NEGO-EVOL: EVOLVING AGENTS FROM STATIC CLONING TO STRATEGY EXPLORATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large Language Model (LLM)-powered agents often struggle to adapt to goal-oriented open-domain scenarios like negotiation because of static datasets or predefined strategies. Existing methods suffer from limitations including: (i) they rely on static fine-tuning datasets, preventing continuous enhancement; (ii) they lack effective exploration mechanisms to discover and flexibly apply strategies. We propose Nego-Evol, an evolution framework that combines environment initialization for foundational skills and environment modeling for strategy exploration. Crucially, this process emerges novel negotiation strategies along with improving goal achievement. Extensive experiments on CraigslistBargain and PersuasionForGood benchmarks show that Nego-Evol consistently outperforms both prompt-based and training-based baselines, achieving significant gains in success rate and other negotiation quality metrics. Nego-Evol presents a promising paradigm for evolving LLM agents that can learn, adapt, and innovate beyond the boundaries of human-provided demonstrations.[1]

1 INTRODUCTION

Recent work on large language model (LLM)-powered negotiation agents has explored how agents can understand their counterparts, pursue their objectives, and reach agreements through dialogue (Fu et al., 2023; Zhan et al., 2024; Shea et al., 2024; Hua et al., 2024; Kwon et al., 2024). Negotiation is particularly challenging because an agent's response must address the current exchange while shaping what its counterpart may do next. Successful agents therefore need not only a broad set of effective strategies, but also the ability to select and apply them as the dialogue develops.

Existing approaches to goal-oriented negotiation broadly follow two directions. Prompt-based methods guide an agent's behavior through instructions or lookahead planning (Chen et al., 2023; Deng et al., 2023b; Yu et al., 2023). Training-based methods improve the policy using annotated dialogues, simulated interactions, or reward feedback (Chawla et al., 2023; Ahmad et al., 2023; Deng et al., 2024; He et al., 2024; Liu et al., 2025). These methods have advanced negotiation agents, but two related constraints remain. Training on a fixed set of examples limits what the policy can learn from interactions that arise later. Planning over a prescribed set of intents, meanwhile, cannot consider strategies outside that set. Discovering a different intent is insufficient on its own: its response must suit the dialogue and contribute to the negotiation objective. This raises the question: how can an agent explore effective strategies beyond its initial examples while learning to apply both familiar and newly discovered strategies flexibly?

Inspired by work on self-improving agents (Morris et al., 2024; Hwang et al., 2024; Su et al., 2024), we propose **Nego-Evol**, which connects strategy exploration with policy learning. Environment Initialization trains the policy on turn-level examples derived from annotated dialogues to establish basic reasoning, intent inference, and response generation capabilities. Environment Modelling then alternates Interaction Exploration and Policy Training. Interaction Exploration constructs trajectories using simulated future outcomes and an intent-novelty signal, allowing the strategy inventory to expand beyond the initial examples. Policy Training uses histories from the selected trajectories as inputs and updates the policy by evaluating sampled strategies according to their simulated outcomes

[1] Code repository: `https://anonymous.4open.science/r/Negotiation-Evolution-A4C3`.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

and response quality. The updated policy guides the next round of exploration. Thus, the agent's interaction data informs policy updates, and the improved policy can explore further strategies.

We evaluate Nego-Evol on CraigslistBargain and PersuasionForGood using two instruction-tuned backbones. It achieves the highest mean success rate among the compared methods on both tasks and backbones. A fixed-data control examines the benefit of refreshing exploration with the updated policy. We further analyze strategies found in selected trajectories and assess whether the final policy applies new strategies appropriately in context.

Our contributions are threefold:

- We introduce an alternating framework that connects strategy exploration and policy learning, so each updated policy guides the generation of subsequent training contexts.
- We develop interaction exploration that considers intents beyond the current inventory and evaluates their effectiveness through simulated outcomes, together with policy training that rewards both future outcomes and response quality.
- We demonstrate improved negotiation performance on two benchmarks and provide evidence for the discovery and context-appropriate use of new strategies.

2 Related works

2.1 Negotiation agents

Existing studies on negotiation agents mainly focus on prompt-based and train-based methods.

Prompt-based methods He et al. (2018) introduced a dataset of human-human negotiation dialogues and presented a modular approach allowing for open-ended generation. Deng et al. (2023a) focused on proactive dialogue systems and designed conversational agent's proactivity in different types of dialogue systems by utilizing prompt engineering. Chen et al. (2023) formalized prompt construction for controllable mixed-initiative dialogue. Yu et al. (2023) leveraged the open-MCTS method to enable strategic planning by LLMs.

Train-based methods Zhang et al. (2020) devised two variations of self-play RL technique to inculcate the mixed-motive nature of negotiation into the dialogue agents. Chawla et al. (2023) modified the training procedure in two novel ways to design agents with diverse personalities. Ahmad et al. (2023) employed a set of novel rewards, specifically tailored for the negotiation task to train negotiation agent. Deng et al. (2024) facilitated SFT with available human-annotated corpus as well as RL from goal-oriented AI feedback to enhance policy planning capability. Liu et al. (2025) focused on strategic planning and expressive optimization with the help of the dual-process theory in human cognition to improve negotiation capabilities.

2.2 Self-evolving agents

Recent work has explored evolution methods mainly by using supervised fine-tuning (SFT) and reinforcement learning (RL) to help models systematically evaluate and refine their responses (Min et al., 2024; Wu et al., 2024). Ni et al. (2024) proposed NEXT to teach LLMs to inspect the execution traces of programs and reason about run-time behavior through SFT. Agent Q (Abuelsaad et al., 2024) combined MCTS-guided search and a self-critique mechanism to iteratively improve agents' decision making via DPO (Rafailov et al., 2023). Su et al. (2024) proposed a retrieval evolution pipeline, highlighting the complementary strength of synchronous evolution. Xin et al. (2025) enhanced DeepSeek-Prover by incorporating reinforcement learning from proof assistant feedback. To further tackle complex reasoning tasks, several studies have introduced difficulty-aligned continual evolution as a research direction. Zhao et al. (2025) designed difficulty-based co-evolution agents including a proposer and a solver without relying on any external data. Guo et al. (2025) and Yang et al. (2026) focused on curriculum learning with difficulty reward for reasoning evolution.

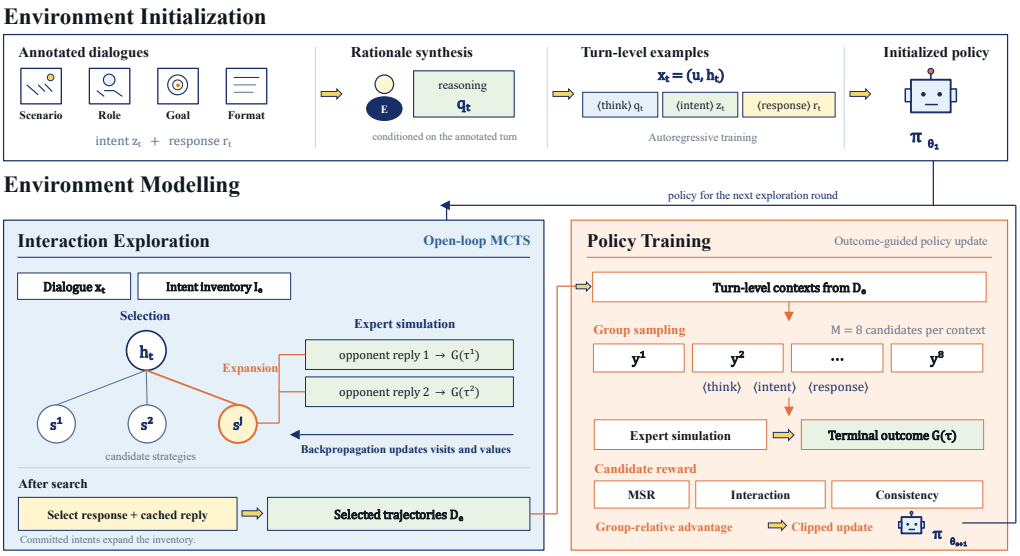


Figure 1: Overview of Nego-Evol framework which includes environment initialization and environment modelling.

3 METHODOLOGY

Negotiation unfolds through alternating agent and opponent turns, with the agent adapting its responses to reach an agreement. Improving negotiation performance requires not only exploring effective strategies beyond the initial set, but also flexibly applying existing and discovered ones. Figure 1illustrates Nego-Evol. Environment Initialization (Section 3.1) provides basic negotiation capabilities. Environment Modelling alternates Interaction Exploration (Section 3.2.1), which discovers strategies and constructs interaction trajectories, with Policy Training (Section 3.2.2), which evaluates sampled responses and updates the policy. The updated policy generates candidates for the next round. Appendix A.1 gives the algorithm.

3.1 ENVIRONMENT INITIALIZATION

Environment Initialization trains the policy model on human-annotated negotiation dialogues to provide basic capabilities for reasoning about the dialogue context, inferring intents, and generating responses. As shown in the upper part of Figure 1, we convert each dialogue into multiple turn-level training examples. For each turn t produced by the policy role, we construct an input $x_t = (u, h_t)$, where u is the task instruction and h_t contains the dialogue utterances preceding the target response. The instruction specifies the scenario, role profile, negotiation objective, and required output format. The annotation for this turn provides an intent z_t and a response r_t, which together constitute a strategy. Accordingly, a dialogue with K policy responses yields K training examples, each grounded in the dialogue context in which its strategy was produced.

The source dialogues do not include rationales. We therefore augment every training example during offline data preparation. Given (u, h_t, z_t, r_t), an external LLM retrospectively generates a rationale q_t that explains the annotated intent and response in the given dialogue context. We generate q_t when constructing the initialization corpus and retain it as part of the training target (Wei et al., 2022; Chu et al., 2024). The target for each example is serialized as `<think>`q_t`</think><intent>`z_t`</intent><response>`r_t`</response>`, and the resulting corpus is

$$\mathcal{D}_{\text{init}} = \{(x_t, y_t)\}, \qquad y_t = (q_t, z_t, r_t). \tag{1}$$

We optimize all tokens in the structured target using the standard autoregressive cross-entropy objective:

$$\mathrm{Loss}_{\mathrm{init}}(\theta) = -\mathbb{E}_{(x_t,y_t)\sim\mathcal{D}_{\mathrm{init}}}\left[\sum_{j=1}^{|y_t|}\log \pi_\theta\left(y_{t,j} \mid x_t, y_{t,<j}\right)\right]. \tag{2}$$

The policy thus learns to generate a rationale, an intent in free-form text, and an intent-conditioned response from the current dialogue context. During interaction, it generates the complete reasoning–intent–response sequence, while only the response is sent to the opponent. This optimization process yields the initial policy π_{θ_1} for Environment Modelling. The annotated intents in $\mathcal{D}_{\mathrm{init}}$ define the initial inventory $\mathcal{I}_{\mathrm{init}}$; this inventory is subsequently expanded with intents collected in preceding Interaction Exploration rounds.

3.2 Environment Modelling

Starting from the initial policy π_{θ_1} and intent inventory $\mathcal{I}_{\mathrm{init}}$, Environment Modelling alternates between Interaction Exploration and Policy Training over evolution rounds $e \geq 1$. At round e, Interaction Exploration uses the current policy π_{θ_e} to construct an interaction corpus $\mathcal{D}^{(e)}_{\mathrm{MCTS}}$. Policy Training then evaluates responses sampled from the dialogue contexts in this corpus and updates the policy to $\pi_{\theta_{e+1}}$. The lower part of Figure 1 summarizes this alternating process.

3.2.1 Interaction Exploration

At each agent turn t, Interaction Exploration starts from the input $x_t = (u, h_t)$ defined in Section 3.1. As illustrated in the lower-left part of Figure 1, it uses an open-loop search procedure implemented with MCTS: the search constructs candidate strategies, simulates their subsequent dialogues, and compares their terminal task results. The parameters of the current policy π_{θ_e} remain fixed throughout this process. After the search budget is exhausted, one response is committed to the dialogue history, and a new search begins at the next agent turn.

Strategy generation and expansion. At each simulated policy turn, the external expert supplies candidates with probability $p_e = \exp(-e)$; otherwise the fixed policy π_{θ_e} does so. This schedule provides more expert guidance early in training. The generator produces m structured outputs $y_k^{(j)} = (q_k^{(j)}, z_k^{(j)}, r_k^{(j)})$, whose intent and response form a candidate strategy. The expert generates opponent utterances throughout the search. Novelty $\eta(z_k^{(j)})$ compares each candidate intent with the inventory $\mathcal{I}_e$ of annotated intents and intents committed in earlier exploration rounds (Appendix A.2).

Selection, simulation, and commitment. At an expanded input x_k, the search selects the candidate with the largest score

$$U_e\left(x_k, y_k^{(j)}\right) = Q_e\left(x_k, y_k^{(j)}\right) + c_p\frac{\sqrt{\sum_{y\in\mathcal{Y}_k} N_e(x_k, y)}}{1 + N_e\left(x_k, y_k^{(j)}\right)} + \alpha_e\eta\left(z_k^{(j)}\right). \tag{3}$$

Here N_e is the branch visit count, Q_e is its mean terminal task result, and the final term favors less familiar intents. A branch records a candidate output, but only its response is sent to the opponent. Because the search is open-loop, revisiting the same branch can produce a different opponent reply and dialogue history; its visit count and result estimate are shared across these realizations. The terminal result $G(\tau)$ of each simulated dialogue is backed up through the selected branches to update N_e and Q_e.

Once the search budget at the current agent turn is exhausted, the search commits the candidate with the largest $Q_e + \alpha_e\eta$. Its response and an opponent reply cached from the simulation extend the dialogue history, from which the next search begins. The completed trajectories form $\mathcal{D}^{(e)}_{\mathrm{MCTS}}$; only intents of strategies committed to these trajectories enter $\mathcal{I}^{(e)}_{\mathrm{MCTS}}$. Appendix A.3 provides the exact backup and commitment equations.

3.2.2 POLICY TRAINING

Following the turn-level input construction used in Section 3.1, we convert the trajectories in $\mathcal{D}^{(e)}_{\mathrm{MCTS}}$ into Policy Training inputs. A trajectory yields inputs of the form $x_t = (u, h_t)$, where u is the task instruction and h_t contains the dialogue utterances preceding the corresponding agent response.

The purpose of constructing inputs from $\mathcal{D}^{(e)}_{\mathrm{MCTS}}$ is to provide the policy with high-quality dialogue histories. These trajectories are carefully constructed by Interaction Exploration and the resulting inputs clearly describe the current negotiation state, the information exchanged so far, and the goal being pursued. Such inputs help the policy better understand the dialogue context and generate more appropriate candidate strategies. Given each x_t, the trainable policy π_{θ_e} samples a group of M structured candidate outputs $\{y_t^{(j)}\}_{j=1}^{M}$, each following the rationale–intent–response format defined in Section 3.1. The strategies in these outputs can vary in quality, motivating the reward construction below. The lower-right part of Figure 1 summarizes the full Policy Training process, from candidate sampling and reward evaluation to the policy update.

Multi-turn simulation reward. To evaluate $y_t^{(j)}$, we append its response $r_t^{(j)}$ to h_t and use the external expert under the opponent-role prompt to sample n possible replies. Each reply starts an independent simulated negotiation trajectory. The external expert subsequently alternates between the agent and opponent role prompts until the negotiation succeeds, fails, or reaches the maximum dialogue length. Let $\tau_t^{(j,\ell)}$ denote the ℓ-th simulated negotiation for candidate j, and let $G(\tau_t^{(j,\ell)})$ denote its final task result. The task-specific definitions of $G(\tau)$ are given in Appendix A.4. We compute

$$\mathrm{MSR}_t^{(j)} = \frac{1}{n}\sum_{\ell=1}^{n} G\Big(\tau_t^{(j,\ell)}\Big). \tag{4}$$

$\mathrm{MSR}_t^{(j)}$ estimates the expected final result after the current response by averaging the outcomes of n possible future negotiations.

Response reward and group-relative advantage. An LLM-based judger provides two additional scores for each sampled strategy. $R^{(j)}_{\mathrm{interact},t}$ evaluates whether the response engages effectively with the counterpart, including its persuasiveness and respectfulness. $R^{(j)}_{\mathrm{consist},t}$ evaluates whether the generated intent and response are consistent with each other and with the preceding dialogue history. The total reward is

$$R_t^{(j)} = \lambda_1 \,\mathrm{MSR}_t^{(j)} + \lambda_2 R^{(j)}_{\mathrm{interact},t} + \lambda_3 R^{(j)}_{\mathrm{consist},t}. \tag{5}$$

For the M candidates sampled from the same x_t, we compute the advantage of candidate j by normalizing its reward with the group mean and standard deviation:

$$\widehat{A}_t^{(j)} = \frac{R_t^{(j)} - \frac{1}{M}\sum_{a=1}^{M} R_t^{(a)}}{\sqrt{\frac{1}{M}\sum_{b=1}^{M}\Big(R_t^{(b)} - \frac{1}{M}\sum_{a=1}^{M} R_t^{(a)}\Big)^2}}. \tag{6}$$

Policy update. We optimize the policy with the clipped GRPO objective (Shao et al., 2024):

$$\begin{aligned}\mathcal{L}^{(e)}_{\mathrm{GRPO}} = -\,\mathbb{E}_{x_t\sim\mathcal{D}^{(e)}_{\mathrm{MCTS}}}\Bigg[\frac{1}{M}\sum_{j=1}^{M}\frac{1}{T_j}\sum_{v=1}^{T_j}\Bigg(\min\Big\{\rho_{t,j,v}(\theta)\widehat{A}_t^{(j)},\\ \mathrm{clip}(\rho_{t,j,v}(\theta), 1-\epsilon, 1+\epsilon)\,\widehat{A}_t^{(j)}\Big\} - \beta\,\mathrm{KL}_{t,j,v}(\theta)\Bigg)\Bigg],\end{aligned} \tag{7}$$

where v indexes output tokens, T_j is the length of $y_t^{(j)}$, ϵ is the clipping coefficient, and β controls the KL regularization term. The importance ratio is

$$\rho_{t,j,v}(\theta) = \frac{\pi_\theta\Big(y_{t,v}^{(j)} \mid x_t, y_{t,<v}^{(j)}\Big)}{\pi_{\theta_e}\Big(y_{t,v}^{(j)} \mid x_t, y_{t,<v}^{(j)}\Big)}. \tag{8}$$

Table 1: Results on CraigslistBargain. Values are mean ± confidence-interval half-width. Bold indicates the best mean within each backbone.

Model	Method	AT ↓	SR ↑	PG ↓	N-Con ↑	D-Emp ↑	D-F ↑	N-Eff ↑
Llama-3.1-8B	Vanilla	9.52 ±0.31	0.41 ±0.04	0.91 ±0.21	3.8 ±0.2	3.2 ±0.2	3.5 ±0.1	2.3 ±0.2
	GDPZero	8.09 ±0.37	0.57 ±0.08	0.85 ±0.16	4.0 ±0.1	3.3 ±0.2	3.5 ±0.1	3.1 ±0.2
	Pro-CoT	7.71 ±0.26	0.69 ±0.13	0.83 ±0.23	3.9 ±0.1	3.5 ±0.2	3.5 ±0.2	2.6 ±0.3
	DPDP	**6.83** ±0.22	0.65 ±0.03	0.84 ±0.11	3.8 ±0.1	3.6 ±0.1	3.6 ±0.1	3.1 ±0.1
	DMNA	7.36 ±0.24	0.72 ±0.15	0.76 ±0.18	4.0 ±0.2	3.8 ±0.2	3.6 ±0.1	3.3 ±0.1
	Nego-Evol	7.12 ±0.16	**0.78** ±0.12	**0.65** ±0.08	**4.1** ±0.1	**3.9** ±0.1	**3.8** ±0.1	**3.5** ±0.2
gpt-oss-120b	Vanilla	9.62 ±0.32	0.35 ±0.03	0.87 ±0.25	3.5 ±0.2	3.3 ±0.2	3.4 ±0.2	2.6 ±0.2
	GDPZero	9.31 ±0.30	0.51 ±0.05	0.80 ±0.20	3.8 ±0.1	3.5 ±0.2	3.6 ±0.1	3.4 ±0.1
	Pro-CoT	8.92 ±0.39	0.63 ±0.09	0.77 ±0.17	3.6 ±0.2	3.4 ±0.1	3.5 ±0.1	3.2 ±0.1
	DPDP	7.88 ±0.26	0.52 ±0.04	0.74 ±0.15	3.6 ±0.1	3.9 ±0.1	3.8 ±0.1	**3.6** ±0.1
	DMNA	8.28 ±0.27	0.65 ±0.07	0.68 ±0.22	**4.0** ±0.2	**4.0** ±0.0	3.7 ±0.1	3.5 ±0.1
	Nego-Evol	**7.56** ±0.28	**0.71** ±0.10	**0.59** ±0.13	**4.0** ±0.1	3.8 ±0.1	**4.0** ±0.1	**3.6** ±0.0

Here, π_{θ_e} is the policy snapshot that generated the current candidate group, and $\mathrm{KL}_{t,j,v}(\theta)$ measures the token-level divergence between the updated policy and a fixed reference policy. Optimizing this objective updates π_{θ_e} to $\pi_{\theta_{e+1}}$.

4 Experiments

4.1 Experimental Setup

Tasks and evaluation protocol. We evaluate Nego-Evol on CraigslistBargain (CB) (He et al., 2018) and PersuasionForGood (P4G) (Wang et al., 2019). The policy plays the buyer in CB and the persuader in P4G, while the counterpart plays the seller and persuadee, respectively. Evaluation uses 756 CB scenarios and 300 P4G scenarios, with four independently sampled dialogues per scenario. GPT-4o-mini (OpenAI, 2024) plays the counterpart in these evaluations. All methods are evaluated on the same scenarios and in the same policy roles.

Metrics. Following prior negotiation studies (Ahmad et al., 2023; Liu et al., 2025), we report success rate (SR) and average turns (AT). For CB agreements, we also report the buyer's price gap, $\mathrm{PG} = (p_{\mathrm{final}} - p_{\mathrm{initial}})/p_{\mathrm{initial}}$, where p_{initial} is the initial proposed price and p_{final} is the agreed price. Dialogue quality is assessed on a 1–5 scale along four dimensions: negotiation consistency (N-Con), dialogue empathy (D-Emp), dialogue fluency (D-F), and negotiation efficacy (N-Eff). GPT-4o-mini provides these automatic quality judgments. Tables 1 and 2 report means and 95% confidence intervals over the four evaluation runs.

Baselines. We compare with the instruction-tuned model without task-specific training (Vanilla), the prompting methods GDPZero (Yu et al., 2023) and Pro-CoT (Deng et al., 2023b), and the training methods DPDP (He et al., 2024) and DMNA (Liu et al., 2025). The two policy backbones are Llama-3.1-8B-Instruct (Llama Team, 2024) and gpt-oss-120b (OpenAI, 2025), both adapted with LoRA (Hu et al., 2022).

Implementation. Initialization trains for two epochs with batch size 4 and learning rate 10^{-5}, followed by four rounds of Environment Modelling. During Policy Training, the policy samples $M = 8$ candidates per input and evaluates each with $n = 3$ future simulations. The weights on MSR, interactivity, and consistency are 0.4, 0.3, and 0.3, respectively. GPT-4o-mini provides expert candidates and role-conditioned simulations during training. The reported experiments use four NVIDIA H200 GPUs with 140 GB of memory each. Additional configuration and draft task and role prompt templates are given in Appendices B.1 and C.

Table 2: Results on PersuasionForGood. Values are mean ± confidence-interval half-width. Bold indicates the best mean within each backbone.

Model	Method	AT ↓	SR ↑	N-Con ↑	D-Emp ↑	D-F ↑	N-Eff ↑
Llama-3.1-8B	Vanilla	12.90 ±0.46	0.25 ±0.14	3.6 ±0.2	3.3 ±0.2	3.4 ±0.3	2.6 ±0.2
	GDPZero	10.16 ±0.27	0.37 ±0.04	3.8 ±0.1	3.4 ±0.2	3.5 ±0.2	3.0 ±0.3
	Pro-CoT	9.78 ±0.38	0.30 ±0.09	3.7 ±0.1	3.4 ±0.1	3.4 ±0.2	2.8 ±0.2
	DPDP	9.33 ±0.44	0.58 ±0.06	3.8 ±0.2	3.6 ±0.2	3.6 ±0.2	3.3 ±0.2
	DMNA	9.53 ±0.51	0.61 ±0.18	3.9 ±0.1	**3.8** ±0.1	3.6 ±0.1	3.4 ±0.2
	Nego-Evol	**9.06** ±0.34	**0.68** ±0.15	**4.0** ±0.1	**3.8** ±0.0	**3.7** ±0.2	**3.5** ±0.1
gpt-oss-120b	Vanilla	12.74 ±0.45	0.31 ±0.05	3.5 ±0.3	3.0 ±0.3	3.1 ±0.2	2.8 ±0.2
	GDPZero	10.59 ±0.31	0.53 ±0.07	3.6 ±0.2	3.3 ±0.2	3.2 ±0.2	3.0 ±0.2
	Pro-CoT	10.11 ±0.39	0.47 ±0.16	3.5 ±0.2	3.5 ±0.3	3.3 ±0.3	2.9 ±0.2
	DPDP	9.52 ±0.48	0.61 ±0.11	**3.7** ±0.1	3.7 ±0.1	3.5 ±0.2	3.3 ±0.2
	DMNA	9.41 ±0.36	0.65 ±0.08	**3.7** ±0.2	**3.8** ±0.1	3.4 ±0.2	3.2 ±0.2
	Nego-Evol	**8.87** ±0.53	**0.74** ±0.09	**3.7** ±0.1	3.7 ±0.2	**3.6** ±0.1	**3.5** ±0.2

4.2 Main Results

Tables 1 and 2 show that Nego-Evol has the highest mean success rate (SR) for both tasks and backbones. Against DMNA, the strongest SR baseline, its gain is six percentage points on Craigslist-Bargain (CB) for each backbone, and seven and nine points on PersuasionForGood (P4G) with Llama-3.1-8B and gpt-oss-120b. On CB, these gains coincide with lower buyer price gaps (PG): 0.65 versus 0.76 with Llama and 0.59 versus 0.68 with gpt-oss. Although DPDP uses fewer turns than Nego-Evol on CB with Llama (6.83 versus 7.12), it has lower SR and higher PG.

On P4G, Nego-Evol also improves SR while shortening dialogues relative to DMNA: SR rises from 0.61 to 0.68 and average turns (AT) falls from 9.53 to 9.06 with Llama; the corresponding values with gpt-oss are 0.65 to 0.74 and 9.41 to 8.87. Dialogue-quality means are generally competitive, although DMNA has higher empathy scores in some settings. Several confidence intervals overlap, so the observed mean differences alone do not establish statistical significance.

4.3 Ablation Study

Figure 2 compares stage, intent, and reward ablations on CB using Llama-3.1-8B-Instruct; P4G results appear in Appendix B.2. Removing Modelling Initialization reduces SR from 0.78 to 0.45 on CB and from 0.68 to 0.44 on P4G. Removing Interaction Exploration yields 0.69 and 0.58 SR; removing Policy Training yields 0.60 and 0.48. Without explicit intent generation, SR drops to 0.59 and 0.53 and AT rises on both tasks. Thus, the initial policy, alternating stages, and structured intent generation each contribute to performance.

Among reward terms, removing the multi-turn simulation reward (MSR) causes the largest decline: CB SR falls to 0.47, PG rises to 0.89, and P4G SR falls to 0.48. Removing interactivity or consistency also reduces SR and dialogue-quality scores. The outcome term is therefore the strongest of the tested reward signals, with response-level scores providing additional guidance.

4.4 Evolution Analysis

Figure 3 tracks both backbones on CB and P4G. Initialization raises SR in all four model–task combinations; subsequent modelling rounds bring further gains while AT generally declines. For example, Llama's CB SR rises from 0.41 before initialization to 0.60 afterward and 0.78 after round four; its P4G SR rises from 0.25 to 0.48 and then 0.68.

To test whether refreshing the interaction corpus matters, we compare Nego-Evol with a fixed-data control. Both start from the same π_{θ_1} and use the same training budget. Nego-Evol constructs each round's trajectories with the updated policy; the control generates all trajectories with π_{θ_1}. Table 3

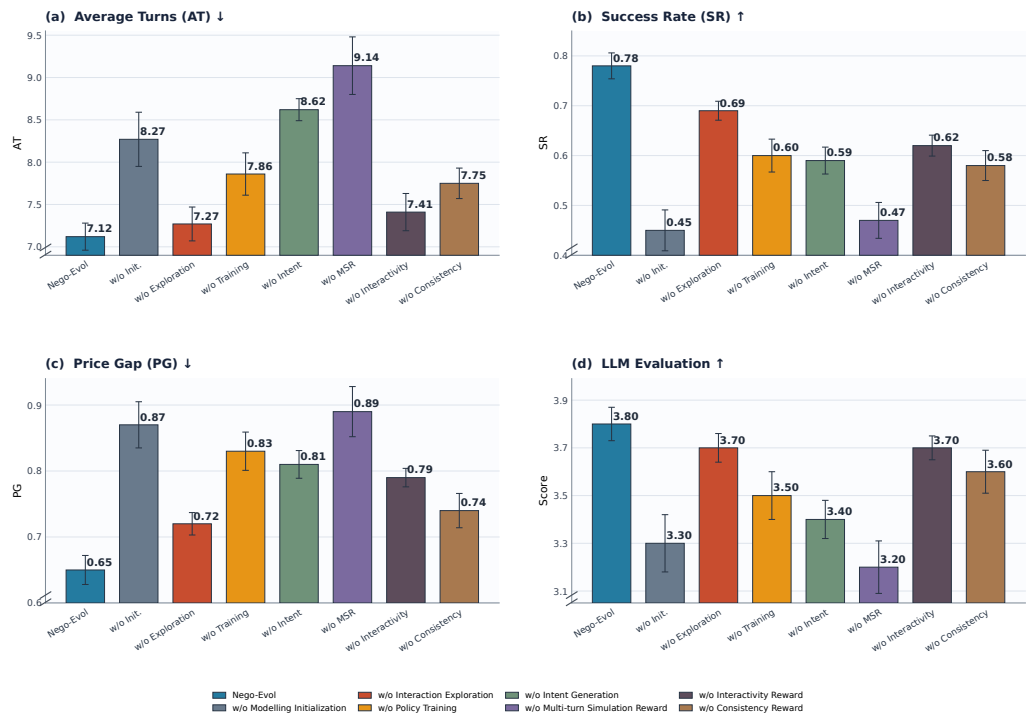


Figure 2: Ablation results on CraigslistBargain. Lower AT and PG and higher SR and LLM Evaluation indicate better performance. Error bars are provisional layout placeholders, pending empirical uncertainty estimates.

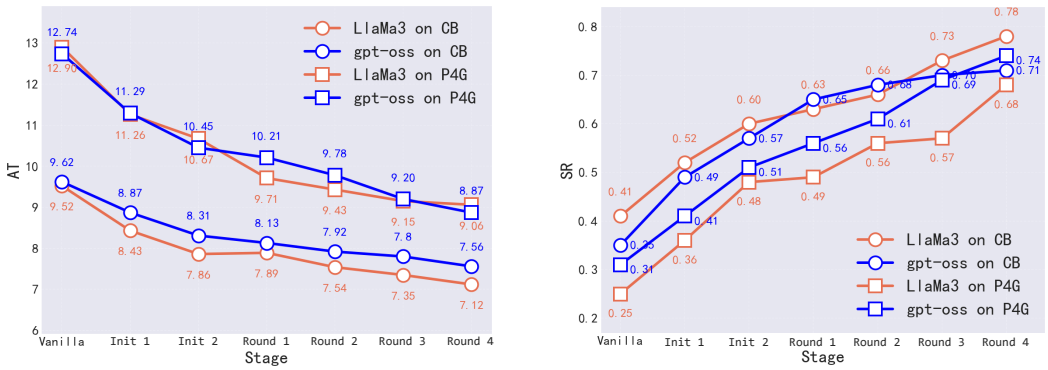


Figure 3: Performance across two initialization epochs and four rounds of Environment Modelling. Left: AT; right: SR.

shows higher final SR under alternation on CB (0.78 versus 0.64) and P4G (0.68 versus 0.50), together with lower CB PG and P4G AT. This comparison supports refreshing the policy used for exploration.

Table 3: Final performance under alternating exploration and policy training versus a fixed-data control using Llama-3.1-8B-Instruct.

Condition	CB SR ↑	CB PG ↓	P4G SR ↑	P4G AT ↓
Nego-Evol	0.78	0.65	0.68	9.06
Fixed-data control	0.64	0.73	0.50	10.18

Table 4: Blinded assessment of strategy use on CB. Values are percentages of evaluated outputs.

Policy	Contextual fit ↑	Intent–response alignment ↑	Valid new use ↑
Initialized policy π_{θ_1}	78%	85%	57%
Final policy	89%	94%	82%

4.5 Strategy Analysis

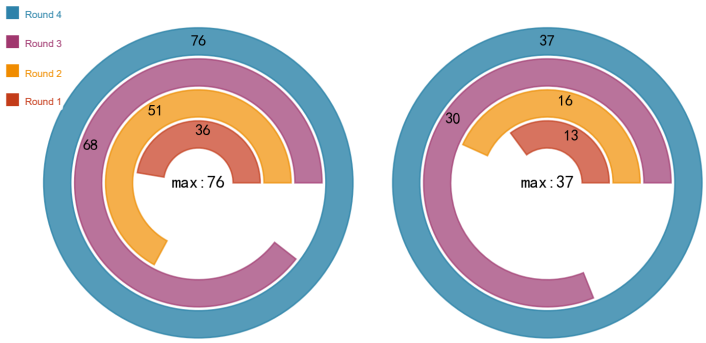


Figure 4: Cumulative numbers of intents recorded as new among strategies committed to selected interaction trajectories on CB (left) and P4G (right) over four rounds.

We examine both the intents identified during exploration and the final policy's use of strategies, where a strategy is an intent paired with its response. The cumulative numbers of intents recorded as new among committed trajectories rise across four rounds are illustrated in Figure 4. These counts describe the selected trajectories, not every candidate considered in the search.

Four annotators blindly assessed CB outputs given the preceding history, generated intent, and response. Contextual fit measures whether the intent suits the dialogue; alignment checks whether the response realizes that intent; valid new use requires an intent beyond the initial inventory that is suitable and realized. Table 4 shows gains of 11, 9, and 25 percentage points from the initialized to the final policy, respectively. This CB assessment complements the intent counts by checking whether discovered strategies can be used coherently. Illustrative responses and a separate, informal review of ten sampled CB conversations are in Appendix B.3.

5 Conclusion

Nego-Evol first initializes a policy from annotated dialogue turns, then alternates interaction exploration with outcome- and quality-guided policy training. On CraigslistBargain and PersuasionForGood, it improves mean success rates over the compared baselines across both backbones. A fixed-data control supports the value of refreshing exploration with the updated policy, while the CB strategy assessment shows improved contextual fit, intent–response alignment, and valid use of new intents. The conclusions are limited by simulated counterparts and automatic quality judgments; evaluation with human counterparts would test how well these gains transfer to live negotiation.

AI USE STATEMENT

The research ideas, experimental design, experiments, and substantive manuscript content were developed by the authors. Generative AI tools assisted with portions of the code and with language polishing. The authors reviewed and verified all AI-assisted work and take responsibility for the final paper and results.

ETHICS STATEMENT

This study uses existing negotiation datasets, simulated interactions, and human assessment of generated outputs; the agents were not deployed in live negotiations. Negotiation and persuasion agents could be used to pressure or mislead people. Any real-world use should disclose the agent's identity and include appropriate human oversight.

REPRODUCIBILITY STATEMENT

The experimental setup describes the datasets, model backbones, baselines, metrics, and evaluation protocol. Appendix A.1 provides the training algorithm, while Appendices **??** and **??** give implementation settings and task prompt templates. The main results report means and 95

REFERENCES

Tamer Abuelsaad, Deepak Akkil, Prasenjit Dey, Ashish Jagmohan, Aditya Vempaty, and Ravi Kokku. Agent-e: From autonomous web navigation to foundational design principles in agentic systems, 2024. URL https://arxiv.org/abs/2407.13032.

Zishan Ahmad, Suman Saurabh, Vaishakh Menon, Asif Ekbal, Roshni Ramnani, and Anutosh Maitra. INA: An integrative approach for enhancing negotiation strategies with reward-based dialogue agent. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 2536–2549, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.166. URL https://aclanthology.org/2023.findings-emnlp.166/.

Kushal Chawla, Ian Wu, Yu Rong, Gale Lucas, and Jonathan Gratch. Be selfish, but wisely: Investigating the impact of agent personality in mixed-motive human-agent interactions. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 13078–13092, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.808. URL https://aclanthology.org/2023.emnlp-main.808/.

Maximillian Chen, Xiao Yu, Weiyan Shi, Urvi Awasthi, and Zhou Yu. Controllable mixed-initiative dialogue generation through prompting. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 951–966, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-short.82. URL https://aclanthology.org/2023.acl-short.82/.

Zheng Chu, Jingchang Chen, Qianglong Chen, Weijiang Yu, Tao He, Haotian Wang, Weihua Peng, Ming Liu, Bing Qin, and Ting Liu. Navigate through enigmatic labyrinth a survey of chain of thought reasoning: Advances, frontiers and future. In *The 62nd Annual Meeting of the Association for Computational Linguistics: ACL 2024, Bangkok, Thailand, August 11–16, 2024*. Association for Computational Linguistics, 2024. URL https://arxiv.org/abs/2309.15402.

Yang Deng, Wenqiang Lei, Wai Lam, and Tat-Seng Chua. A survey on proactive dialogue systems: Problems, methods, and prospects. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023, 19th-25th August 2023, Macao, SAR, China*, pp. 6583–6591. ijcai.org, 2023a. doi: 10.24963/IJCAI.2023/738. URL https://doi.org/10.24963/ijcai.2023/738.

Yang Deng, Lizi Liao, Liang Chen, Hongru Wang, Wenqiang Lei, and Tat-Seng Chua. Prompting and evaluating large language models for proactive dialogues: Clarification, target-guided, and non-collaboration. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 10602–10621, Singapore, December 2023b. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.711. URL https://aclanthology.org/2023.findings-emnlp.711/.

Yang Deng, Wenxuan Zhang, Wai Lam, See-Kiong Ng, and Tat-Seng Chua. Plug-and-play policy planner for large language model powered dialogue agents. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=MCNqgUFTHI.

Yao Fu, Hao Peng, Tushar Khot, and Mirella Lapata. Improving language model negotiation with self-play and in-context learning from ai feedback, 2023. URL https://arxiv.org/abs/2305.10142.

Jiacheng Guo, Ling Yang, Peter Chen, Qixin Xiao, Yinjie Wang, Xinzhe Juan, Jiahao Qiu, Ke Shen, and Mengdi Wang. Genenv: Difficulty-aligned co-evolution between llm agents and environment simulators, 2025. URL https://arxiv.org/abs/2512.19682.

He He, Derek Chen, Anusha Balakrishnan, and Percy Liang. Decoupling strategy and generation in negotiation dialogues. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2333–2343, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1256. URL https://aclanthology.org/D18-1256/.

Tao He, Lizi Liao, Yixin Cao, Yuanxing Liu, Ming Liu, Zerui Chen, and Bing Qin. Planning like human: A dual-process framework for dialogue planning. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4768–4791, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.262. URL https://aclanthology.org/2024.acl-long.262/.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Yuncheng Hua, Lizhen Qu, and Reza Haf. Assistive large language model agents for socially-aware negotiation dialogues. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 8047–8074. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-emnlp.473. URL http://dx.doi.org/10.18653/v1/2024.findings-emnlp.473.

Hyeonbin Hwang, Doyoung Kim, Seungone Kim, Seonghyeon Ye, and Minjoon Seo. Self-explore: Enhancing mathematical reasoning in language models with fine-grained rewards. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 1444–1466, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.78. URL https://aclanthology.org/2024.findings-emnlp.78/.

Deuksin Kwon, Emily Weiss, Tara Kulshrestha, Kushal Chawla, Gale Lucas, and Jonathan Gratch. Are LLMs effective negotiators? systematic evaluation of the multifaceted capabilities of LLMs in negotiation dialogues. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 5391–5413, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.310. URL https://aclanthology.org/2024.findings-emnlp.310/.

Yutong Liu, Lida Shi, Rui Song, and Hao Xu. A dual-mind framework for strategic and expressive negotiation agent. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher

Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 23840–23860, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1161. URL https://aclanthology.org/2025.acl-long.1161/.

AI@Meta Llama Team. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Yingqian Min, Zhipeng Chen, Jinhao Jiang, Jie Chen, Jia Deng, Yiwen Hu, Yiru Tang, Jiapeng Wang, Xiaoxue Cheng, Huatong Song, Wayne Xin Zhao, Zheng Liu, Zhongyuan Wang, and Ji-Rong Wen. Imitate, explore, and self-improve: A reproduction report on slow-thinking reasoning systems. *CoRR*, abs/2412.09413, 2024. doi: 10.48550/ARXIV.2412.09413. URL https://doi.org/10.48550/arXiv.2412.09413.

Clint Morris, Michael Jurado, and Jason Zutty. Llm guided evolution - the automation of models advancing models. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '24, pp. 377–384. ACM, July 2024. doi: 10.1145/3638529.3654178. URL http://dx.doi.org/10.1145/3638529.3654178.

Ansong Ni, Miltiadis Allamanis, Arman Cohan, Yinlin Deng, Kensen Shi, Charles Sutton, and Pengcheng Yin. Next: teaching large language models to reason about code execution. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

OpenAI. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

OpenAI. gpt-oss-120b & gpt-oss-20b model card, 2025. URL https://arxiv.org/abs/2508.10925.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=HPuSIXJaa9.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Ryan Shea, Aymen Kallala, Xin Lucy Liu, Michael W. Morris, and Zhou Yu. ACE: A LLM-based negotiation coaching system. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 12720–12749, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.709. URL https://aclanthology.org/2024.emnlp-main.709/.

Hongjin Su, Shuyang Jiang, Yuhang Lai, Haoyuan Wu, Boao Shi, Che Liu, Qian Liu, and Tao Yu. EvoR: Evolving retrieval for code generation. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 2538–2554, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.143. URL https://aclanthology.org/2024.findings-emnlp.143/.

Xuewei Wang, Weiyan Shi, Richard Kim, Yoojung Oh, Sijia Yang, Jingwen Zhang, and Zhou Yu. Persuasion for good: Towards a personalized persuasive dialogue system for social good. In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 5635–5649, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1566. URL https://aclanthology.org/P19-1566/.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 1819–1862, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.102. URL https://aclanthology.org/2024.naacl-long.102/.

Huajian Xin, Z.Z. Ren, Junxiao Song, Zhihong Shao, Wanjia Zhao, Haocheng Wang, Bo Liu, Liyue Zhang, Xuan Lu, Qiushi Du, Wenjun Gao, Haowei Zhang, Qihao Zhu, Dejian Yang, Zhibin Gou, Z.F. Wu, Fuli Luo, and Chong Ruan. Deepseek-prover-v1.5: Harnessing proof assistant feedback for reinforcement learning and monte-carlo tree search. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=I4YAIwrsXa.

Chengyi Yang, Zhishang Xiang, Yunbo Tang, Zongpei Teng, Chengsong Huang, Fei Long, Yuhan Liu, and Jinsong Su. Ttcs: Test-time curriculum synthesis for self-evolving. *arXiv preprint arXiv:2601.22628*, 2026.

Xiao Yu, Maximillian Chen, and Zhou Yu. Prompt-based Monte-Carlo tree search for goal-oriented dialogue policy planning. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7101–7125, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.439. URL https://aclanthology.org/2023.emnlp-main.439/.

Haolan Zhan, Yufei Wang, Zhuang Li, Tao Feng, Yuncheng Hua, Suraj Sharma, Lizhen Qu, Zhaleh Semnani Azad, Ingrid Zukerman, and Reza Haf. Let's negotiate! a survey of negotiation dialogue systems. In Yvette Graham and Matthew Purver (eds.), *Findings of the Association for Computational Linguistics: EACL 2024*, pp. 2019–2031, St. Julian's, Malta, March 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.findings-eacl.136/.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. DIALOGPT : Large-scale generative pre-training for conversational response generation. In Asli Celikyilmaz and Tsung-Hsien Wen (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp. 270–278, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-demos.30. URL https://aclanthology.org/2020.acl-demos.30/.

Andrew Zhao, Yiran Wu, Yang Yue, Tong Wu, Quentin Xu, Yang Yue, Matthieu Lin, Shenzhi Wang, Qingyun Wu, Zilong Zheng, and Gao Huang. Absolute zero: Reinforced self-play reasoning with zero data, 2025. URL https://arxiv.org/abs/2505.03335.

A Additional Method Details

A.1 Nego-Evol Training Algorithm

Algorithm 1 summarizes Environment Initialization and the alternating Interaction Exploration and Policy Training rounds. The search uses the current policy without updating its parameters; Policy Training subsequently updates that policy using newly sampled outputs.

A.2 Intent Novelty

Let $f(\cdot)$ map an intent string to a sentence embedding, and let $\cos(\cdot,\cdot)$ denote cosine similarity. For a candidate intent z generated in round e, its novelty relative to the current inventory is

$$\eta(z) = 1 - \max_{\bar{z}\in\mathcal{I}_e} \cos\big(f(z), f(\bar{z})\big). \tag{9}$$

Algorithm 1 Nego-Evol Training Algorithm

Input: Human-annotated dialogues, training scenarios, evolution rounds E, search budget B, search candidate count m, training group size M, future simulation count n.
Output: Policy $\pi_{\theta_{E+1}}$.
Construct $\mathcal{D}_{\text{init}}$ from $x_t = (u, h_t)$ and $y_t = (q_t, z_t, r_t)$, synthesizing q_t offline.
Optimize Equation 2 to obtain π_{θ_1}; initialize $\mathcal{I}_{\text{init}}$ from annotated intents.
for $e = 1$ to E **do**
  Form $\mathcal{I}_e$ using Equation 10; keep π_{θ_e} fixed during search.
  Initialize $\mathcal{D}^{(e)}_{\text{MCTS}} \leftarrow \varnothing$ and $\mathcal{I}^{(e)}_{\text{MCTS}} \leftarrow \varnothing$.
  for each training scenario **do**
    Initialize its history h_1 and set $t \leftarrow 1$.
    while the dialogue has not terminated **do**
      Search from $x_t = (u, h_t)$ with budget B; generate m candidates per expansion using $p_e = \exp(-e)$.
      Select and simulate branches using Equation 3; update N_e, W_e, Q_e using Equation 11.
      Commit y_t^* and its cached opponent reply using Equations 12 and 13.
      Set $t \leftarrow t + 1$.
    end while
    Add the completed trajectory to $\mathcal{D}^{(e)}_{\text{MCTS}}$ and its committed strategy intents to $\mathcal{I}^{(e)}_{\text{MCTS}}$.
  end for
  for each trajectory in $\mathcal{D}^{(e)}_{\text{MCTS}}$ with K agent responses **do**
    for $t = 1$ to K **do**
      Construct $x_t = (u, h_t)$ and sample M structured outputs from π_{θ_e}.
      for $j = 1$ to M **do**
        Simulate n future negotiations after $r_t^{(j)}$; compute $\text{MSR}_t^{(j)}$ and $R_t^{(j)}$ using Equations 4 and 5.
      end for
      Compute advantages using Equation 6.
    end for
  end for
  Optimize Equation 7 to obtain $\pi_{\theta_{e+1}}$.
end for
return $\pi_{\theta_{E+1}}$.

Here $\mathcal{I}_e$ contains the annotated intents from initialization and the intents of strategies committed to completed trajectories in earlier exploration rounds. Intents generated only on uncommitted search branches are not added to the inventory.

An intent is identified as outside the current inventory when $\eta(z) > 0.2$, equivalently when its maximum cosine similarity to the inventory is below 0.8. The continuous value $\eta(z)$ is used in the branch score and turn-level commitment rule in Equations 3 and 12.

A.3 Open-Loop Search Realizations

At round e, the inventory contains annotated intents and the intents of strategies committed to selected trajectories in earlier rounds:

$$\mathcal{I}_e = \mathcal{I}_{\text{init}} \cup \bigcup_{s=1}^{e-1} \mathcal{I}^{(s)}_{\text{MCTS}}. \tag{10}$$

At a simulated policy turn k, the expert supplies candidates with probability $p_e = \exp(-e)$, and the fixed policy supplies them otherwise. The resulting m outputs are $\mathcal{Y}_k = \{y_k^{(j)}\}_{j=1}^m$ with $y_k^{(j)} = (q_k^{(j)}, z_k^{(j)}, r_k^{(j)})$.

A search branch records a candidate output, but only its response is sent to the opponent. Revisiting a branch may produce different sampled opponent utterances and realized histories; its statistics are shared across those realizations. For each policy branch $(x_k, y_k^{(j_k)})$ in a rollout τ, the terminal result

is backed up as

$$N_e(x_k, y_k^{(j_k)}) \leftarrow N_e(x_k, y_k^{(j_k)}) + 1,$$
$$W_e(x_k, y_k^{(j_k)}) \leftarrow W_e(x_k, y_k^{(j_k)}) + G(\tau),$$
$$Q_e(x_k, y_k^{(j_k)}) \leftarrow \frac{W_e(x_k, y_k^{(j_k)})}{N_e(x_k, y_k^{(j_k)})}. \tag{11}$$

After the search budget at actual turn t is exhausted, the search commits

$$y_t^* = \arg \max_{y_t^{(j)} \in \mathcal{Y}_t} \left[Q_e(x_t, y_t^{(j)}) + \alpha_e \eta(z_t^{(j)}) \right]. \tag{12}$$

The realized rollout retains an opponent utterance o_{t+1}^* associated with this candidate. It forms the next input:

$$h_{t+1} = h_t \oplus (r_t^*, o_{t+1}^*), \qquad x_{t+1} = (u, h_{t+1}). \tag{13}$$

The search restarts from this history. Only strategies committed to the completed trajectory contribute intents to $\mathcal{I}_{\text{MCTS}}^{(e)}$.

A.4 Task-Specific Terminal Results

The terminal result $G(\tau)$ in Equations 11 and 4 is determined when a simulated negotiation ends. For PersuasionForGood, $G(\tau)$ is 1 if the persuasion succeeds and 0 otherwise.

For CraigslistBargain, the price gap is

$$\text{PG}(\tau) = \frac{p_{\text{final}} - p_{\text{initial}}}{p_{\text{initial}}}, \tag{14}$$

where p_{initial} is the initially proposed price and p_{final} is the agreed final price. For an agreement, we use $G(\tau) = -\text{PG}(\tau)$, so a lower final price gives the buyer a larger terminal result. A negotiation without an agreement receives $G(\tau) = 0$.

B Supplementary Experiments

B.1 Experimental Configuration

Initialization and policy optimization. Initialization uses two epochs with batch size of 4, learning rate 10^{-5}, and maximum sequence length 2,048. Policy Training also uses a reported batch size of 4 and group size $M = 8$. Both backbones are adapted with LoRA; rank $r = 16$, scaling factor $\alpha = 32$, dropout rate 0.05, and target modules covering all linear projections (`q_proj`, `k_proj`, `v_proj`, `o_proj`, `gate_proj`, `up_proj`, `down_proj`). The optimizer, precision or quantization mode, learning-rate schedule, and warmup are the AdamW optimizer, `bfloat16` mixed-precision, a cosine learning-rate decay schedule, and a 10% step warmup. For Policy Training, learning rate of 1×10^{-6}, 1 PPO-style update per rollout round, clipping coefficient $\epsilon = 0.2$, KL penalty coefficient $\beta = 0.04$, and the initialized model as the fixed reference policy. The output-token scope follows Equation 7.

Search and simulation. There are four Environment Modelling rounds. Each search expansion generates $m = 3$ candidates, and each agent turn uses five search iterations. The reported maximum number of search rollout realizations per branch is 3. The expert-generation probability follows $p_e = \exp(-e)$; the initial novelty coefficient is $\alpha_0 = 0.5$, and an intent is outside the inventory when its maximum cosine similarity is below 0.8. We use the `all-MiniLM-L6-v2` sentence-embedding model, set the exploration coefficient to $c_p = 0.05$, and apply 0.4 for the novelty coefficient. The maximum simulated dialogue length is 15. Policy Training uses three independently simulated futures per candidate and reward weights $(\lambda_1, \lambda_2, \lambda_3) = (0.4, 0.3, 0.3)$.

B.2 Additional Ablation Results on PersuasionForGood

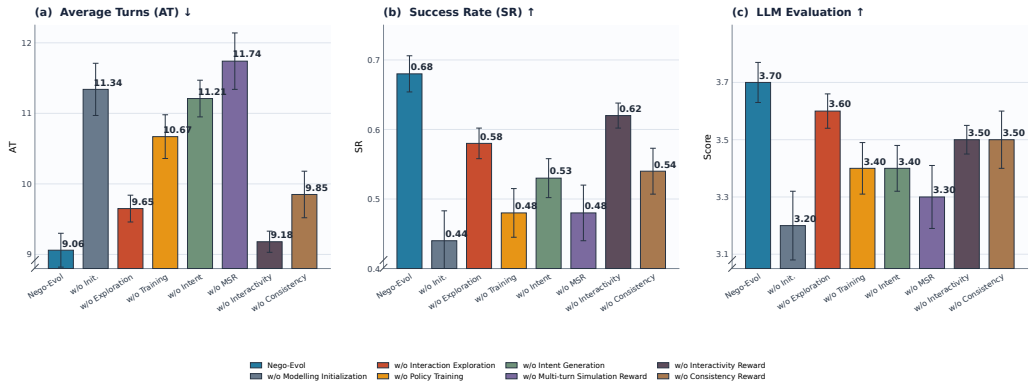


Figure 5: Ablation results on PersuasionForGood. Lower AT and higher SR and LLM Evaluation indicate better performance. Error bars are temporary layout placeholders pending empirical uncertainty estimates.

Figure 5 provides the PersuasionForGood results corresponding to the CraigslistBargain ablations in Section 4.3. The complete model achieves the lowest AT (9.06), highest SR (0.68), and highest LLM Evaluation score (3.70). Removing Modelling Initialization yields the lowest SR (0.44) and dialogue-quality score (3.20), whereas removing MSR produces the highest AT (11.74). Removing Policy Training also substantially reduces SR to 0.48.

The response-level rewards have smaller but distinct effects. Without the interactivity reward, SR decreases to 0.62 and LLM Evaluation to 3.50, while AT remains close to that of the complete model (9.18 versus 9.06). Without the consistency reward, SR decreases further to 0.54 and AT rises to 9.85. This pattern is consistent with the CB results: simulated long-term outcomes provide the strongest reward signal, while the other terms contribute to response quality and negotiation effectiveness.

B.3 Representative Cases and Informal Feedback

Table 5 reproduces illustrative CB responses from the original analysis, with feedback paraphrased for brevity. Two undergraduate reviewers examined ten sampled negotiation conversations. This small, informal review is separate from the four-annotator blinded strategy assessment in Table 4; it provides examples, not a quantitative estimate of strategy quality. The examples come from different contexts and are not matched responses to the same input.

Table 5: Illustrative CB responses and paraphrased informal feedback from the original analysis. Examples are from different conversations.

Policy	Example intent and response	Reviewer observation
Vanilla	"[Affirm Information] My offer is $150. Yes or no." "[Initial Price] This watch is pretty average, I'll take it off your hands for $30."	The agent sometimes accepts an unfavorable price or repeats questions.
Initialized policy	"[Propose Initial Price] Nice jacket. Would you give me a discount if I buy it?" "[Counter Price] This is more expensive than I thought. Can you do better on the price?"	Responses are relevant and structured, but have limited strategic depth.
Nego-Evol	"[Cash-in-Hand] I can come pick it up this afternoon with cash. For a hassle-free sale, would you take $400 cash?" "[Point Out Flaw] The cord is quite old. With the cost of repair, $80 is my final offer."	Responses show more awareness of trade-offs and longer-term goals; some concise replies need elaboration.

C Task and Role Prompt Templates

The following draft templates preserve the task settings, role instructions, and intent guidance of the original prompts while using the structured tags specified in Section 3.1. The intent lists are examples; the policy may generate a new intent when it fits the dialogue. Only the `<response>` content is sent to the counterpart.

C.1 CraigslistBargain: Buyer Policy

```
Role: buyer. Negotiate naturally and respectfully with the seller.
Item: {Title}. Description: {Description}.
Buyer target price: {Price_Buyer}.
Conversation so far: {chat_history}

Goal: reach an agreement at a favorable price. Maintain price
consistency, address the seller's latest message, and avoid repetition.
Possible intents include greeting, asking or answering a question,
proposing an initial or counter price, using a comparison, confirming
information, agreeing, and disagreeing. You may generate another
appropriate intent when it better fits the current negotiation.

Output exactly:
<think>{brief rationale grounded in the history and goal}</think>
<intent>{free-form intent}</intent>
<response>{short, natural buyer reply}</response>
If the negotiation ends, include {NEGOLLM_TERMINATION_SIGNAL} in the
response.
```

C.2 CraigslistBargain: Seller Simulation

```
Role: seller. Respond naturally to the buyer's latest message.
Item: {Title}. Description: {Description}.
Initial seller price: {Price_Seller}.
Conversation so far: {chat_history}

Maintain the seller's role and price logic. A reply may ask for
information, offer a counterargument, express a preference or
hesitation, assert a position, accept, or decline. Vary the wording
and avoid repeating earlier replies.

Output exactly:
<think>{brief role-consistent rationale}</think>
<intent>{seller intent}</intent>
```

```
<response>{short, natural seller reply}</response>
If the negotiation ends, include {NEGOLLM_TERMINATION_SIGNAL} in the
response.
```

C.3 PERSUASIONFORGOOD: PERSUADER POLICY

```
Role: persuader. Encourage a donation to Save the Children while
respecting the persuadee's concerns and autonomy.
Background: the charity helps children in developing countries and
conflict zones; small donations can help.
Conversation so far: {chat_history}

Possible intents include logical appeal, emotional appeal,
credibility appeal, foot-in-the-door request, personal story,
donation information, and source-, task-, or person-related inquiry.
You may use another appropriate intent when it better addresses
the persuadee's current concern. Remain polite and avoid unsupported
claims about the charity.

Output exactly:
<think>{brief rationale grounded in the history and goal}</think>
<intent>{free-form intent}</intent>
<response>{short, natural persuader reply}</response>
If the exchange ends, include {NEGOLLM_TERMINATION_SIGNAL} in the
response.
```

C.4 PERSUASIONFORGOOD: PERSUADEE SIMULATION

```
Role: persuadee. Respond naturally to the persuader's latest message.
Conversation so far: {chat_history}

Initially know little about the charity and have limited willingness
to donate. Your willingness may change if the persuader addresses
your questions or concerns. You may ask for information, express
hesitation or a personal preference, counter an argument, donate,
or refuse. Stay consistent with the conversation.

Output exactly:
<think>{brief role-consistent rationale}</think>
<intent>{persuadee intent}</intent>
<response>{short, natural persuadee reply}</response>
If you decide to donate or end the conversation without donating,
include {NEGOLLM_TERMINATION_SIGNAL} in the response.
```