

# CoEvoLoop: Closed-Loop RL Training of Customer-Service Agents by Co-Evolving Environment, Reward, and Policy

Hui Cai, Jianda Chen, Yuhong Zhou, Wei Ren, Choy Zhen Wen Marcus,  
Tan Li Hui Faith, Lyuxin Xue, Hao Chen, Jiawei Luo, Xuanwei Hu, Xin Zhang\*,  
Ant International

{biyu.ch, jianda.chen, zhoyuhong.zyh, modi.rw, marcus.choy,  
faith.t, xuelyuxin.xlx, yahao.ch, laurent.ljw, xuanwei.hwx, evan.zx}@ant-intl.com

## Abstract

Developing enterprise customer-service assistants with Reinforcement Learning (RL) critically depends on high-fidelity user simulators and well-aligned reward functions, since real-customer interactions are impractical for large-scale training and rule-based rewards fail to capture complex business requirements. We present CoEvoLoop, a framework that co-evolves the assistant policy together with these RL components through an expert-calibrated, multi-round loop. Beyond an LLM-based simulator for rollout generation and a multi-dimensional grader for rubric-based evaluation of service quality and compliance, CoEvoLoop integrates an expert annotation interface where sampled trajectories and grader judgments are reviewed and corrected. Each round, expert feedback updates the simulator and grader, after which the assistant is optimized with trajectory-level RL against the updated evaluation signal; monitoring interfaces support rollout inspection, dimension-wise failure attribution, and cross-round tracking. In an industrial evaluation, the system improves multi-turn service quality and reduces compliance violations over training rounds; component ablations and blind expert evaluation confirm that co-evolving the simulator and grader is load-bearing for these gains. We release the system demo code<sup>1</sup>, along with a demo video<sup>2</sup>.

## 1 Introduction

Reinforcement learning (RL) is the natural fit for the multi-turn, long-horizon problem solving that *enterprise customer service* now demands (Chen et al., 2025; Zhao et al., 2025a; Xi et al., 2025). But RL needs two things that production cannot safely provide: customers to practice on, and a ground

truth for what “good service” means under enterprise policy. Both are therefore replaced by LLM proxies: a **user simulator** that plays the customers, acting as the **environment** (Wang et al., 2025), and an LLM-as-a-judge **grader** that supplies the **reward** (Gu et al., 2025). These proxies inherit the flaws of the LLMs they are built from: they are biased, inconsistent, and only loosely aligned with organization-specific requirements, while also having to stay efficient enough for high-throughput rollouts (Li et al., 2025a).

The prevailing recipe treats both proxies as fixed infrastructure, frozen while the policy trains; a frozen proxy is a stationary target. In our experiments, RL steadily raised the frozen grader’s reward while the policy fell *below* its supervised starting point under expert-calibrated evaluation (Section 3): it learned the grader’s blind spots, not better service. A frozen simulator likewise keeps rehearsing yesterday’s customers as the policy’s weaknesses move elsewhere, and a single scalar reward can neither express the trade-off between helpfulness and strict compliance nor say which rule was broken. The remedy is shared: the proxies must improve together with the policy, re-anchored to enterprise criteria by experts every round.

CoEvoLoop embodies this remedy: a system in which the RL environment, the reward model, and the policy are all trainees of one expert-anchored loop, whereas prior pipelines typically hold at least one of them fixed. CoEvoLoop composes standard training techniques (SFT, DPO, GRPO) into the **co-evolving training loop** of Figure 1: (1) the user simulator generates multi-turn rollouts with the assistant; (2) the multi-dimensional grader assigns per-dimension feedback; (3) experts review a sampled subset under the same rubric; (4) the grader and simulator are updated from these annotations; and (5) the assistant is retrained with SFT and trajectory-level RL against the newly calibrated grader. Interfaces for dialogue synthesis,

\*Corresponding author

<sup>1</sup><https://github.com/ant-intl/CoEvoLoop>

<sup>2</sup><https://youtu.be/sd1fVKKzp1M>

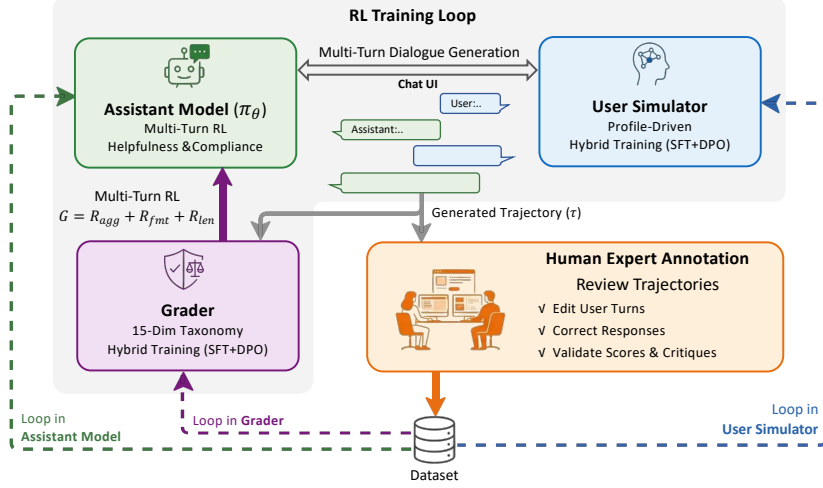


Figure 1: The CoEvoLoop framework: the user simulator (environment), the grader (reward), and the assistant (policy) co-evolve under human-in-the-loop calibration.

expert annotation, and RL inspection make each step observable: the demo does not just present the system, it makes reward exploitation itself visible as a rising training reward against a falling calibrated score.

Concretely, we contribute: (1) the CoEvoLoop system implementing this co-evolving loop with its three interfaces; (2) the expert-calibration design behind it, pairing a multi-dimensional quality/compliance rubric with a compliance-aware reward; (3) an industrial validation of the loop with component ablations and blind expert evaluation; and (4) actionable practical findings (Section 5).

## 2 Methodology

### 2.1 Overview of System Demo: A Closed-loop for Training

CoEvoLoop consists of three components, one for each role in the RL problem. The **assistant** (the *policy*) is the customer-service agent optimized for helpful multi-turn responses under enterprise constraints (Section 2.4). The **user simulator** (the *environment*) generates multi-turn customer behavior conditioned on enterprise scenarios (e.g., intent, tools/permissions, sensitive attributes, escalation triggers), enabling controllable rollouts without production traffic (Section 2.2). The **grader** (the *reward model*) evaluates trajectories with an enterprise rubric decomposed into dialog-quality and compliance dimensions (Section 2.3); its per-dimension outputs serve training and diagnosis.

Figure 2 walks through the training loop as it appears in the demo UI. (1) **Synthesis**: after a

scenario is selected and the simulator configured, the assistant generates multi-turn dialogues that are saved to a dataset. (2) **Annotation**: experts open saved trajectories, review the rollout end-to-end, and edit rubric-based grades and judgments; when necessary, they revise specific turns to provide clean supervision. (3) **Update & Train**: the annotated dataset refines the grader and simulator (e.g., via SFT/DPO), after which the assistant is trained with SFT and trajectory-level RL against the newly calibrated grader; progress and failures are monitored on the RL Inspection page.

### 2.2 Loop in User Simulator

Our user simulator mimics real customers to generate dialogues conditioned on a structured *user profile*: a representation reverse-engineered from anonymized real-world dialogues by a strong open-source LLM, comprising background description, knowledge gap, action history, and problem description (example in Appendix A.4). Given a profile, a smaller model (Qwen3-8B) performs profile-driven dialogue generation.

The simulator is trained in two phases for human-like, robust responses: SFT on a hybrid dataset (anonymized real-world dialogues plus synthetic data teaching reactions to common AI assistant failures) builds foundational competence, and DPO (Rafailov et al., 2023) then bolsters robustness with preference pairs contrasting desirable, non-repetitive reactions to flawed assistant behavior with undesirable ones.

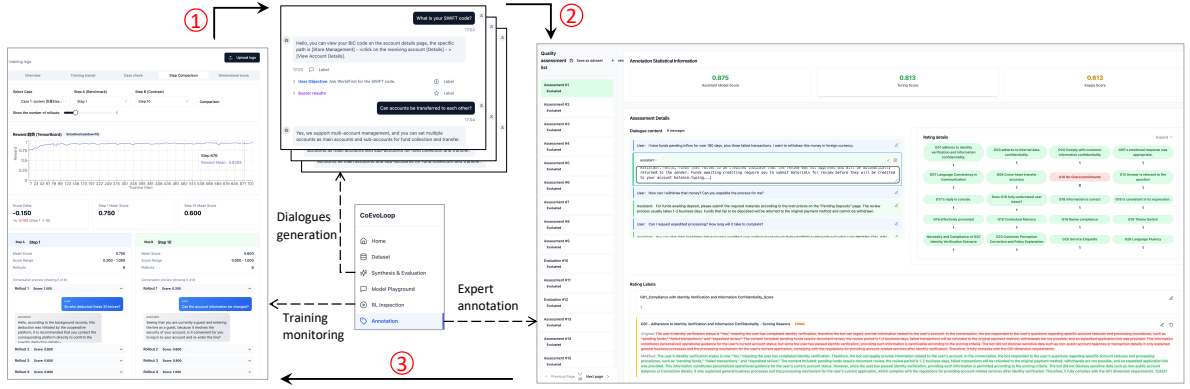


Figure 2: Demo workflow across the three interfaces: dialogue **Synthesis** (simulator configuration, rollout generation), expert **Annotation** (per-dimension grade review and turn-level correction), and **RL Inspection** (reward curves and dimension-level failure tracking).

**Continuous Improvement** As mentioned in Section 2.1, generated dialogues are annotated by human experts for *profile rationality* and *dialogue-profile consistency*, and validated by *human-likeness* evaluation (Turing test, Appendix A.2.1). This annotated data is appended to the SFT and DPO datasets and fed into the next round of training, continuously improving the simulator.

### 2.3 Loop in Grader

The multi-dimensional grader is the core of our closed-loop framework: a specialized evaluator that proxies human judgment, balancing conversational helpfulness with strict policy compliance. Given a dialogue context and an agent response, it outputs (i) a scalar score per dimension and (ii) an interpretable critique justifying the scores.

**Fine-grained Taxonomy** The grader is founded on an evaluation taxonomy of 15 dimensions in four pillars: *Security & Compliance*, *Information Correctness*, *Logical Reasonability*, and *Service Quality* (Appendix A.2.2). The decomposition is not a priori: it is distilled from the enterprise quality-assurance rubric and a review of historical service incidents. Dimensions are binary to maximize annotator consistency and make failure attribution unambiguous (coarser merged dimensions lowered inter-annotator agreement in pilots); the pillars separate hard compliance constraints from graded service qualities, exactly the tension the reward must arbitrate.

**Data construction** We build the grader dataset from raw Human-to-Human (H2H) and Human-to-Machine (H2M) logs: logs are sanitized to remove personally identifiable information (PII)

and normalize formatting, and a subsample is labeled by human experts with the 15 per-dimension scores. To scale supervision, strong teacher models (e.g., Qwen3-235B and Qwen3-80B-Next) generate pseudo-labels and critiques for unlabeled data; for each instance we perform multiple stochastic rollouts and define *teacher confidence* as the cross-rollout consistency of per-dimension scores.

**Difficulty-Aware SFT** We fine-tune a student model (Qwen3-32B) to predict the 15 per-dimension scores and generate critiques. Training examples are stratified by *teacher confidence* (multi-sample consistency) and *difficulty* (accuracy): all **high-confidence & hard** instances strengthen complex reasoning (e.g., subtle policy violations), **high-confidence & simple** instances prevent overfitting to trivial patterns, and **low-confidence** instances are discarded as noisy. This curriculum distills both the scoring behavior and the underlying reasoning.

**Stability-Enhancing DPO** To address inconsistencies in borderline cases, we subsequently apply DPO: for each input we sample multiple candidate grading outputs from the SFT model and select winners/losers ( $y_w, y_l$ ) by rubric consistency and agreement with available expert labels, sharpening the model’s decision boundaries and correcting specific error modes.

**Continuous Improvement** During closed-loop training (Section 2.1), the grader scores dialogues generated by the simulator and assistant, and experts annotate a sampled subset with the same rubric. Grader-expert disagreements are collected as *bad cases* and fed back (primarily as additional

DPO supervision), enabling continual adaptation as business policies evolve. On a held-out test set, the grader demonstrates robust alignment with human experts, achieving an overall accuracy of 92.44% and a Cohen’s Kappa of 0.663, i.e., substantial agreement (Landis and Koch, 1977) (see Appendix A.2.3).

## 2.4 Loop in Assistant Model

### 2.4.1 Supervised Fine-Tuning

The initial phase is Supervised Fine-Tuning (SFT) to establish a strong baseline policy  $\pi_{\text{SFT}}$ . We synthesize multi-turn dialogues between the user simulator and a strong teacher model (Gemini 2.5 Pro), elicited for step-by-step reasoning and a natural, helpful persona, and keep only dialogues that pass grader filtering; imitating these dense behaviors reduces the exploration challenge and accelerates RL convergence.

### 2.4.2 Multi-Turn Reinforcement Learning

We formulate multi-turn dialogue as an RL problem: the policy  $\pi_\theta$  is trained with GRPO (Shao et al., 2024) to maximize the expected episode-level reward  $\mathbb{E}_{\tau \sim \pi_\theta}[G]$  over full trajectories  $\tau$ , encouraging long-horizon coherence rather than turn-level myopic gains.

The total episode reward  $G$  is defined as a composite score that aggregates quality, structural adherence, and brevity across the full trajectory:

$$G = R_{\text{agg}} + R_{\text{fmt}} + R_{\text{len}} \quad (1)$$

**Multi-Dimensional Reward Aggregation** The Grader (Section 2.3) scores the assistant on 15 binary dimensions. To ensure holistic quality, the aggregated reward  $R_{\text{agg}} \in [0, 1]$  uses a stepped degradation function of the number of zero-valued dimensions  $n_z$ :

$$R_{\text{agg}} = \max(0, 1 - 0.2 \cdot n_z) \quad (2)$$

This imposes a strict penalty for partial failure: any single failed dimension sharply reduces the total reward.

**Format and Length Penalties** To enforce structured reasoning, the agent must encapsulate its latent thought process within `<think>` tags; the binary penalty  $R_{\text{fmt}} \in \{0, -1\}$  is  $-1$  when the response violates this format. To mitigate verbosity-driven reward hacking,  $R_{\text{len}} \in [-1, 0]$  applies a DAPO-inspired soft length penalty (Yu et al., 2025):

0 for responses up to a preferred length  $L_{\text{buf}}$ , decreasing linearly to  $-1$  as the length approaches the hard limit  $L_{\text{max}}$ , and  $-1$  beyond.

**Continuous Improvement** Each round retrains the assistant against the latest simulator and grader from the previous round, after which fresh dialogues are generated under the updated components for the next round of the loop.

## 2.5 Loop in Expert Calibration

Expert annotation links rollout generation to model updates in each round. Experts review a sampled subset of generated trajectories on the Annotation page (Figure 2) using the same rubric as the grader: they inspect the per-dimension score breakdown, correct labels that do not match enterprise criteria, add brief notes for ambiguous cases, and, when needed, edit dialogue turns to provide clean supervision (e.g., rewriting an assistant response to be compliant). Annotation volume does not grow with the loop: each round has a fixed review budget, and trajectories are prioritized by grader uncertainty and historical grader–expert disagreement (costs in Section 3; protocol in Appendix A.2.4). These annotations feed the next round’s grader and simulator updates, and corrected trajectories serve supervised fine-tuning and guide subsequent RL.

## 3 Experiments

**Model Configuration** We build on the Qwen3 family: Qwen3-32B is the backbone for the assistant policy and the reward models, while the user simulator uses the lighter Qwen3-8B. The multi-dimensional grader serves as the reward model, decomposed into two sub-models: one for knowledge-intensive dimensions requiring external retrieval, and one for non-knowledge-based evaluation.

**Evaluation Settings** We evaluate on 241 held-out user simulator profiles, disjoint from the 2,906 RL-training profiles. The final grader’s training and per-round calibration dialogues are also disjoint from its 414-dialogue held-out benchmark and from all profiles and trajectories in the assistant-evaluation set. The task is knowledge-grounded cross-border-payment customer service (account funding, transfers, refunds, policy inquiries); the assistant is grounded with retrieved knowledge-base context, without transactional tool-calling. Each profile drives a fixed 6-turn simulator–assistant dialogue; we run 5 independent simulations per profile

| Model                   | strict-agg $\uparrow$ | Human $\uparrow$ |
|-------------------------|-----------------------|------------------|
| Claude-3.5-Haiku        | 0.867                 | —                |
| Gemini-2.5-Pro          | 0.875                 | —                |
| GPT-5.2                 | 0.875                 | 82.50            |
| Qwen3-32B               | 0.821                 | —                |
| +SFT                    | 0.879                 | 85.00            |
| +SFT+RL (no loop)       | 0.855                 | 82.50            |
| +SFT+RL (frozen sim.)   | 0.892                 | 87.50            |
| +SFT+RL (frozen grader) | 0.863                 | 81.25            |
| +SFT+RL (full loop)     | <b>0.904</b>          | <b>88.75</b>     |

Table 1: Main results and loop ablations, all scored by the final expert-calibrated grader on the 241 held-out profiles (5 runs each). Human: strict-rubric pass rate (%) on 40 trajectories per system, averaged over two blind experts (Appendix A.2.5). *No loop*: one round with the initial simulator and grader; *frozen*: all rounds with that component fixed.

(sampling stochasticity makes each trajectory distinct), yielding  $241 \times 5 = 1,205$  trajectories per system, assessed across 15 dimensions. The strict aggregation rule (strict-agg@eval) scores a trajectory 1 only if all 15 dimensions score 1; the reported average is thus a pass rate, e.g., 90.4% of our best model’s trajectories pass all 15 dimensions. To avoid self-evaluation artifacts, *all* systems (including loop variants) are scored by the same *final expert-calibrated grader* (the grader produced by the last loop round), never by the grader a variant was trained against.

**Main Results** Table 1 reports the main comparison. Proprietary baselines are similar, with Claude-3.5-Haiku, Gemini-2.5-Pro, and GPT-5.2 scoring from 0.867 to 0.875. The base Qwen3-32B lags behind (0.821), suggesting that pretraining alone is insufficient to meet stringent multi-dimensional criteria. SFT provides a large gain, improving Qwen3-32B from 0.821 to 0.879, indicating effective alignment to the dialogue objectives and rubric. Running the full three-round loop yields the best results on both metrics (0.904 strict-agg and an 88.75% human pass rate), surpassing all baselines. This improvement suggests that RL particularly effectively reduces hard failure cases across challenging evaluation dimensions, which is crucial under the strict aggregation metric.

**Component Ablation** Table 1’s lower block disentangles the loop. *No loop*, a single round of RL against the initial simulator and grader, falls *below* the SFT baseline (0.855 vs. 0.879): naive RL

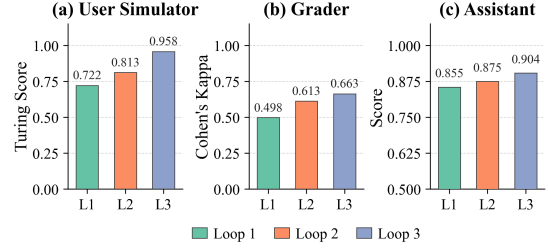


Figure 3: Performance of user simulator, grader, and assistant model at each loop round. Assistant scores are measured with the final expert-calibrated grader.

against an uncalibrated reward is worse than no RL at all. Evolving only the simulator (*frozen grader*) recovers little (0.863), since the reward remains exploitable. Evolving only the grader (*frozen sim.*) captures most of the gain (0.892), confirming reward calibration is the dominant factor, while the full loop adds a further +1.2 points through more realistic rollouts; both updates are load-bearing.

**Improvement over Loop** Figure 3 shows steady gains across three rounds: the user simulator becomes more realistic (Turing score  $0.722 \rightarrow 0.958$ ), the grader aligns better with human judgments (Cohen’s Kappa  $0.498 \rightarrow 0.663$ ), and the assistant’s overall score rises from 0.855 to 0.904.

**Do Policies Exploit Grader Bias?** Assistant scores in Figure 3 use the final calibrated grader, so the round-1 policy *is* the no-loop variant of Table 1, starting *below* the SFT baseline (0.855 vs. 0.879). The cause is reward exploitation: during round-1 training, the policy steadily raised the reward assigned by the round-1 grader it was optimized against, yet its expert-calibrated score fell below its own SFT starting point, i.e., it learned the grader’s blind spots rather than better service. Per-round recalibration closes this loophole: the calibrated score recovers and surpasses SFT in later rounds (0.875 and 0.904, Figure 3).

**Blind Human Evaluation** To verify that gains are not grader artifacts, experts blindly rated 40 held-out trajectories per system under the strict rubric (six systems marked in Table 1; Appendix A.2.5). Human pass rates track strict-agg@eval closely (system-level Pearson  $r = 0.93$  across the six systems), and the full loop improves the human-measured rate over SFT by +3.75 points, at least as large as the grader-measured gain, indicating the improvement is not metric gaming.

**Grader Reliability as a Reward Signal** Since the grader supplies the RL reward, its noise profile matters. The grader agrees substantially with experts (92.44% accuracy,  $\kappa = 0.663$ ; Appendix A.2.3), and three mechanisms bound the impact of residual noise: (i) stepped aggregation caps any single mis-judged dimension at 0.2 reward; (ii) GRPO’s group-relative advantage cancels profile-level biases; (iii) per-round recalibration shrinks the exploitation gap.

**Annotation Cost** The expert effort behind Figure 3 is modest: annotating one trajectory on all 15 rubric dimensions takes about 10 minutes (Appendix A.2.4). The one-time initialization labeled 476 dialogues, or roughly 79 expert-hours; each loop round then reviews a fixed budget of about twenty sampled trajectories (Table 3), i.e., under four expert-hours per round. The entire three-round improvement thus cost about 60 expert-labeled trajectories and ten expert-hours, well within reach of teams already running manual quality assurance.

## 4 Related Work

**User simulator** Early task-oriented dialogue simulators relied on hand-crafted policies such as the agenda-based model (Schatzmann et al., 2007). Neural and LLM-based simulators improve cross-domain fidelity and controllability via transformer simulators, persona/role-playing, dual-LLM generation–verification, and reliability-focused designs (Lin et al., 2021; Shao et al., 2023; Luo et al., 2024; Sekulic et al., 2024). Simulators also teach assistants via multi-round curricula (Kong et al., 2024; Sun et al., 2024), implicit-profile modeling targets long-horizon personalization (Wang et al., 2025), and self-chat post-training further boosts dialogue ability (Xu et al., 2023; Ding et al., 2023). Prior work also explores joint simulator–policy training (Tseng et al., 2021; Chi et al., 2022) and purpose-trained user models (Naous et al., 2026). CoEvoLoop additionally updates a multi-dimensional grader and uses recurring expert calibration.

**RL for customer service** Recent work moves from single-turn rewards to long-horizon, user-centered RL with tool use (Zhao et al., 2025a; Qian et al., 2025), optimizes personalization and proactivity (Zhao et al., 2025b; Sun et al., 2026; Acikgoz et al., 2025), and provides practical training stacks and agent foundations (Xi et al., 2025;

Luo et al., 2025; Li et al., 2025b). Verifiable/process rewards (Lambert et al., 2024; Lightman et al., 2023; Uesato et al., 2022) and segment-level or environment-shaped objectives further stabilize multi-turn learning (Kong et al., 2025; Lu et al., 2025). Policy–reward co-adaptation has also been studied (Shi et al., 2025). CoEvoLoop integrates simulator, grader, and policy updates under expert calibration.

## 5 Findings and Conclusion

### Instruct models are preferable to think models.

Our simulator and grader are both instruct models: their higher inference throughput suits high-concurrency RL rollouts, and when distilling from multiple teachers, final-answer styles are far more consistent than chain-of-thought patterns, which introduce conflicting supervision and destabilize training.

**H2H data distribution is misaligned with the H2M setting.** Training on human-to-human-assistant (H2H) corpora can degrade performance when deployment is human-to-machine-assistant (H2M): H2H dialogue carries implicit assumptions, shorthand, and turn-taking patterns that users do not exhibit toward a system. Collecting H2M data, or transforming H2H data to reflect H2M conditions, is therefore critical for generalization.

**Helpful-assistant LLMs are poor user simulators.** Instruction-tuned LLMs carry a strong “helpful assistant” prior: as simulated users they over-explain, rationalize, or try to solve the task instead of expressing authentic user behaviors such as confusion, impatience, or negative feedback. Targeted fine-tuning with explicit profile and style conditioning is necessary, especially for adversarial probing of assistant failure modes.

**RL training is sensitive to reward design.** Multi-turn RL depends on the choice and weighting of reward components, especially the coefficients of the length and format penalties, which require careful tuning for stable convergence. Above all, it depends on grader calibration: as Table 1 shows, RL against an uncalibrated grader regresses below the SFT baseline under calibrated evaluation.

**Conclusion** We presented CoEvoLoop, a closed-loop RL training system that co-evolves a user simulator, a fine-grained multi-dimensional grader, and an assistant policy through an expert-calibrated,

multi-round loop. Industrial validation, including component ablations under a common calibrated grader and blind expert evaluation, shows that the loop improves service quality, reduces compliance failures, and that co-evolution is load-bearing. We also highlight practical takeaways for scalable deployment, such as preferring instruct models for rollout throughput and matching training data.

## Limitations

**Scale of external validation.** Human validation is at spot-check scale (480 expert ratings); online indicators such as customer satisfaction and escalation-to-human rates are not yet available. **Annotation dependency.** The loop assumes access to qualified domain experts. The recurring cost is small (under four expert-hours per round), but organizations without an existing quality-assurance function would need to bootstrap one, and `strict-aggr@eval` is a custom metric whose cross-paper comparability is limited. **Domain scope and data availability.** Experiments in this paper cover only one proprietary cross-border-payment setting. Customer-derived training and evaluation data are not released due to privacy and confidentiality.

## Ethical Considerations

All training and evaluation data derive from enterprise customer-service logs that are sanitized before any use: personally identifiable information is masked, and user profiles are abstracted representations reverse-engineered from anonymized dialogues rather than raw transcripts. Expert annotators are in-house professionals performing annotation as part of their regular compensated duties and were informed of how their annotations are used. The user simulator imitates human customers strictly as a training and evaluation environment; it is never exposed to real users, and the released demo contains only synthetic or sanitized examples. The trained assistant is designed to operate with compliance safeguards (identity verification, PII masking, promise management) and human escalation paths, not as a replacement for human oversight in high-stakes decisions. The released code is under the MIT license and contains no customer data.

## References

- Emre Can Acikgoz, Jinoh Oh, Jie Hao, Joo Hyuk Jeon, Heng Ji, Dilek Hakkani-Tür, Gokhan Tur, Xiang Li, Chengyuan Ma, and Xing Fan. 2025. [Speakrl: Syn-ergizing reasoning, speaking, and acting in language models with reinforcement learning](#).
- Kevin Chen, Marco Cusumano-Towner, Brody Huval, Aleksei Petrenko, Jackson Hamburger, Vladlen Koltun, and Philipp Krähenbühl. 2025. [Reinforcement learning for long-horizon interactive llm agents](#). *Preprint*, arXiv:2502.01600.
- Dafeng Chi, Yuzheng Zhuang, Yao Mu, Bin Wang, Jianzhu Bao, Yasheng Wang, Yuhao Dong, Xin Jiang, Qun Liu, and Jianye Hao. 2022. [Offline-to-online co-evolutional user simulator and dialogue system](#). In *Proceedings of the Towards Semi-Supervised and Reinforced Task-Oriented Dialog Systems (SereTOD)*, pages 98–113, Abu Dhabi, Beijing (Hybrid). Association for Computational Linguistics.
- Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. [Enhancing chat language models by scaling high-quality instructional conversations](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3029–3051, Singapore. Association for Computational Linguistics.
- Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2025. [A survey on llm-as-a-judge](#). *Preprint*, arXiv:2411.15594.
- Aobo Kong, Wentao Ma, Shiwan Zhao, Yongbin Li, Yuchuan Wu, Ke Wang, Xiaoqian Liu, Qicheng Li, Yong Qin, and Fei Huang. 2025. [Sdpo: Segment-level direct preference optimization for social agents](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.
- Chuyi Kong, Yaxin Fan, Xiang Wan, Feng Jiang, and Benyou Wang. 2024. [Platolm: Teaching llms in multi-round dialogue via a user simulator](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7841–7863. Association for Computational Linguistics.
- Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Øyvind Tafjord, Chris Wilhelm, Luca Soldaini, and 4 others. 2024. [Tulu 3: Pushing frontiers in open language model post-training](#). *arXiv preprint arXiv:2411.15124*.

- J. Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.
- Songze Li, Chuokun Xu, Jiaying Wang, Xueluan Gong, Chen Chen, Jirui Zhang, Jun Wang, Kwok-Yan Lam, and Shouling Ji. 2025a. [Llms cannot reliably judge \(yet?\): A comprehensive assessment on the robustness of llm-as-a-judge](#). *Preprint*, arXiv:2506.09443.
- Weizhen Li, Jianbo Lin, Zhuosong Jiang, Jingyi Cao, Xinpeng Liu, Jiayu Zhang, Zhenqiang Huang, Qianben Chen, Weichen Sun, Qiexiang Wang, Hongxuan Lu, Tianrui Qin, Chenghao Zhu, Yi Yao, Shuying Fan, Xiaowan Li, Tiannan Wang, Pai Liu, King Zhu, and 11 others. 2025b. [Chain-of-agents: End-to-end agent foundation models via multi-agent distillation and agentic rl](#). *arXiv preprint arXiv:2508.13167*.
- Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. [Let’s verify step by step](#). *arXiv preprint arXiv:2305.20050*.
- Hsien-chin Lin, Nurul Lubis, Songbo Hu, Carel van Niekerk, Christian Geisshauser, Michael Heck, Shutong Feng, and Milica Gasic. 2021. [Domain-independent user simulation with transformers for task-oriented dialogue systems](#). In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 445–456, Singapore and Online. Association for Computational Linguistics.
- Siyuan Lu, Zechuan Wang, Hongxuan Zhang, Qintong Wu, Leilei Gan, Chenyi Zhuang, Jinjie Gu, and Tao Lin. 2025. [Don’t just fine-tune the agent, tune the environment](#). *arXiv preprint arXiv:2510.10197*.
- Xiang Luo, Zhiwen Tang, Jin Wang, and Xuejie Zhang. 2024. [Duetsim: Building user simulator with dual large language models for task-oriented dialogues](#). *arXiv preprint arXiv:2405.13028*.
- Xufang Luo, Yuge Zhang, Zhiyuan He, Zilong Wang, Siyun Zhao, Dongsheng Li, Luna K. Qiu, and Yuqing Yang. 2025. [Agent lightning: Train any ai agents with reinforcement learning](#). *arXiv preprint arXiv:2508.03680*.
- Tarek Naous, Philippe Laban, Wei Xu, and Jennifer Neville. 2026. [Flipping the dialogue: Training and evaluating user language models](#). In *International Conference on Learning Representations*.
- Cheng Qian, Zuxin Liu, Akshara Prabhakar, Jielin Qiu, Zhiwei Liu, Haolin Chen, Shirley Kokane, Heng Ji, Weiran Yao, Shelby Heinecke, Silvio Savarese, Caiming Xiong, and Huan Wang. 2025. [Userll: Training interactive user-centric agent via reinforcement learning](#). *arXiv preprint arXiv:2509.19736*.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. [Direct preference optimization: Your language model is secretly a reward model](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Jost Schatzmann, Blaise Thomson, Karl Weilhammer, Hui Ye, and Steve Young. 2007. [Agenda-based user simulation for bootstrapping a pomdp dialogue system](#). In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pages 149–152, Rochester, New York. Association for Computational Linguistics.
- Ivan Sekulic, Silvia Terragni, Victor Guimarães, Nghia Khau, Bruna Guedes, Modestas Filipavicius, Andre Ferreira Manso, and Roland Mathis. 2024. [Reliable LLM-based user simulator for task-oriented dialogue systems](#). In *Proceedings of the 1st Workshop on Simulating Conversational Intelligence in Chat (SCI-CHAT 2024)*, pages 19–35, St. Julians, Malta. Association for Computational Linguistics.
- Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. 2023. [Character-llm: A trainable agent for role-playing](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13153–13187. Association for Computational Linguistics.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. [Deepseekmath: Pushing the limits of mathematical reasoning in open language models](#). *Preprint*, arXiv:2402.03300.
- Tianyuan Shi, Canbin Huang, Fanqi Wan, Longguang Zhong, Ziyi Yang, Weizhou Shen, Xiaojun Quan, and Ming Yan. 2025. [Mutual-taught for co-adapting policy and reward models](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16285–16298, Vienna, Austria. Association for Computational Linguistics.
- Weiwei Sun, Xuhui Zhou, Weihua Du, Xingyao Wang, Sean Welleck, Graham Neubig, Maarten Sap, and Yiming Yang. 2026. [Training proactive and personalized llm agents](#).
- Yuchong Sun, Che Liu, Kun Zhou, Jinwen Huang, Ruihua Song, Wayne Xin Zhao, Fuzheng Zhang, Di Zhang, and Kun Gai. 2024. [Parrot: Enhancing multi-turn instruction following for large language models](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9729–9750. Association for Computational Linguistics.
- Bo-Hsiang Tseng, Yinpei Dai, Florian Kreyssig, and Bill Byrne. 2021. [Transferable dialogue systems and user simulators](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1:*

*Long Papers*), pages 152–166, Online. Association for Computational Linguistics.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. 2022. [Solving math word problems with process- and outcome-based feedback](#). *arXiv preprint arXiv:2211.14275*.

Kuang Wang, Xianfei Li, Shenghao Yang, Li Zhou, Feng Jiang, and Haizhou Li. 2025. [Know you first and be you better: Modeling human-like user simulators via implicit profiles](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Zhiheng Xi, Jixuan Huang, Chenyang Liao, Baodai Huang, Honglin Guo, Jiaqi Liu, Rui Zheng, Junjie Ye, Jiazheng Zhang, Wenxiang Chen, Wei He, Yiwen Ding, Guanyu Li, Zehui Chen, Zhengyin Du, Xuesong Yao, Yufei Xu, Jiecao Chen, Tao Gui, and 4 others. 2025. [Agentgym-rl: Training llm agents for long-horizon decision making through multi-turn reinforcement learning](#).

Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. 2023. [Baize: An open-source chat model with parameter-efficient tuning on self-chat data](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6268–6278. Association for Computational Linguistics.

Qiyang Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, YuYue, Weinan Dai, Tiantian Fan, Gao-hong Liu, Juncai Liu, LingJun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, and 17 others. 2025. [DAPO: An open-source LLM reinforcement learning system at scale](#). In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Weikang Zhao, Xili Wang, Chengdi Ma, Lingbin Kong, Zhaohua Yang, Mingxiang Tuo, Xiaowei Shi, Yitao Zhai, and Xunliang Cai. 2025a. [Mua-rl: Multi-turn user-interacting agent reinforcement learning for agentic tool use](#).

Weixiang Zhao, Xingyu Sui, Yulin Hu, Jiahe Guo, Haixiao Liu, Biye Li, Yanyan Zhao, Bing Qin, and Ting Liu. 2025b. [Teaching language models to evolve with users: Dynamic profile modeling for personalized alignment](#). *arXiv preprint arXiv:2505.15456*.

## A Appendix

- Demo Video URL: <https://youtu.be/sd1fVKKzp1M>
- System Github URL: <https://github.com/ant-intl/CoEvoLoop>

### A.1 Training Details

#### A.1.1 Hyperparameters

| Category  | Hyperparameter         | Value              |
|-----------|------------------------|--------------------|
| Trainer   | Nodes                  | 1                  |
|           | GPUs per node          | 8                  |
|           | Total steps            | 726                |
| Algorithm | Advantage estimator    | GRPO               |
|           | Use KL in reward       | False              |
|           | Learning rate          | $1 \times 10^{-6}$ |
|           | Batch size             | 4                  |
|           | Mini batch size        | 64                 |
|           | Use KL loss            | True               |
|           | KL loss coefficient    | 0.001              |
|           | Optimizer warmup style | Cosine             |
| Rollout   | Warmup steps ratio     | 0.05               |
|           | Mode                   | Async              |
|           | Temperature            | 1.0                |
|           | Top-p                  | 1.0                |
|           | Rollout N              | 8                  |

Table 2: Hyperparameters for RL training.

#### A.1.2 Datasets

| Dataset for user simulator              | amount           |
|-----------------------------------------|------------------|
| Initial SFT dataset                     | 15,312 dialogues |
| Initial DPO dataset                     | 419 pairs        |
| Evaluation benchmark for user simulator | 1000 dialogues   |
| Dataset for grader                      | amount           |
| Labeled dialogues by human              | 476 dialogues    |
| Evaluation benchmark for grader         | 414 dialogues    |
| Annotated dialogues per loop (avg.)     | 20 dialogues     |
| Dataset for Assistant                   | amount           |
| SFT data                                | 500 dialogues    |
| RL data                                 | 2,906 profiles   |
| Evaluation benchmark for assistant      | 241 profiles     |

Table 3: Number of data instances used for training.

## A.2 Evaluation and Reliability

### A.2.1 Turing Score

The Turing Score evaluates whether the user side of a dialogue appears to come from a real customer or from the user simulator. We mix 1000 simulator-generated dialogues with real human-to-human

(H2H) customer-service dialogues and ask annotators to classify the source of the user behavior as “Human”, “Uncertain”, or “Machine”. The *Human-likeness* score is the proportion labeled “Human” or “Uncertain”. This user-side human-versus-machine evaluation is separate from the assistant-side H2H–H2M distribution mismatch discussed in Section 5.

As shown in Table 4, the user simulator achieves a Human-likeness score of 0.958, compared with 0.967 for real human users under this protocol.

| User source    | Human-likeness |
|----------------|----------------|
| Real human     | 0.967          |
| User simulator | 0.958          |

Table 4: User Simulator Turing Test Results. Simulator-generated user behavior approaches the real-user Human-likeness score under this protocol.

### A.2.2 The 15-Dimensional Grader Taxonomy

To align with the reward components described in Section 2, the **Grader** evaluates the assistant across **4 core categories** encompassing 15 distinct dimensions. Table 5 defines the refined scoring logic.

### A.2.3 Cohen’s Kappa

To assess the reliability of our multi-dimensional grader, we measure agreement with human expert annotations using Cohen’s Kappa. For each of the 15 evaluation dimensions, we compute:

$$\kappa = \frac{p_o - p_e}{1 - p_e}$$

where  $p_o$  is the observed agreement and  $p_e$  is the chance agreement, computed on the held-out grader benchmark (414 dialogues). The grader achieves an overall  $\kappa = 0.663$ , which falls in the *substantial agreement* band (0.61–0.80) of the [Landis and Koch \(1977\)](#) scale. This level of agreement ensures that the grader’s reward signal is sufficiently aligned with human judgment for reliable RL training.

### A.2.4 Annotation Protocol and Instructions

All expert annotation is performed by in-house senior customer-service quality-assurance specialists on the Annotation interface (Figure 2). Shown the full dialogue, the driving user profile, and the relevant knowledge-base snapshot, the annotator: (1) labels each of the 15 dimensions strictly per the rubric in Table 5, independent of the grader’s

| Category                              | Evaluation Dimension      | Refined Scoring Logic (Pass = 1, Fail = 0)                                             |
|---------------------------------------|---------------------------|----------------------------------------------------------------------------------------|
| <b>I. Security &amp; Compliance</b>   | 1. Identity Verification  | Must strictly verify user identity (KYC) before disclosing any account-sensitive data. |
|                                       | 2. Data Confidentiality   | No exposure of internal URLs, non-public SOPs, or confidential internal materials.     |
|                                       | 3. PII Protection         | Personally Identifiable Information (phone, email, etc.) must be masked in responses.  |
|                                       | 4. Language Consistency   | The response language must strictly adhere to the user’s input language.               |
|                                       | 5. Promise Management     | Assistant must not commit to actions beyond its authority (e.g., "I will call you").   |
| <b>II. Information Correctness</b>    | 6. Factual Correctness    | All product/policy details must strictly align with the official Knowledge Base (KB).  |
|                                       | 7. Business Relevance     | Directly address the user’s problem without providing irrelevant or distracting info.  |
|                                       | 8. Intent Coverage        | Comprehensively address both explicit requests and latent needs within the session.    |
| <b>III. Logic &amp; Reasonability</b> | 9. Logical Consistency    | Maintain a coherent stance; no contradictory statements throughout the dialogue.       |
|                                       | 10. SOP Adherence         | Strictly follow standard operating procedures for info-gathering and troubleshooting.  |
|                                       | 11. Context Retention     | Demonstrate memory of prior turns to avoid repetitive questioning or loss of context.  |
|                                       | 12. Topic Adaptation      | Smoothly handle transitions when the user switches between different business topics.  |
| <b>IV. Service Quality</b>            | 13. Professional Attitude | Maintain emotional neutrality; avoid offensive, mocking, or robotic language.          |
|                                       | 14. Response Brevity      | Ensure the solution is concise and efficient, avoiding wordy or redundant fillers.     |
|                                       | 15. Educational Empathy   | Use a warm tone to correct user misconceptions and explain complex rules clearly.      |

Table 5: The 15-dimensional Expert Annotation Rubric. The framework is categorized into Security, Information, Logic, and Service Quality to provide fine-grained reward signals for the Grader.

displayed judgment; (2) writes a short critique for every failed dimension; (3) rewrites the offending turn when a compliant alternative exists (used as clean supervision); and (4) flags uncertain cases for adjudication by a lead reviewer against the enterprise policy handbook. Annotating one trajectory takes about 10 minutes.

### A.2.5 Blind Human Evaluation Protocol

For the human evaluation in Table 1, we sample 40 held-out trajectories per system for the six marked systems (all loop variants plus the strongest baseline). Two experts, blind to system identity and to grader scores, independently apply the strict rubric to every trajectory (a trajectory passes only if all 15 dimensions pass), giving  $40 \times 6 \times 2 = 480$  expert ratings in total; the reported human score is the pass rate averaged over the two experts.

## A.3 Additional Experiments

### A.3.1 Ablation Study

| Model Size | Reward Agg              | $L_{\text{buf}}$ | $L_{\text{max}}$ | Avg. strict-agg@eval |
|------------|-------------------------|------------------|------------------|----------------------|
| 8B         | $R_{\text{agg}}$        | 200              | 300              | 0.880                |
| 32B        | $R_{\text{agg}}$        | 500              | 1100             | 0.892                |
| 32B        | $R_{\text{agg-strict}}$ | 200              | 300              | 0.879                |
| 32B        | $R_{\text{agg}}$        | 200              | 300              | <b>0.904</b>         |

Table 6: Ablation Experiments

We conduct an ablation study to examine the impact of model scale, reward aggregation strategy, and sequence length constraints on *strict-agg@eval* performance as shown in Table 6. First, reducing both buffer and maximum sequence lengths (200/300 vs. 500/1100) yields a better score, sug-

gesting that tighter length control may improve optimization stability and reduce noisy long-horizon behaviors. Secondly, comparing  $R_{\text{agg}}$  and  $R_{\text{agg-strict}}$  under similar length settings shows that stricter aggregation does not consistently improve evaluation performance, indicating potential over-constraint or reduced learning flexibility. Lastly, increasing model size from 8B to 32B with comparable reward aggregation improves performance, suggesting that larger models benefit more from the aggregated reward signal.

### A.4 User Profile Example

Below is an example of a user profile used in our system.

```
{
  "Background Description": [
    "The user is currently engaged in cross-border e-commerce business and has funding flow needs between multiple WorldFirst accounts."
  ],
  "Knowledge Gap": [
    "Does not understand that cross-border e-commerce accounts only support receiving funds from online physical goods transactions via partner e-commerce platforms or payment gateways.",
    "Unaware that direct transfers between WorldFirst accounts are not allowed, especially for funds not originating from e-commerce platforms.",
    "Does not know the refund process and specific operation methods if a payment fails or cannot be processed."
  ],
  "Action History": [
    {
      "timestamp": "Unknown",
      "action": "Made a payment from a UK entity WorldFirst account to another WorldFirst account",
      "target": "Payment to a specific WorldFirst account",
      "result": "Payment not received",
      "details": "Paid two transactions of 14 EUR"
    }
  ],
  "Problem Description": [
    "The user attempted to make a payment from a UK entity WorldFirst account to another WorldFirst account, but the funds were not received",
    "The user wants to know if the payment can be refunded and what the timeframe for the refund is."
  ]
}
```