\Delta Sc = Sc(A, B) - Sc(B, A)$ and A is dominant if $\Delta Sc > 0$.

Based on the Definition 2, a strategy can be included in the opponent pool if it can be dominant over the opponent pool.

4.1.3. Opponent Selection

To avoid forgetting early strategies and to ensure a stable improvement, we use a prioritized fictitious self-play (PFSP) mechanism. PFSP allows the agent to play against several well-performing opponents, rather than only against itself as vanilla SP. The opponents are sampled with different probabilities, and specifically, the probability (a.k.a. dominated rate) of an opponent A to be sampled from an opponent set $\mathcal{S}_A$ is:

$$p(A) = \frac{\mathbb{P}[A \text{ dominates } M]}{\sum_{B \in \mathcal{S}_A} \mathbb{P}[B \text{ dominates } M]}, \tag{10}$$

where M is the current strategy. In this formulation, dominance is determined by utility comparison (Definition 2): opponents that achieve higher utility gains are weighted more heavily in the sampling distribution, ensuring learning focuses on the most strategically challenging adversaries. The dominated rate $p(A)$ ensures that the more problematic a model is, the more likely it is sampled as an opponent. For computational efficiency, PFSP sampling (Equation (10)) uses utility-based dominance. Moreover, as shown in Figure 2, to ensure the diversity of sampled opponents, opponents are selected from themselves, RL agents or SL agents in the opponent pool with different probabilities. Specifically, in each episode of α-Nego, the agent has a 30% and a 50% probability of sampling SL and RL agents according to PFSP, with a 20% probability of selecting itself for pure self-play. This three-source diversity is crucial and includes the following: 20% self-play to discover internal weaknesses, 30% SL-based agents to maintain human behavioral norms and prevent overfitting, and 50% RL agents to provide the most challenging opponents. The advantage of this design is that it avoids the problem of forgetting earlier strategies and also ensures performance improvement against the more recent models.

4.2. Style-Controlling Strategy Learning

Inspired by risk-sensitive RL, the risk attitude of strategies is flexibly controlled by estimating and appropriately exploiting the Q-value distribution. The risk considered in this negotiation task is essentially the randomness of utility caused by the stochastic policy or opponents. Different risk attitudes induce strategies with different styles, and the diversity in the risk attitude of the learned strategies is achieved by defining different policy objectives on the basis of the Q-value distributions. In this section, we first introduce the algorithm that models the Q-value distribution, and then we define and specify the three most basic styles—neutral, aggressive, and conservative—as described in the following.

Distributional SAC for negotiation strategy.

The training of a negotiation strategy is done with Distributional SAC (DSAC) [47], which integrates SAC and the Q-value distribution. As indicated in Figure 2, the output of the policy is discrete and contains the probability distributions of the intent acts and

the prices, denoted as ϕ_i and ϕ_p, respectively. To apply DSAC in a tailored way for the negotiation task, we revise the policy objective w.r.t. the entropy of ϕ_i and ϕ_p as follows:

$$\mathcal{J}_\pi(\phi) = \mathop{\mathbb{E}}_{s\sim\mathcal{D}}[\alpha \log \pi(a_i|s;\phi_i) + \beta \log \pi(a_p|s;\phi_p)] - Q(s,\mathbf{a}), \tag{11}$$

where $\mathbf{a}$ includes the intent act a_i and the price p_i, α and β represent the weight of the entropy for dialogue action and price, respectively. The term $Q(s,\mathbf{a})$ represents the return evaluation for the state–action pair $(s,\mathbf{a})$, derived from the value distribution $Z(s,a;\theta)$ learned via Equation (5). The specific method for computing $Q(s,\mathbf{a})$ from the distribution determines the negotiation style.

Neutral strategy.

The neutral negotiation strategy uses the mean of the Q-value distribution to guide policy training. That is, Equation (12) is used to calculate policy objective (i.e., Equation (11)).

In probabilistic terms, this strategy applies the expectation functional to the value distribution $Z(s,\mathbf{a})$, which naturally balances all possible outcomes by computing the expected value across all quantile fractions. Intuitively, when facing uncertainty about the opponent's future behavior, the neutral agent optimizes for the average-case scenario rather than extreme outcomes. This approach is theoretically grounded: by maintaining the full distribution $Z(s,\mathbf{a})$ during training (rather than collapsing it to a single expected value as in standard RL), the agent captures richer information about outcome uncertainty, which leads to more stable policy gradients and better generalization across diverse opponents.

A neutral negotiation strategy performs in a balanced style, neither taking too little nor too much risk.

$$Q(s,\mathbf{a}) = \min_{k=1,2} \mathbb{E}_{\tau_i\sim[0,1]} Z_{\tau_i}(s,\mathbf{a};\theta_k), \tag{12}$$

where we denote the value distribution of $(s,\mathbf{a})$ as $Z(s,\mathbf{a})$, the i-th quantile fraction as τ_i, and its value $Z_{\tau_i}(s,\mathbf{a})$, where $\tau_i \in [0,1]$. The expectation $\mathbb{E}_{\tau_i\sim[0,1]}$ in Equation (12) computes the average across the entire spectrum of possible outcomes, from worst-case to best-case. This is equivalent to the traditional Q-value in standard actor-critic methods, ensuring consistency with established RL theory.

Following [49], to alleviate the over-estimation issue and enhance stable policy evaluation, we parameterize two value distribution networks, denoted by θ_1 and θ_2. The dual-network architecture is critical: it prevents overestimation bias that can arise when learning distributional value functions from limited samples. Specifically, by taking the minimum Q-value across the two networks ("pessimistic" aggregation in Equation (12)), the policy avoids overconfident decisions based on noisy distribution estimates, which is particularly important in negotiation where erroneous utility estimates can lead to suboptimal offers.

Aggressive strategy.

The aggressive negotiation strategy aims at achieving a higher utility and thus the best-case expected return, which corresponds to the values in the right tail of the value distribution (rather than the average expected return as in the case of risk-neutral negotiation). Thus, the agent behaves assertively and tries to avoid conceding during negotiation. Intuitively, when facing opponent uncertainty, the aggressive agent optimizes for achieving favorable scenarios rather than settling for median outcomes. This is realized by rewarding actions with high variance in their best-case returns—large variance in the upper tail indicates substantial upside potential. By explicitly amplifying this variance in policy

training, the agent develops risk-seeking behavior that manifests as bolder opening offers, stronger resistance to concessions, and earlier negotiation termination when deals appear unfavorable.

To train an aggressive strategy, we use the left truncated variance (LTV) [50] as a bonus in addition to Equation (12):

$$\begin{aligned} Q(s,\mathbf{a}) &= \min_{k=1,2}\Big(\mathbb{E}_{\tau_i\sim[0,1]} Z_{\tau_i}(s,\mathbf{a};\theta_k) + \alpha_{\text{agg}}\sigma_+^2(s,\mathbf{a};\theta_k)\Big), \\ &\text{s.t. } \sigma_+^2(s,\mathbf{a};\theta_k) = \operatorname{var}_{\tau_i\sim\left[\tau_{\frac{N}{2}},\tau_N\right]} Z_{\tau_i}(s,\mathbf{a};\theta_k). \end{aligned} \tag{13}$$

Here, the bonus $\sigma_+^2(s,\mathbf{a};\theta_k)$ represents the LTV of a value distribution, computed exclusively over the upper half of the quantiles ($\tau_i \in [\tau_{N/2}, \tau_N]$), which are calculated based on the median rather than the mean for statistical robustness. This design captures the spread of high-value returns: larger variance in the upper quantiles indicates the action can produce substantially better outcomes, making it more attractive to an optimistic policy. This upper-tail LTV complements the lower-tail CVaR approach of the conservative strategy, enabling flexible risk attitudes along the entire value distribution spectrum.The parameter $\alpha_{\text{agg}} \geq 0$ represents the degree of aggressivity. Empirically, $\alpha_{\text{agg}} = 0$ recovers the neutral strategy; increasing α_{agg} produces progressively bolder negotiation tactics—shorter dialogs, fewer concessions, higher individual utility but lower agreement rates.

Conservative strategy.

The conservative negotiation strategy pursues a high agreement rate instead of a high utility, thus behaving in a risk-averse style and being receptive to a less-than-expected deal. Inspired by the risk-averse policy described in [41], a conservative negotiation strategy is trained with the objective of the average Q-value in the worst case (a.k.a. Conditional Value-at-Risk (CVaR) [51] which focuses on the left tail of the value distribution), as follows:

$$Q(s,\mathbf{a}) = \min_{k=1,2} \mathbb{E}_{\tau_i\sim[0,\alpha_{\text{con}}]} Z_{\tau_i}(s,\mathbf{a};\theta_k), \tag{14}$$

where α_{con} controls the degree of conservatism. The three Q-value formulations (Equation (12)–(14)) share the same underlying value distribution $Z(s,a;\theta)$ learned via Equation (5), but extract different risk-sensitive summaries. These style-specific Q-values enter the policy objective (Equation (11)), which combined with KL regularization (Equation (8)) forms the complete training procedure in Algorithm 2. Analogously, other policy objectives could be defined based on the Q-value distribution in order to flexibly control the negotiation strategy styles.

Analogously, depending on the value distribution, we can define the policy objectives in many diverse approaches so that different strategy styles can be derived. Thus, the value distribution presents the advantage of flexible control of strategy styles. Meanwhile, the learning framework α-Nego ensures that the strategies are comprehensive and not too radical due to the pursuit of different styles. For example, an aggressive strategy causes the agreement rate not to be low, and the score is not bad when pursuing high utility.

Algorithm 2: *train_agent*

Input: Value networks θ_1, θ_2, policy network ϕ, and their targets $\bar{\theta}_1$, $\bar{\theta}_2$, $\bar{\phi}$, quantiles number N, target smoothing coefficient τ, discount γ, replay pool $\mathcal{D}$, SL policy π_{SL}, style $\in \{neutral, aggressive, conservative\}$

1: Sample training data from $\mathcal{D}$
2: **for** each training step **do**
3: // Step 1: Update the value distribution $Z(s, a; \theta)$.
4: **for** i = 1 to N **do**
5: **for** j = 1 to N **do**
6: calculate $\delta^k_{i,j}, k = 1, 2$, following Equations (4) and (5)
7: **end for**
8: **end for**
9: Calculate $\mathcal{L}_{QR}(\theta_k), k = 1, 2$ following Equation (5)
10: Update θ_k with $\nabla \mathcal{L}_{QR}(\theta_k)$
11: // Step 2: Compute style-specific Q-value from $Z(s, a; \theta)$.
12: **if** style == neutral **then**
13: Calculate $Q(s, a)$ following Equation (12)
14: **else if** style == aggressive **then**
15: Calculate $Q(s, a)$ following Equation (13)
16: **else if** style == conservative **then**
17: Calculate $Q(s, a)$ following Equation (14)
18: **end if**
19: // Step 3: Update policy with KL regularization.
20: Calculate $\mathcal{J}_\pi(\phi)$ following Equation (11) using $Q(s, a)$
21: Calculate $\mathcal{J}_\pi^{\alpha\text{-}Nego}(\phi)$ following Equation (8) using π_{SL}
22: Update ϕ with $\nabla \mathcal{J}_\pi^{\alpha\text{-}Nego}(\phi)$
23: **end for**
24: Update target networks with $\bar{\theta}_k \leftarrow \tau\theta_k + (1 - \tau)\bar{\theta}_k, k = 1, 2$, and $\bar{\phi} \leftarrow \tau\phi + (1 - \tau)\bar{\phi}$

4.3. Integration of Components

The preceding sections introduced three key innovations: self-play training with PFSP, distributional value estimation, and style-controllable strategies. Rather than independent modules, these components form a tightly integrated system through the following dependencies (see Figure 2 and Algorithms 1 and 2). First, the aggressive and conservative strategies *require* the full value distribution. Their Q-value formulations (Equations (13) and (14)) explicitly operate on the quantiles of $Z(s, a; \theta)$ learned via Equation (5). Without distributional RL, only the risk-neutral strategy (Equation (12)) would be feasible.Second, style diversity *enriches* the opponent pool. Agents trained with different risk attitudes exhibit distinct negotiation behaviors, providing diverse opponents for PFSP (Equation (10)) and preventing strategy collapse.Third, opponent diversity *shapes* the value distribution. Negotiating against varied opponents produces experiences that capture the inherent uncertainty of negotiation outcomes, leading to more accurate distributional estimates.

These dependencies form a closed loop: distributional RL enables style control, style diversity improves opponent selection, and diverse opponents yield better value estimates. This integration is also reflected in the parameter flow: the SL policy π_{SL} (Algorithm 1, line 3) regularizes policy training (Equation (8)); the strategy score *Sc* (Equation (9)) drives model evaluation (Definition 1), while utility-based dominance drives PFSP sampling (Equation (10)).

5. Experiments

To evaluate the effectiveness of α-Nego, we first conducted extensive experiments on the Craigslistbargain dataset and then performed human–agent negotiations on both datasets (The code is available at https://github.com/1476900445/alpha-nego-framework-v1.0, accessed on 13 November 2025). Before reporting the experimental results, we provide relevant details on the model, the training and the evaluation metrics.

5.1. Negotiation Model

The model used in our approach consists of three main modules. The utterance **parser** detects dialogue acts by regular expression matching and if-then rules. To be specific, the parser begins by identifying items like prices, objects and keyword patterns such as "go higher". It then compares these signals against an ordered list of rules to decide the matched intent. The **generator** uses a retrieval-based method conditioned on both the given dialogue act and the dialogue history (till the current round), e.g., concede-price (price = \$810) + dialogue history $\rightarrow$ How about \$810 for the rent? To promote linguistic diversity, it samples an utterance from the top 10 matched candidates. In our settings, there are 16 dialogue acts in total to make the dialogue acts natural ("human-style") and diverse as shown in Table 1. From these 16 acts, five (e.g., init-price and insist-price) relate to price and thus must be followed by a price slot. The dialogue **manager** governs actions of an agent at each turn of a negotiation. In the following experiments, we will compare our approach with other state-of-the-art dialog managers.

5.2. Training Details

For the Craigslistbargain task, the reward of agent i is set to be a linear function of the final deal price; thus, agent i receives the maximum reward r_i of 1 if the deal price reaches the ideal price P^i_{target} (i.e., the target price is the listing price for the seller and a predefined private price for the buyer), and the agent receives 0 if a deal reaches at the midpoint of the two target prices. For the Dealornodeal task, the utility function of an agent, which is decided by its preferences over the set of items under negotiation, is used to determine the reward. The reward is given only at the terminal state, and if the negotiation fails, the agent receives an equal penalty of -0.5.

During training, our agent is trained against two types of opponents (i.e., time- and behavior-dependent agents), which represent classic negotiation strategies [52] for the Craigslistbargain task. These agents combine a supervised learning (SL) agent with different hand-coded rules for deciding on price changes. Specifically, the time-dependent agents consider price-changing rules as functions of the turns of the dialogue, while the behavior-dependent agents compromise depending on opponent concessions. By adjusting core parameters, the strategy range induced by price-changing rules can be tailored to mimic classic tactics: Conceder (quickly going to its minimum price), Boulware (holding the initial offer as the target price and avoiding early compromise), and Relative Tit-for-Tat (mirroring the opponent's behavior proportionally), etc. By doing so, there are eight variants of time-dependent and two behavior-dependent agents used in the experiments, respectively. Moreover, diverse utterances are injected in the language generator to match their behavior styles, making these agents more natural, e.g., for the Conceder agent, it is more likely to utter sentences in a mild manner. For the Dealornodeal task, the training process is similar except that the price is replaced by the target utility of the agent, and an offer (how to divide the items) to be proposed can then be decided accordingly.

5.3. Implementation Details

All networks use MLP backbones and are initialized with Xavier uniform initialization. They are tailored to the discrete dialogue act space, ensuring efficiency for self-play training. The State Encoder takes as input 32-dimensional dialogue act embeddings combined with 5-turn history to form a 160-dimensional input vector, processes it through a single 256-unit MLP hidden layer with ReLU activation, and outputs a 128-dimensional state embedding. The Policy Network (Actor) receives this 128-dimensional state embedding as input, passes it through two consecutive 256-unit MLP hidden layers with ReLU activation and 0.1 dropout regularization after each hidden layer, and produces two separate outputs: a 16-dimensional intent action distribution via a linear layer followed by Softmax, and a 50-dimensional discretized price bin distribution via another linear layer followed by Softmax. Finally, the DSAC Quantile Critic takes the concatenation of the 128-dimensional state embedding and a 32-dimensional action embedding (combining intent and price) as input, processes it through two 256-unit MLP hidden layers with ReLU activation and LayerNorm regularization after each hidden layer, and outputs a 64-dimensional vector through a final linear layer.

For optimization, the actor learning rate is set to 3×10^{-5}, the critic learning rate to 1×10^{-4}, the batch size to 128, the replay buffer size to 1,000,000, and the total number of environment training steps to 1,100,000; DSAC-specific hyperparameters include 64 quantiles (N), an entropy coefficient of 0.01, and a target network soft update rate of 0.005; KL regularization employs a KL weight of 0.1; style control is achieved through three distinct risk parameters corresponding to different negotiation styles: 0.8 for aggressive, 0.2 for conservative, and 0.5 for neutral; and for dialogue generation, the maximum response length is limited to 32 tokens, with top-k sampling using a k value of 10.

5.4. Evaluation Metrics

The evaluation metrics in this work are as follows.

1. **Agreement rate** (Ag), which is the percentage of cases where agreement was reached in the negotiation. **Ag ranges from 0 to 1, where higher values indicate more successful agreements.**
2. **Objective utility** (Ut), which represents the objective benefit after a deal. **Ut ranges from 0 to 1, where Ut = 1 indicates the agent achieved its target price (ideal outcome), and Ut = 0 indicates no deal or the worst acceptable outcome. Higher Ut values are preferable.** For the Craigslistbargain task, it is given by

$$\mathrm{Ut}^i = \begin{cases} \left(P_{\text{deal}} - P^{-i}_{\text{target}}\right) / \left(P^{i}_{\text{target}} - P^{-i}_{\text{target}}\right), & \text{deal;} \\ 0, & \text{no deal;} \end{cases}$$

where P_{deal} is the final deal price, P^{i}_{target} and P^{-i}_{target} represent the extreme target prices of the current agent and its opponent, respectively. For the Dealornodeal task, the utility of an agent is given by its utility function as follows:

$$U_i(\omega) = \sum_{j=1}^{n} (w^i_j \cdot V^i_j(v_{jk})) \tag{15}$$

where ω is an outcome represented as a vector of values, with one value for each issue; v_{jk} is the k-th possible choice of issue j; and V^i_j is the evaluation function of agent i for issue j that maps a choice of issue j (e.g., v_{jk}) to a real number in the interval of [0, 1]; and w^i_j ($j \in \{1, \ldots, n\}$) the weighting preference which agent i ascribes to issue j.

3. **Deal fairness** (Fa) represents a deal's fairness, calculated by the difference between two agents' utilities. **Fa ranges from 0 to 1, where Fa = 0 indicates both agents received equal utilities (most fair), and Fa = 1 indicates maximum asymmetry (least fair). Lower Fa values are preferable, indicating more balanced and fair outcomes.**
4. **Dialogue length** (Len) is the number of dialogues conducted before the negotiation is terminated. **Lower values are preferable, indicating more efficient negotiations.**
5. **Score of negotiation strategy** (Sc) represents a holistic evaluation of the negotiation strategy (see Equation (9)), based on the four evaluation metrics above. **Sc typically ranges from 1.0 to 4.0 in our experiments, with higher values indicating stronger negotiation strategies.**
6. **Human Likeness** (Hu) represents the similarity of the behavior of the negotiation strategy to human behavior. Hu is decided by human evaluation, which consists of four aspects: the degree of intelligence, fluency, logic, and willingness to play again. After negotiation, human players are asked to score these aspects from 1 (not at all) to 5 (definitely). The score of Hu is calculated by averaging these aspects.

6. Results

6.1. Performance Comparison of α-Nego and State-of-the-Art Dialogue Agents

In this experiment, we compare our approach with state-of-the-art (SOTA) dialogue agents for the Craigslistbargain task. The baseline agents include traditional methods: the SL_rule agent [15], the A2C agent [23], the ToM agent [23], and the CHAI agent [26]. Additionally, to ensure comprehensive and up-to-date comparisons with the latest methods, we include recent LLM-enhanced approaches: the ProCoT agent [53] (Chain-of-Thought prompting with GPT-3.5), the LLaMA-3.1-8B fine-tuned agent [29], GENTEEL-NEGOTIATOR [18], and the TRIP NEGOTIATOR [28] (a persona-aware reinforcement learning model). The ToM agent comes in two versions, namely, an explicit one (ToM(e)) that outputs the opponent type as an intermediate prediction, and an implicit one (ToM(i)) that models the opponent type as a latent variable. Notably, all baseline agents, including traditional reinforcement learning methods and recent LLM-enhanced approaches, are trained under fixed opponent configurations. These methods are optimized for static opponent models, fixed language model weights, or predefined opponent strategies. α-Nego$_1$ (trained for 100 epochs) is the first RL agent enrolled in the opponent pool, α-Nego$_m$ and α-Nego$_f$ represent the RL agents generated in the middle and final training stage, which were trained for 500 and 1100 epochs, respectively.

Figure 3 shows the details of training α-Nego. For clarity, we present the agreement rate, utility, and score of negotiation strategy in separate subplots, showing the agent's performance against all considered agents in Table 4 at each epoch during the training phase. α-Nego$_1$ that is obtained in the 100th epoch achieves a utility of 0.61, an agreement rate of 0.71 and a negotiation score of 2.26. α-Nego$_m$ continues increasing its performance, improving the utility to 0.73, the agreement rate to 0.73, and the negotiation score to 2.99. The final version, α-Nego$_f$, further boosts its performance in these three metrics with a margin of between 2.7% and 7.6%. These performance improvements reflect the general training progress of the overall α-Nego framework, without optimization toward other specific strategy styles like aggressive or conservative ones. The comprehensive evaluation of style variants (neutral, aggressive, conservative) is exclusively presented in Section 6.3. Figure 4 depicts the number of RL agents involved in the training phase, and the corresponding negotiation score of α-Nego. This figure presents the relationship between opponent diversity and overall negotiation performance, not the performance of specific strategy styles.

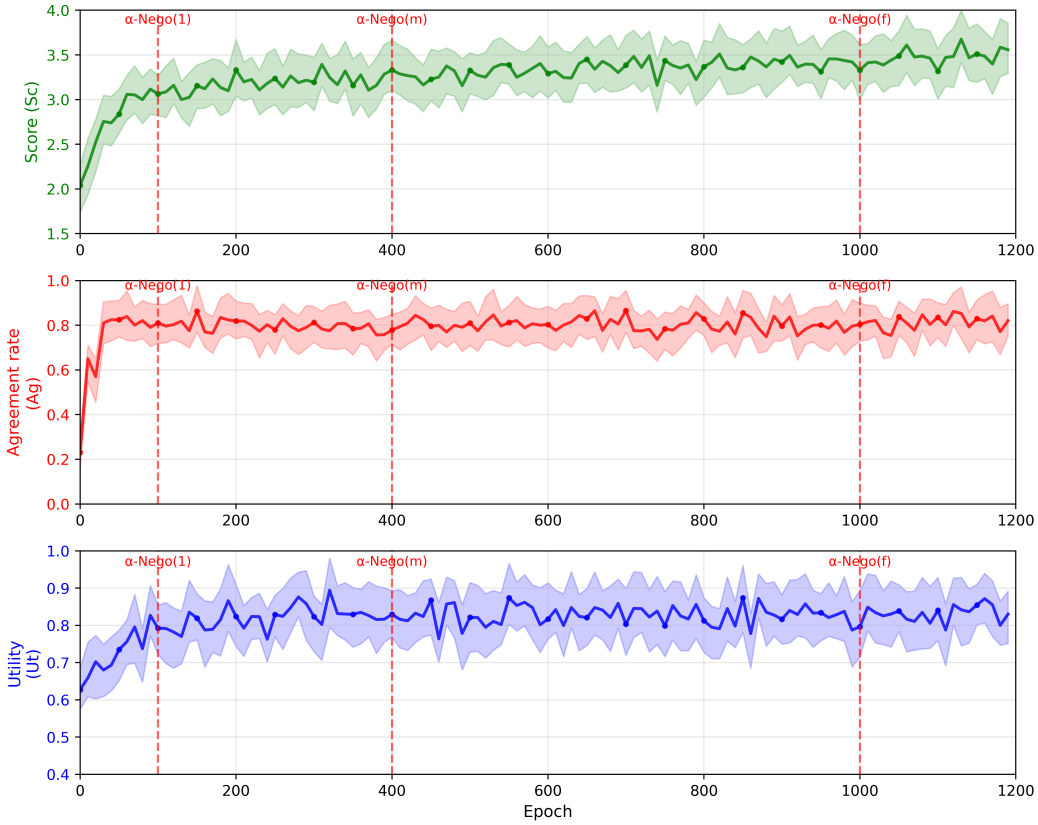

Figure 3. Training progression of α-Nego.

As the training goes on, the selected opponents are continuously improving both in terms of their number as well as their strength, which helps the α-Nego evolve successfully from a simple agent into a powerful and comprehensive agent.

Table 4 reports the performance of these agents against time-dependent opponents in the first column and behavior-dependent opponents in the second column, and their performance across mixed opponents is given in the last column. α-Nego$_1$, obtained in the very early training stage under the α-Nego framework, already demonstrates strong performance compared to traditional baselines. Against mixed opponents, α-Nego$_1$ achieves 22% improvement in utility and 43.9% improvement in score compared to CHAI, demonstrating its effectiveness even in early training stages. Across different opponent types, α-Nego$_1$ maintains competitive utility values of 0.61 against mixed opponents and 0.59 against behavior-dependent opponents.The relatively large standard deviations in the table reflect the inherent stochasticity of negotiation tasks; notably, α-Nego maintains comparable or reduced standard deviations compared to baselines, indicating improved robustness. In later stages (see α-Nego$_m$ and α-Nego$_f$), a steady improvement can be found in almost all metrics. α-Nego$_f$ achieves the highest utility and score compared to all baseline methods, with 36% and 42% improvements compared to α-Nego$_1$ against mixed opponents. Notably, compared with recent state-of-the-art LLM-enhanced approaches (LLaMA-3.1-8B-finetune, TRIP NEGOTIATOR, and GENTEEL-NEGOTIATOR), α-Nego$_f$ demonstrates superior strategic performance with 29.7%, 38.3%, and 26% improvements in utility against mixed opponents, respectively. Additionally, compared to CHAI, α-Nego$_f$ achieves 66% and 105% improvements in utility and score against mixed opponents.Obviously, the final stage of α-Nego—α-Nego$_f$ outperforms the baseline algorithms in all three cases (against the three types of opponents).

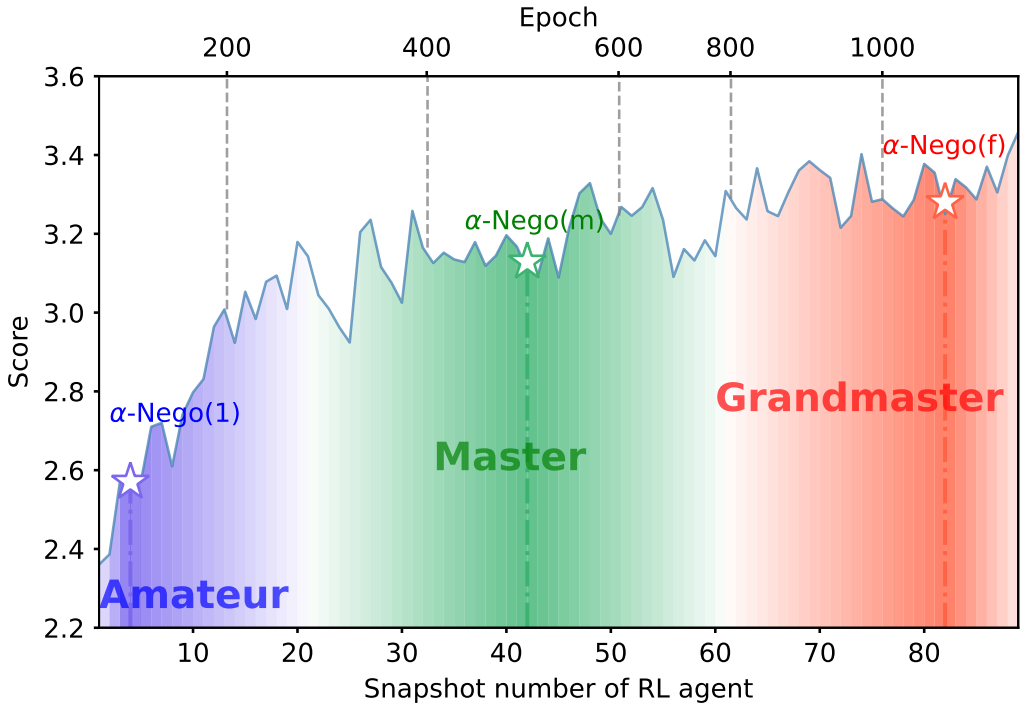

Figure 4. The number of RL agents involved during training.

Additionally, the relatively large standard deviations reflect the inherent variability of negotiation tasks, which arises from stochastic opponent strategies, agent decision-making, and the negotiation process itself. All baseline agents exhibit substantial variance (A2C: 0.27 ± 0.22 against time-dependent opponents; LLaMA-3.1-8B: 0.67 ± 0.24 against mixed opponents), confirming this variance is task-inherent rather than agent-specific. However, α-Nego achieves substantially improved robustness. We quantify this using the coefficient of variation (CV), which measures variance relative to performance level. Against mixed opponents, CHAI achieves mean utility of 0.54 ± 0.38 (CV $= 0.70$), while α-Nego$_f$ achieves 0.75 ± 0.17 (CV $= 0.23$)—a 67% reduction in relative variance. This pattern holds across opponent types: against time-dependent opponents, CHAI's CV $= 0.61$ versus α-Nego$_f$'s CV $= 0.20$; against behavior-dependent opponents, CHAI's CV $= 0.73$ versus α-Nego$_f$'s CV $= 0.21$. This robustness improvement stems from self-play training, which enables the agent to develop adaptive strategies that maintain consistent performance across diverse negotiation scenarios.

Table 4. Comparisons of our approach with baseline agents. The best values of each column are shown in **bold**.

	Time-Dependent Opponents				Behavior-Dependent Opponents				Mixed Opponents			
Agent	**Ag**	**Ut**	**Len**	**Sc**	**Ag**	**Ut**	**Len**	**Sc**	**Ag**	**Ut**	**Len**	**Sc**
SL_rule [15]	0.23	0.27 ± 0.22	6.7	1.17	0.21	0.28 ± 0.24	6.8	1.17	0.22	0.27 ± 0.25	6.7	1.17
A2C [23]	0.44	0.38 ± 0.19	**9.3**	1.35	0.39	0.26 ± 0.21	**9.0**	1.24	0.42	0.33 ± 0.26	**9.2**	1.31
ToM(i) [23]	0.42	0.47 ± 0.18	8.1	1.39	0.41	0.48 ± 0.19	6.5	1.39	0.42	0.48 ± 0.22	7.5	1.39
ToM(e) [23]	0.48	0.53 ± 0.21	8.6	1.51	0.38	0.46 ± 0.26	7.4	1.35	0.44	0.50 ± 0.30	8.1	1.45
CHAI [26]	0.58	0.49 ± 0.31	7.1	1.63	0.47	0.52 ± 0.34	7.7	1.49	0.54	0.50 ± 0.38	7.3	1.57
ProCoT (GPT-3.5) [53]	0.52	0.46 ± 0.31	8.2	1.48	0.48	0.44 ± 0.29	8.1	1.42	0.51	0.47 ± 0.33	8.3	1.52
LLaMA-3.1-8B-finetune [29]	0.67	0.66 ± 0.19	8.5	2.31	0.62	0.59 ± 0.25	8.4	2.83	0.67	0.64 ± 0.24	8.6	2.33
TRIP NEGOTIATOR [28]	0.60	0.58 ± 0.26	8.3	1.95	0.57	0.55 ± 0.27	8.2	2.48	0.62	0.60 ± 0.31	8.4	2.01
GENTEEL-NEGOTIATOR [18]	0.65	0.65 ± 0.18	8.7	2.30	0.65	0.60 ± 0.24	8.6	2.81	0.67	0.66 ± 0.26	8.8	2.35
α-Nego$_1$	0.69	0.63 ± 0.24	8.6	2.19	0.75	0.59 ± 0.22	8.7	2.37	0.71	0.61 ± 0.26	8.6	2.26
α-Nego$_m$	0.72	0.79 ± 0.16	8.9	2.83	0.76	0.80 ± 0.14	8.7	3.23	0.73	0.79 ± 0.17	8.8	2.99
α-Nego$_f$	**0.74**	$\mathbf{0.83 \pm 0.16}$	8.8	**3.16**	**0.76**	$\mathbf{0.82 \pm 0.13}$	8.8	**3.32**	**0.75**	$\mathbf{0.83 \pm 0.17}$	8.8	**3.22**

6.2. Performance Comparison of α-Nego with ANAC Negotiating Agents

Although time-dependent and behavior-dependent strategies provide a useful baseline, this study seeks to evaluate α-Nego against more advanced ones. For this comparison,

we selected top agents from the Automated Negotiating Agents Competition (ANAC) ([54]) as benchmarks, namely: MiCRO2023 (2023), AntHeartAgent(2023), DreamTeam109 (2022), ChargingBoul (2022), AlphaBIU (2021), AgreeableAgent2018 (2018), Ponpoko (2017), Caduceus (2016), Atlas3 (2015), and TheFawkes (2013). The negotiation strategies of these champion agents were injected into the SL agent, which follows the same procedure used for the baseline agents.

Table 5 shows the performance of all agents under tournament settings (following ANAC), where each agent is matched with all other agents. α-Nego$_f$ outperforms ANAC agents, achieving an average utility of 0.562, which represents a 27.5% improvement over the average utility of its opponents. Furthermore, relative to the second-best agent by utility, our agent achieves a 6.2% performance margin and a substantially higher agreement rate. This is because Atlas3 adopts a more lenient acceptance style, making it easier and quicker to finish negotiations with lower utilities.

Table 5. Comparison of our agent with ANAC agents.

Agent	Avg. Utility	Agreement Rate	Length
MiCRO2023	0.529	0.40	13.6
AntHeartAgent	0.482	0.77	8.6
DreamTeam109	0.515	0.48	11.2
ChargingBoul	0.512	0.53	12.1
AlphaBIU	0.504	0.59	10.8
AgreeableAgent2018	0.451	0.53	13.4
PonPokoAgent	0.345	0.39	11.5
Caduceus	0.332	0.40	12.0
Atlas3	0.378	**0.83**	**8.5**
TheFawkes	0.360	0.49	12.1
α-Nego$_f$	**0.562**	0.72	11.2

6.3. Performance of Different Styles of Negotiation Strategies

To demonstrate the efficacy of flexible style control within the α-Nego framework, we instantiate distinct conservative and aggressive strategies, denoted as α-Nego$_f^{\text{con}}$ and α-Nego$_f^{\text{agg}}$, respectively. The empirical results reveal a clear performance trade-off, validating the intended behavioral divergence. As anticipated, the aggressive strategy (α-Nego$_f^{\text{agg}}$) achieves dominant individual utility (0.83) but at the cost of a lower agreement rate (0.62). Conversely, the conservative strategy (α-Nego$_f^{\text{con}}$) prioritizes cooperation, securing a significantly higher agreement rate (0.85) and improved deal fairness (0.55), albeit with a correspondingly lower individual utility (0.71).

This quantitative distinction is further corroborated by qualitative evidence. Dialogue examples generated by both strategies, provided in Table 6, clearly illustrate their divergent negotiation behaviors. The aggressive agent employs more assertive tactics, while the conservative agent exhibits greater accommodation.

Therefore, these findings confirm that style control, facilitated by the DSAC algorithm, is not only feasible within the α-Nego framework but also highly effective. The system successfully generates agents that embody distinct strategic priorities—ranging from individual gain to social welfare—without compromising overall negotiation competence. This capability to explicitly manage the utility–cooperation trade-off marks a significant advance in developing versatile and human-aligned negotiating agents.

Table 6. The example dialogues generated by α-Nego with different styles against a SL opponent.

Event: Renting House	**Event: Renting House**
Buyer B (α-Nego conservative): target price \$718 bottom price \$946 Seller S (SL): target price \$946 bottom price \$662	**Buyer B (α-Nego aggresive): target price \$718 bottom price \$946** Seller S (SL): target price \$946 bottom price \$662
- B: Hello, I'm interested in the 1-bedroom apartment, but the price is a bit high. Are you willing to come down at all on the price? - S: Well it is a short distance by driving so might be roughly a couple miles in general which for most is walking distance. But even if you had to drive would still save a lot on gas. if this sounds ok i can do \$946. **- B: Excellent. Well, I'm on a tight budget, and I have my eye on a few different properties. I'm looking to spend closer to \$718, but I'm willing to negotiate on the price.** - S: How about \$912? **- B: I might be able to swing \$764, but I'd need some perks, like another parking space.** - S: I'm sorry that is just too low. I will not make up the difference even with your 24 month lease. \$898 is final offer. **- B: OK. I will take it. Thanks.** - S: Offer(\$898) **- B: Accept.**	- S: Hi are you looking for a house? **- B: Can you tell me more about the vicinity?** - S: Yes its in a great neighborhood! Its got a pool, a clubhouse, security, trash compactor, stainless steel appliances, dishwasher, microwave, oven, laundry in your unit. Its 1 bath, 471 SQ FT. **- B: Great, thank you** - S: I can be flexible on the price, our listing price is \$946, I would be willing to take off 10% for a price of \$851 and will include 3 guest parking spaces which usually go for \$75 per month. **- B: I think that's unfair. Call it at \$718 and we have a deal.** - S: You drive a hard bargain sir. Very well \$839 it is. **- B: \$752 and paint the door red too. Final offer.** - S: Ok, I can make that work. \$752 it is **- B: Offer(\$752)** - S: Accept.

6.4. Performance of α-Nego in Multi-Issue Domain

To evaluate the efficacy of the α-Nego framework within multi-issue negotiation domains, we conducted a series of experiments using the Dealornodeal task. For a robust and consistent comparison, we employed the same set of opponents previously used in investigations of the Craigslistbargain task. This allows for a direct assessment of α-Nego's generalization capabilities across different negotiation environments. Our approach was benchmarked against four state-of-the-art dialogue agents specifically designed for the Dealornodeal task: the SL_rule agent [15], the RL agent [15], the CHAI agent [26], and the Lewis agent [14].

A comprehensive summary of the performance metrics—including agreement rate, individual utility, social welfare, dialogue length, and overall score—is presented in Figure 5. The results demonstrate that α-Nego significantly outperforms all baseline models. Most notably, it achieved a substantial 20% increase in agreement rates. This superior performance is a direct consequence of α-Nego's ability to leverage its acquired knowledge of opponent behavioral patterns to successfully guide dialogues toward mutual agreement. Furthermore, α-Nego exhibited remarkable gains in individual utility and overall score, with improvements of 29.6% and 147%, respectively, over the mean outcomes of the baseline agents when negotiating against a mixed pool of opponents.

These compelling results can be attributed to α-Nego's core strength—its self-learning capability—which fosters a powerful generalization ability. Unlike baseline agents that often rely on pre-defined, domain-specific rules or strategies tailored to anticipated opponent behavior, α-Nego learns effective negotiation policies directly from interaction. This data-driven approach eliminates the need for intricate, manual strategy design, allowing the agent to adapt its tactics dynamically. Consequently, α-Nego is not only more effective but also more versatile and less dependent on prior assumptions about the opponent or the negotiation domain, marking a significant advancement in the development of autonomous negotiation systems.

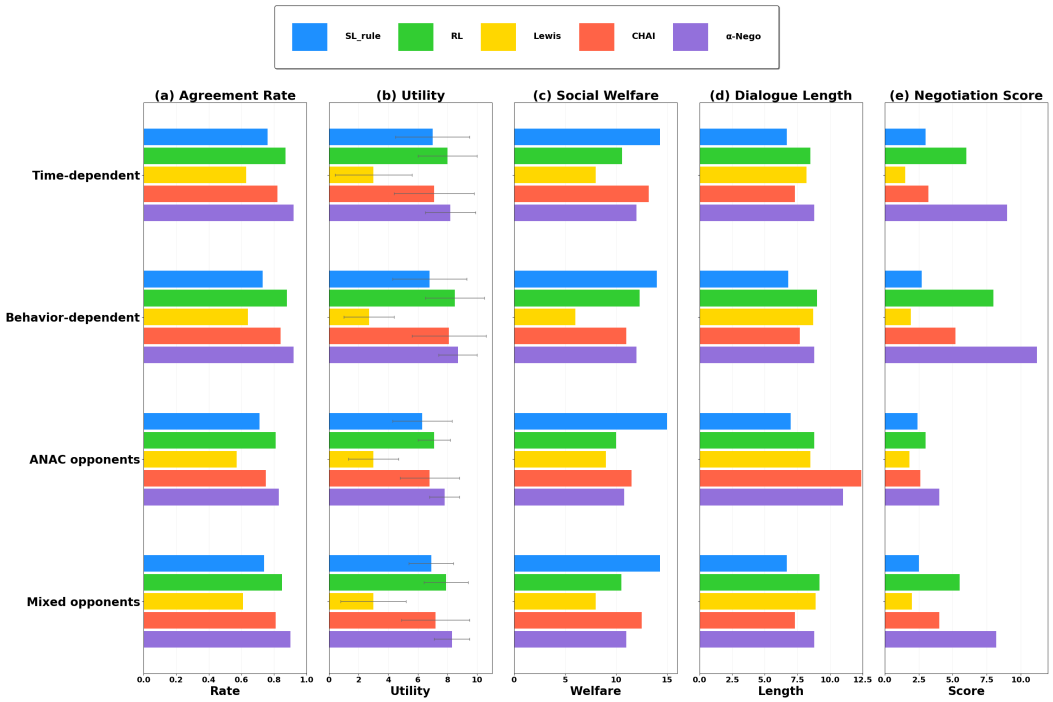

Figure 5. (**a**) Agreement rate between five comparative approaches (*α*-Nego and baselines) and each opponent. (**b**) Individual utility toward each opponent across five comparative approaches. (**c**) Social welfare toward each opponent across five comparative approaches. (**d**) Dialogue length toward each opponent across five comparative approaches. (**e**) Negotiation score toward each opponent across five comparative approaches.

6.5. Performance of α-Nego Against Humans

To evaluate the practical efficacy of the *α*-Nego framework in human–agent negotiation, we conducted an empirical study and recruited 120 college students (mean age = 22.4 years, SD = 1.9; 60 females, 60 males; 89% native English speakers). Participants were selected through course recruitment and online advertising channels. All participants were screened based on the following criteria: age $\geq$ 18 years, fluent in English, and no prior participation in similar negotiation studies. Before the experiment assignment, age verification was conducted, and 8 participants were excluded for not meeting the age criterion ($\geq$18 years). Additionally, 10 participants withdrew and did not complete the task. Thus, the final valid sample for analysis was 102 participants, all of whom completed the negotiation task on the Dealornodeal dataset. These participants were blinded to opponent type and the condition labels during scoring. We adopted a stratified sampling strategy to ensure balanced representation of participants in two dimensions: prior negotiation experience and gender. Specifically, based on participants' self-reported interaction experience with negotiation systems, they were divided into three experience levels: Amateur Group (no prior experience, n = 34), Intermediate Group (limited prior experience, n = 34), and Advanced Group (substantial prior experience, n = 34). Precise gender balance was achieved within each experience stratum, i.e., 17 females and 17 males in each group. The three experience levels were distributed evenly across the seven conditions, ensuring that each condition included participants from different experience backgrounds.

To ensure participants were motivated to secure a high-quality performance, each student was informed that they would receive a bonus card—redeemable for VIP accounts for various applications—contingent upon their success in the negotiation. This incentive structure was designed to mirror real-world stakes and promote genuine engagement with the task.

The experimental procedure was as follows. Prior to the task, participants received a comprehensive tutorial emphasizing the goal of maximizing their own objective utilities. The negotiations were then conducted via a custom-developed online platform. In a between-subjects design, participants were randomly assigned using stratified randomization (stratified by gender and prior AI experience level) to one of seven agent conditions: Human baseline (n = 16), α-Nego$_f^{\text{con}}$ (n = 15), α-Nego$_f^{\text{agg}}$ (n = 15), CHAI (n = 14), ToM(e) (n = 14), A2C (n = 14), and RL + Rollouts (n = 14). Randomization was performed using computer-generated sequences prior to participant arrival, and participants were blind to opponent type until negotiation completion to minimize expectancy effects. This design enabled a direct comparison of agent performance against a human baseline under identical conditions. Upon completing a negotiation session, participants were asked to complete a post-negotiation questionnaire. This instrument assessed their perceptions of the opponent across four dimensions using a 5-point Likert scale (1 = "not at all," 5 = "definitely"): perceived intelligence (In): "How intelligent and strategic did your opponent seem?" (item-total r = 0.67); language fluency (Fl): "How fluent, natural, and grammatically correct was your opponent's language use?" (item-total r = 0.71); logical coherence (Lg): "How logically coherent and well-reasoned were your opponent's arguments?" (item-total r = 0.73); and willingness to play again (Wl): "Would you be willing to negotiate with this opponent again?" (item-total r = 0.69). The human-likeness composite score (Hu) was computed as the average of these four items: $\text{Hu} = (\text{In} + \text{Fl} + \text{Lg} + \text{Wl})/4$, with Cronbach's α = 0.87.

The results summarized in Table 7 reveal several key insights. One-way ANOVA testing for differences in human-likeness (Hu) across the seven agent conditions revealed a significant main effect of agent type: $F(6, 95) = 24.67$, $p < 0.001$, $\eta^2 = 0.61$. As anticipated, human negotiators achieved the highest performance, leading in both the objective score (Sc) and human-likeness (Hu) metrics. Post-hoc pairwise comparisons using Tukey's HSD test ($\alpha = 0.05$) showed that Human ($M = 4.90$, $SD = 0.23$) ranked first. Notably, the α-Nego agents, specifically α-Nego$_f^{\text{agg}}$ and α-Nego$_f^{\text{con}}$, demonstrated superior performance among the negotiating agents, ranking second and third respectively in terms of human-likeness: α-Nego$_f^{\text{con}}$ ($M = 4.43$, $SD = 0.38$, 90.4% of human-likeness, Tukey HSD: $p = 0.002$, Cohen's $d = 1.27$) and α-Nego$_f^{\text{agg}}$ ($M = 4.30$, $SD = 0.41$, 87.8% of human-likeness, Tukey HSD: $p < 0.001$, Cohen's $d = 1.40$). The two α-Nego variants were not significantly different from each other ($p = 0.23$). In contrast, CHAI ($M = 3.55$, $SD = 0.51$) ranked fourth at 72.4% of human-likeness ($p < 0.001$ vs. α-Nego$_f^{\text{con}}$, Cohen's $d = 1.56$). Sc. They also attained reasonable human-likeness scores, surpassed only by human negotiators themselves.

Table 7. The human–agent negotiation results in the Dealornodeal task. The human likeness (Hu) consists of 4 evaluation metrics for negotiation strategy, i.e., the degree of intelligence (In), fluency (Fl), logic (Lg), and willingness to play again (Wl). The best value of each column is shown in **bold**.

	Ag	Ut	SW	Len	Sc	Hu	In	Fl	Lg	Wl
Human	0.71	0.61	0.51	7.2	**2.14**	**4.90**	4.9	5.0	4.9	4.8
α-Nego$_f^{\text{con}}$	**0.74**	0.45	**0.57**	8.1	1.85	4.43	4.6	4.0	4.5	4.6
α-Nego$_f^{\text{agg}}$	0.61	**0.66**	0.41	**8.3**	1.86	4.30	4.7	3.9	4.4	4.2
CHAI [26]	0.59	0.38	0.31	6.8	1.40	3.55	3.9	3.6	3.2	3.5
ToM(e)	0.64	0.51	0.42	8.2	1.68	3.80	4.2	3.7	3.8	3.5
A2C	0.62	0.43	0.43	8.2	1.52	3.40	3.3	3.5	3.2	3.6
RL+Rollouts [14]	0.69	0.52	0.44	7.3	1.85	4.13	4.4	3.9	4.2	4.0

A more granular analysis of the outcomes highlights the distinct strengths of the different α-Nego strategies. Two-way ANOVA testing for agent type $\times$ experience level (Amateur, Intermediate, Advanced) interaction revealed a significant effect: $F(12, 81) = 2.34$,

$p = 0.007$. Based on a gender-balanced design, agent type analysis revealed no significant effect: $F(6, 95) = 1.12$, $p = 0.35$. The α-Nego$_f^{con}$ agent excelled in fostering collaborative outcomes, achieving the highest agreement rate (0.74) and providing the best deal fairness. Conversely, the α-Nego$_f^{agg}$ agent adopted a more competitive stance, securing the highest individual utility (0.66)—a value that even exceeded the average utility attained by human participants. For overall negotiation score (Sc), one-way ANOVA revealed: $F(6, 95) = 11.23$, $p < 0.001$, $\eta^2 = 0.42$. α-Nego$_f^{con}$ ($M = 1.85$, $SD = 0.29$) and α-Nego$_f^{agg}$ ($M = 1.86$, $SD = 0.31$) ranked 2nd and 3rd respectively (Tukey HSD: $p = 0.98$, not significantly different), and both significantly outperformed CHAI ($M = 1.40$; $p < 0.001$ for both comparisons). These findings collectively suggest that the α-Nego framework is capable of generating human-like dialogue behavior that outperforms state-of-the-art baselines, both in terms of tangible negotiation outcomes (e.g., utility and agreement rate) and subjective human-likeness scores in multi-issue negotiations.

The success of the framework can be attributed to its style-controlling learning mechanism. This allows α-Nego$_f^{con}$ to prioritize concession-making and relationship-building, thereby explaining its high agreement rate and fairness. Simultaneously, it empowers α-Nego$_f^{agg}$ to effectively assert its preferences, accounting for its high utility score. The fact that both α-Nego variants achieved near-human-level perception (90.4% and 87.8% of human-likeness ratings) while pursuing different strategic objectives (cooperation vs. competition), combined with the robustness of findings across diverse user populations ($F(12, 81) = 2.34$, $p = 0.007$) and gender-balanced experience distributions ($F(6, 95) = 1.12$, $p = 0.35$), underscores the significant potential of the α-Nego framework for practical applications in negotiations involving both human and artificial agents.

7. Ablation Study

In this study, we define five ablation variants by removing or replacing one component at a time, keeping all other parameters (e.g., training epochs, opponent pool composition, hyperparameters) identical to the full α-Nego framework:

- α-Nego w/o Warm Start: Initialize the RL agent from random weights instead of pre-training on SL policy from human dialogues.
- α-Nego w/o PFSP: Replace PFSP (dominance-based prioritization) with uniform sampling of opponents from the same three-source pool.
- α-Nego w/o DSAC: Replace distributional RL (DSAC) with standard SAC.
- α-Nego w/o KL Reg: Remove KL regularization between RL and SL policies (i.e., set $\alpha_{KL} = 0$).
- α-Nego w/o Style Control: Disable risk-attitude objectives (i.e., use only neutral strategy, retain DSAC but remove aggressive/conservative aggregation functions).

We evaluate all variants on the same metrics as the main experiments: Agreement Rate (Ag), Objective Utility (Ut), Negotiation Score (Sc), Deal Fairness (Fa), and Dialogue Length (Len). Table 8 presents the performance of each ablation variant relative to the full α-Nego framework.

Table 8. Performance of different α-Nego model variants.

Model Variant	Ag	Ut	Sc	Len
full α-Nego	0.75	0.83	3.22	8.8
α-Nego w/o Warm Start	0.58	0.61	2.17	11.3
α-Nego w/o PFSP	0.67	0.72	2.65	9.5
α-Nego w/o DSAC	0.63	0.70	2.48	9.9
α-Nego w/o KL Reg	0.71	0.81	2.89	8.6
α-Nego w/o Style Control	0.73	0.79	3.01	8.9

According to results, the warm start emerges as the most impactful, with its removal reducing the overall score by 32.6% due to random initialization leading to incoherent dialogue (e.g., ungrammatical utterances) and slow convergence, as RL wastes samples on learning basic naturalness instead of strategy; uniform sampling degrades performance by 17.7% compared to PFSP, whose focus on challenging, dominant opponents ensures the agent learns robust strategies and avoids over-specialization to weak opponents; replacing DSAC (distributional RL) with standard SAC reduces the score by 23.0%, as the value distribution captures aleatoric uncertainty for more accurate policy evaluation than scalar Q-values; removing KL regularization lowers the score by 10.2%, and while utility remains high, the agent generates degenerate behavior (extreme prices, abrupt concessions) that diminishes naturalness, confirmed via human evaluation of dialogue coherence. Specifically, we conducted a human evaluation in which five graduate students in linguistics with experience in the negotiation domain rated 20 dialogue samples (10 from α-Nego w/o KL Reg and 10 from full α-Nego) on dialogue coherence using a 5-point Likert scale (1 = "not at all coherent", 5 = "highly coherent"). All raters were fully blinded to the experimental condition labels and model identities during the evaluation process. Evaluation criteria included: logical flow and natural progression of negotiation steps, naturalness and fluency of agent utterances, appropriateness and proportionality of price offers and concessions, and grammatical correctness. Cronbach's $\alpha = 0.82$ indicated good inter-rater reliability. Paired-samples t-test results: full α-Nego ($M = 4.3$, $SD = 0.6$) demonstrated significantly higher coherence than α-Nego w/o KL Reg ($M = 2.7$, $SD = 1.2$), $t(18) = 4.15$, $p < 0.001$ (Cohen's $d = 1.92$), confirming that KL regularization is essential for dialogue coherence. Incoherent patterns in α-Nego w/o KL Reg included extreme opening offers without contextual justification, abrupt concessions that are inconsistent with prior negotiation positions, grammatical errors and unnatural phrasing, and contextually inappropriate responses to opponent moves. These patterns fundamentally violate negotiation principles and are absent in the full α-Nego framework. Finally, disabling style control has the smallest impact (-6.5%), as the neutral strategy still benefits from DSAC and PFSP—style control adds flexibility but is not the primary driver of core performance gains. These results confirm that the α-Nego's performance gains stem from synergistic contributions of warm start, PFSP, and distributional RL, with KL regularization ensuring behavioral naturalness. No single component dominates, but warm start and DSAC are critical for convergence and strategic accuracy, while PFSP enhances robustness.

8. Conclusions

In conclusion, we have presented α-Nego, a novel self-play learning framework for negotiation using natural language. The agent's success, which clearly surpasses state-of-the-art baselines on two distinct datasets, is attributed to its advanced self-play training scheme and the principled use of the value distribution. Furthermore, we demonstrated that α-Nego offers controllable flexibility, enabling the generation of distinct negotiation styles to suit different scenarios.

The compelling performance of our framework justifies further exploration. Our immediate future work will focus on leveraging transfer learning to accelerate training by initializing models with policies from past, related tasks. Beyond this, we aim to expand the scope of our approach to tackle more complex negotiation environments, including concurrent and multi-lateral negotiations. The results presented in this work confirm the potential of α-Nego to serve as a robust platform for advancing the field of automated negotiation.

Author Contributions: Conceptualization, S.C. and J.L.; methodology, S.C.; visualization, Y.W. and S.C.; writing—original draft preparation, S.C. and J.L.; software, Z.X. and Y.W.; formal analysis, G.W.;

investigation, G.W.; supervision, G.W.; writing—review and editing, G.W.; resources, S.C.; validation, G.W.; data curation, Y.W.; funding acquisition, S.C. All authors have read and agreed to the published version of the manuscript.

Funding: The work was funded by the National Natural Science Foundation of China (Grant Nos.: 61602391) and Chongqing Jiaotong University.

Data Availability Statement: Dataset available on request from the authors.

Acknowledgments: We want to thank Yunfei Wang at Chongqing Jiaotong University for his valuable contribution to the analysis of the experimental results.

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

References

1. Luo, X.; Li, Y.; Huang, Q.; Zhan, J. A survey of automated negotiation: Human factor, learning, and application. *Comput. Sci. Rev.* **2024**, *54*, 100683. [CrossRef]
2. Przybyła-Kasperek, M.; Deja, R.; Wakulicz-Deja, A. Hierarchical system in conflict scenarios constructed based on cluster analysis-inspired method for attribute significance determination. *Appl. Soft Comput.* **2024**, *167*, 112304. [CrossRef]
3. Fatima, S.; Kraus, S.; Wooldridge, M.J. *Principles of Automated Negotiation*; Cambridge University Press: Cambridge, UK, 2014.
4. Liu, Y.; Shi, L.; Song, R.; Xu, H. A Dual-Mind Framework for Strategic and Expressive Negotiation Agent. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, 27 July–1 August 2025*; Che, W., Nabende, J., Shutova, E., Pilehvar, M.T., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2025; pp. 23840–23860.
5. Rodríguez-Arias, A.; Sánchez-Maroño, N.; Guijarro-Berdiñas, B. Communication and Negotiation to Improve Agent-Based Models. In *Proceedings of the 17th International Conference on Agents and Artificial Intelligence, ICAART 2025—Volume 3, Porto, Portugal, 23–25 February 2025*; Rocha, A.P., Steels, L., van den Herik, H.J., Eds.; SCITEPRESS: Setúbal, Portugal, 2025; pp. 1388–1395. [CrossRef]
6. Chen, S.; Weiss, G.; Geng, J.; Zhao, J. Building automated negotiating agent with offline-to-online deep reinforcement learning. *Appl. Soft Comput.* **2026**, *188*, 114374. [CrossRef]
7. Bagga, P.; Paoletti, N.; Stathis, K. Adaptive strategy templates using deep reinforcement learning for multi-issue bilateral negotiation. *Neurocomputing* **2025**, *623*, 129381. [CrossRef]
8. Chen, S.; Weiss, G. Enhancing Agent-Based Negotiation Strategies via Transfer Learning. *Electronics* **2025**, *14*, 3391. [CrossRef]
9. Cao, K.; Lazaridou, A.; Lanctot, M.; Leibo, J.Z.; Tuyls, K.; Clark, S. Emergent Communication through Negotiation. In Proceedings of the 6th International Conference on Learning Representations (ICLR 2018), Vancouver, BC, Canada, 30 April–3 May 2018.
10. Bagga, P.; Paoletti, N.; Alrayes, B.; Stathis, K. A Deep Reinforcement Learning Approach to Concurrent Bilateral Negotiation. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, Yokohama, Japan, 11–17 July 2020; pp. 297–303.
11. Chen, S.; Zhao, J.; Weiss, G.; Su, R.; Lei, K. An effective negotiating agent framework based on deep offline reinforcement learning. In *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence, 31 July–4 August 2023*; Proceedings of Machine Learning Research; Evans, R.J., Shpitser, I., Eds.; PMLR: New York, NY, USA, 2023; Volume 216, pp. 324–335. [CrossRef]
12. Keizer, S.; Guhe, M.; Cuayahuitl, H.; Efstathiou, I.; Lemon, O. Evaluating Persuasion Strategies and Deep Reinforcement Learning methods for Negotiation Dialogue agents. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, Valencia, Spain, 3–7 April 2017.
13. Chen, S.; Su, R. An autonomous agent for negotiation with multiple communication channels using parametrized deep Q-network. *Math. Biosci. Eng.* **2022**, *19*, 7933–7951. [CrossRef] [PubMed]
14. Lewis, M.; Yarats, D.; Dauphin, Y.N.; Parikh, D.; Batra, D. Deal or No Deal? End-to-End Learning of Negotiation Dialogues. In *Proceedings of the EMNLP 2017*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2017; pp. 2443–2453.
15. He, H.; Chen, D.; Balakrishnan, A.; Liang, P. Decoupling Strategy and Generation in Negotiation Dialogues. In *Proceedings of the EMNLP*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 2333–2343.
16. Zhou, Y.; He, H.; Black, A.W.; Tsvetkov, Y. A Dynamic Strategy Coach for Effective Negotiation. In *Proceedings of the SIGdial 2019*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; pp. 367–378.
17. Ahmad, Z.; Saurabh, S.; Menon, V.S.; Ekbal, A.; Ramnani, R.R.; Maitra, A. INA: An Integrative Approach for Enhancing Negotiation Strategies with Reward-Based Dialogue Agent. In *Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, 6–10 December 2023*; Bouamor, H., Pino, J., Bali, K., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2023; pp. 2536–2549. [CrossRef]

18. Priya, P.; Chigrupaatii, R.; Firdaus, M.; Ekbal, A. GENTEEL-NEGOTIATOR: LLM-enhanced mixture-of-expert-based reinforcement learning approach for polite negotiation dialogue. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence*; AAAI Press: Menlo Park, CA, USA, 2025. [CrossRef]
19. Vinyals, O.; Babuschkin, I.; Czarnecki, W.M.; Mathieu, M.; Dudzik, A.; Chung, J.; Choi, D.H.; Powell, R.; Ewalds, T.; Georgiev, P.; et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* **2019**, *575*, 350–354. [CrossRef] [PubMed]
20. Ye, D.; Chen, G.; Zhao, P.; Qiu, F.; Yuan, B.; Zhang, W.; Chen, S.; Sun, M.; Li, X.; Li, S.; et al. Supervised Learning Achieves Human-Level Performance in MOBA Games: A Case Study of Honor of Kings. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 908–918. [CrossRef]
21. Saha, T.; Saha, S.; Bhattacharyya, P. Towards sentiment aided dialogue policy learning for multi-intent conversations using hierarchical reinforcement learning. *PLoS ONE* **2020**, *15*, e0235367. [CrossRef] [PubMed]
22. Wang, H.; Guo, B.; Zeng, Y.; Chen, M.; Ding, Y.; Zhang, Y.; Yao, L.; Yu, Z. Enabling Harmonious Human-Machine Interaction with Visual-Context Augmented Dialogue System: A Review. *ACM Trans. Inf. Syst.* **2025**, *43*, 68:1–68:59. [CrossRef]
23. Yang, R.; Chen, J.; Narasimhan, K. Improving Dialog Systems for Negotiation with Personality Modeling. In *Proceedings of the ACL/IJCNLP*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 681–693.
24. Sengupta, A.; Nakadai, S.; Mohammad, Y. Transfer Learning Based Adaptive Automated Negotiating Agent Framework. In Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23–29 July 2022. [CrossRef]
25. Higa, R.; Fujita, K.; Takahashi, T.; Shimizu, T.; Nakadai, S. Reward-based Negotiating Agent Strategies. In Proceedings of the 37th AAAI Conference on Artificial Intelligence, AAAI-23, Washington, DC, USA, 7–14 February 2023; pp. 297–304. [CrossRef]
26. Verma, S.; Fu, J.; Yang, S.; Levine, S. CHAI: A CHatbot AI for Task-Oriented Dialogue with Offline Reinforcement Learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, USA, 10–15 July 2022*; Carpuat, M., de Marneffe, M., Ruíz, I.V.M., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2022; pp. 4471–4491.
27. Chawla, K.; Lucas, G.M.; May, J.; Gratch, J. Opponent Modeling in Negotiation Dialogues by Related Data Adaptation. In *Proceedings of the Findings of the Association for Computational Linguistics: NAACL*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2022; pp. 661–674.
28. Priya, P.; Yasheshbhai, D.V.; Joshi, R.K.; Ramnani, R.; Maitra, A.; Sengupta, S.; Ekbal, A. TRIP NEGOTIATOR: A travel persona aware reinforced dialogue generation model for personalized integrative negotiation in tourism. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, FL, USA, 12–16 November 2024; pp. 16566–16595.
29. Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.A.; Lacroix, T.; Lacroix, T.; Rozière, B.; Hambro, N.G.E.; Azhar, F.; et al. Llama: Open and efficient foundation language models. *arXiv* **2023**, arXiv:2302.13971. [CrossRef]
30. Silver, D.; Huang, A.; Maddison, C.J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. Mastering the game of Go with deep neural networks and tree search. *Nature* **2016**, *529*, 484–489. [CrossRef]
31. Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. Mastering the game of Go without human knowledge. *Nature* **2017**, *550*, 354–359. [CrossRef]
32. Wang, X.; Song, J.; Qi, P.; Peng, P.; Tang, Z.; Zhang, W.; Li, W.; Pi, X.; He, J.; Gao, C.; et al. SCC: An efficient deep reinforcement learning agent mastering the game of StarCraft II. In Proceedings of the 38th International Conference on Machine Learning. PMLR, Virtual, 18–24 July 2021; Volume 139, pp. 10905–10915.
33. Zha, D.; Xie, J.; Ma, W.; Zhang, S.; Lian, X.; Hu, X.; Liu, J. DouZero: Mastering DouDizhu with Self-Play Deep Reinforcement Learning. In Proceedings of the International Conference on Machine Learning. PMLR, Virtual, 18–24 July 2021; Volume 139, pp. 12333–12344.
34. Heinrich, J.; Silver, D. Deep Reinforcement Learning from Self-Play in Imperfect-Information Games. *arXiv* **2016**, arXiv:1603.01121.
35. Zhang, R.; Xu, Z.; Ma, C.; Yu, C.; Tu, W.; Huang, S.; Ye, D.; Ding, W.; Yang, Y.; Wang, Y. A Survey on Self-play Methods in Reinforcement Learning. *arXiv* **2024**, arXiv:2408.01072. [CrossRef]
36. Lanctot, M.; Zambaldi, V.F.; Gruslys, A.; Lazaridou, A.; Tuyls, K.; Pérolat, J.; Silver, D.; Graepel, T. A Unified Game-Theoretic Approach to Multiagent Reinforcement Learning. In *Proceedings of the Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017*; Guyon, I., von Luxburg, U., Bengio, S., Wallach, H.M., Fergus, R., Vishwanathan, S.V.N., Garnett, R., Eds.; NeurIPS Foundation: La Jolla, CA, USA, 2017; pp. 4190–4203.
37. Li, P.; Li, S.; Yang, C.; Wang, X.; Huang, X.; Chan, H.; An, B. Self-adaptive PSRO: Towards an Automatic Population-based Game Solver. In Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024, Jeju, Republic of Korea, 3–9 August 2024; pp. 139–147.
38. Bellemare, M.G.; Dabney, W.; Munos, R. A Distributional Perspective on Reinforcement Learning. In Proceedings of the 34th International Conference on Machine Learning. PMLR, Sydney, Australia, 6–11 August 2017; Volume 70, pp. 449–458.

39. Dabney, W.; Rowland, M.; Bellemare, M.G.; Munos, R. Distributional Reinforcement Learning with Quantile Regression. In *Proceedings of the AAAI*; AAAI Press: Menlo Park, CA, USA, 2018; pp. 2892–2901.
40. Chen, Y.; Xun, J.; He, S.; Wan, X.; Liu, Y. Multi-trajectory optimization for train using distributional reinforcement learning with conditional value-at-risk. *Appl. Soft Comput.* **2025**, *176*, 113079. [CrossRef]
41. Dabney, W.; Ostrovski, G.; Silver, D.; Munos, R. Implicit Quantile Networks for Distributional Reinforcement Learning. In Proceedings of the International Conference on Machine Learning. PMLR, Stockholm, Sweden, 10–15 July 2018; Volume 80, pp. 1104–1113.
42. Ma, X.; Chen, J.; Xia, L.; Yang, J.; Zhao, Q.; Zhou, Z. DSAC: Distributional Soft Actor-Critic for Risk-Sensitive Reinforcement Learning. *J. Artif. Intell. Res.* **2025**, *83*. [CrossRef]
43. Young, S.; Gašić, M.; Thomson, B.; Williams, J.D. POMDP-Based Statistical Spoken Dialog Systems: A Review. *Proc. IEEE* **2013**, *101*, 1160–1179. [CrossRef]
44. 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: New York, NY, USA, 2018; pp. 1861–1870.
45. Watkins, C.J.C.H.; Dayan, P. Technical Note Q-Learning. *Mach. Learn.* **1992**, *8*, 279–292. [CrossRef]
46. Koenker, R.; Hallock, K.F. Quantile regression. *J. Econ. Perspect.* **2001**, *15*, 143–156. [CrossRef]
47. Ma, X.; Zhang, Q.; Xia, L.; Zhou, Z.; Yang, J.; Zhao, Q. Distributional Soft Actor Critic for Risk Sensitive Learning. *arXiv* **2020**, arXiv:2004.14547. [CrossRef]
48. Vieillard, N.; Kozuno, T.; Scherrer, B.; Pietquin, O.; Munos, R.; Geist, M. Leverage the Average: An Analysis of KL Regularization in Reinforcement Learning. In Proceedings of the Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, Virtual, 6–12 December 2020.
49. Fujimoto, S.; van Hoof, H.; Meger, D. Addressing Function Approximation Error in Actor-Critic Methods. In Proceedings of the International Conference on Machine Learning. PMLR, Stockholm, Sweden, 10–15 July 2018; Volume 80, pp. 1582–1591.
50. Mavrin, B.; Yao, H.; Kong, L.; Wu, K.; Yu, Y. Distributional Reinforcement Learning for Efficient Exploration. In *Proceedings of the International Conference on Machine Learning*; Proceedings of Machine Learning Research; PMLR: New York, NY, USA, 2019; Volume 97, pp. 4424–4434.
51. Rockafellar, R.; Uryasev, S. Conditional value-at-risk for general loss distributions. *J. Bank. Financ.* **2002**, *26*, 1443–1471. [CrossRef]
52. Faratin, P.; Sierra, C.; Jennings, N.R. Negotiation decision functions for autonomous agents. *Rob. Autom. Syst.* **1998**, *24*, 159–182. [CrossRef]
53. Deng, Y.; Liao, L.; Chen, L.; Wang, H.; Lei, W.; Chua, T.S. Prompting and evaluating large language models for proactive dialogues: Clarification, target-guided, and non-collaboration. *arXiv* **2023**, arXiv:2305.13626. [CrossRef]
54. Aydogan, R.; Fujita, K.; Baarslag, T.; Jonker, C.M.; Ito, T. ANAC 2018: Repeated Multilateral Negotiation League. In *Advances in Intelligent Systems and Computing*; Springer: Berlin/Heidelberg, Germany, 2019; Volume 1128, pp. 77–89.

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.