

SAGE: An Extensible Benchmark for Service Agent Graph-guided Evaluation

Ling Shi^{1*} Yuqin Dai^{3*} Ziyin Wang¹ Wei Zhang⁵ Ning Gao⁶ Chaozheng Wang⁷
Xinwei Wu² Yujie Wang⁸ Wei He⁸ Jinpeng Wang^{4†} Linlong Xu² Deyi Xiong^{1,9‡}

¹Tianjin University ²Alibaba Group ³Tsinghua University ⁴IQuest Research Ubio Team
⁵BUPT ⁶Beihang University ⁷CUHK ⁸Independent Researcher ⁹Yuanhui AI
{shiling_100, dyxiong}@tju.edu.cn

Abstract

LLMs are transforming customer service, yet evaluating service agents across diverse business scenarios remains costly and fragmented, as each new domain requires building a dedicated benchmark from scratch. Existing benchmarks rely on static, scenario-specific scripts with two fundamental limitations: **Scenario Silos**, where each domain demands costly independent annotation with no shared evaluation structure, and **Dialogue Dynamics Gap**, where fixed scripts fail to capture the full range of real user behaviors. We propose **SAGE** (Service Agent Graph-guided Evaluation), to our knowledge the first scenario-extensible benchmark to jointly assess logical SOP compliance and conversational quality, abstracting any SOP as a directed graph and bridging Scenario Silos through a lightweight two-step extension mechanism. SAGE further enables dynamic multi-turn simulation via adversarial user agents to overcome the Dynamics Gap. Across six industrial scenarios representative of real-world customer service deployment, we evaluate 27 mainstream LLMs spanning both closed-source systems and open-source families, offering a fine-grained capability map to guide real-world model selection. Our experiments further reveal two systematic failure patterns shared across all tested models: while models generally succeed at recognizing user intent, they frequently struggle to follow complex business logic and execute the correct subsequent action; and they sustain a warm and empathetic conversational tone even when their underlying procedural reasoning has already failed. **Code and resources are available at: <https://github.com/lingshi6565/SAGE-Bench>.**

1 Introduction

Large Language Models (LLMs) (Bai et al., 2023; Touvron et al., 2023; Wiggins and Tejani, 2022;

* Equal contribution.

† Project leader.

‡ Corresponding author.

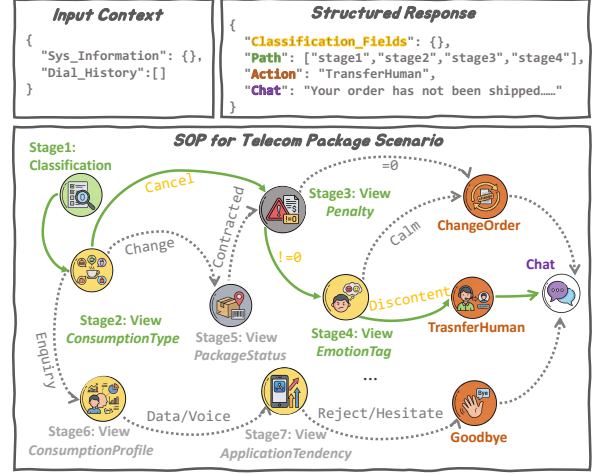


Figure 1: SOP Example for Service Agent. SOPs govern interactions across diverse industries, yet existing benchmarks remain domain-specific. Recognizing that any SOP can be abstracted as a directed graph, we propose SAGE as a scenario-extensible evaluation framework.

Yang et al., 2026; Sun et al., 2024a) are rapidly transforming customer service, enabling enterprises to replace rigid rule-based systems with adaptive service agents (Xi et al., 2025; Wang et al., 2024; Yao et al., 2022; Shinn et al., 2023; Dai et al., 2026; Zhang et al., 2026; Gao et al., 2026). For real-world deployment across high-stakes domains such as healthcare, education, and commerce, these agents must strictly follow enterprise Standard Operating Procedures (SOPs) that govern every interaction, as illustrated in Figure 1, since poor SOP compliance can lead to serious consequences including information leakage and legal liability. Benchmarking their ability to comply with SOPs while maintaining conversational quality has therefore become a critical priority (Cui et al., 2017; Ou et al., 2024; Wang et al., 2025).

Existing benchmarks (Qin et al., 2023; Li et al., 2023; Liu et al., 2023; Budzianowski et al., 2018; Rastogi et al., 2020; Zheng et al., 2023; Quan et al., 2020; Li et al., 2025b) approach this prob-

lem by constructing static, scenario-specific evaluation scripts, each manually annotated for a single domain against a fixed set of predefined dialogue paths. While straightforward, this reliance on static scripts produces two fundamental limitations. **Scenario Silos**: scripts are annotated per domain with no shared structure, making extension to any new scenario prohibitively expensive. As enterprises span multiple business verticals and SOPs evolve continuously, this per-domain construction model cannot scale. **Dialogue Dynamics Gap**: static scripts fix a finite set of dialogue paths, yet real users are unpredictable and the same context can unfold in many ways. A script can only test what its authors anticipated, so benchmark scores systematically overestimate real-world performance.

To solve these issues, we propose **SAGE** (Service Agent Graph-guided Evaluation), a scenario-extensible benchmark for evaluating service agents along two axes: *logical SOP compliance* and *conversational quality*. To address *Scenario Silos*, we observe that any SOP, regardless of domain, can be abstracted as a directed graph, and design SAGE around this unified graph-based representation. A modular *Extension Mechanism* further allows any new domain to be onboarded in two steps, keeping deployment costs low as SOPs evolve. To address *Dialogue Dynamics Gap*, we replace fixed scripts with dynamic simulation driven by an *Adversarial Intent Taxonomy*, instantiating a User Agent with diverse intents and personas to generate multi-turn trajectories that cover the full SOP graph. Unlike prior benchmarks that assess task completion or dialogue quality in isolation, SAGE jointly evaluates both through a hybrid agentic-and-rule-based verification mechanism, enabling precise error localization at each turn. Experiments across 27 mainstream LLMs and six industrial scenarios yield a comprehensive capability map for real-world model selection. In summary, our contributions are:

- We propose **SAGE**, a scenario-extensible benchmark that abstracts any SOP as a directed graph, enabling cross-domain evaluation without per-domain annotation and allowing any new-scenario onboarding in two steps.
- We design an Adversarial Intent Taxonomy that drives dynamic multi-turn simulation across three adversarial user intents.
- We introduce the first dual-axis evaluation

framework for service agents, jointly assessing logical compliance and chat quality via a hybrid agentic-and-rule-based mechanism.

- Experiments across 27 mainstream LLMs and six industrial scenarios reveal two systematic cross-model phenomena: **Execution Gap** and **Empathy Resilience**, providing a fine-grained capability map to guide real-world model selection.

2 Related Works

2.1 Evaluation for Large Language Models

LLM evaluation (Guo et al., 2023; Peng et al., 2026; Liu et al., 2024b) has progressed from instruction-following (Zhou et al., 2023a) and format compliance (Xia et al., 2024; Tang et al., 2024) to complex multi-constraint settings (He et al., 2024; Jiang et al., 2023; Qin et al., 2024; Diao et al., 2025; Yao et al., 2023). Domain-specific benchmarks cover business process management (Berti et al., 2024; Fahland et al., 2025; Kourani et al., 2025b; Fournier et al., 2024; Rebmann et al., 2024; Kourani et al., 2025a; Grohs et al., 2023), finance (Xie et al., 2024, 2023), and procedural SOP compliance (Li et al., 2025b; Nandi et al., 2025), yet focus on tool-calling sequences rather than deep logical reasoning. Broader agent frameworks (Zhou et al., 2023b; Drouin et al., 2024), reasoning benchmarks (Liu et al., 2020; Dua et al., 2019; Yang et al., 2018; Liu et al., 2025), and training methods (Feng et al., 2024; Zhang et al., 2024; Zeng et al., 2024; Madaan et al., 2023; Wang et al., 2023; Ouyang et al., 2022) further advance the field. SAGE is the first service-agent benchmark to jointly assess *logical compliance* and *chat quality* across all service domains without per-domain annotation.

2.2 Benchmarks for Service Dialogue Systems

Service dialogue benchmarks cover human-like conversation (Ou et al., 2024), customer support resolution (Wang et al., 2025), multi-turn shopping interactions (Bernard and Balog, 2023; Li et al., 2025a; Sun et al., 2024b), and real-world dialogue analysis (Zhu et al., 2025; Zheng et al., 2023; Zhang et al., 2025). Agent evaluation also extends to specialized dialogue domains such as diplomacy (Peng et al., 2025). These benchmarks are typically tied to fixed, scenario-specific SOPs, limiting cross-domain adaptability. SAGE breaks this constraint by abstracting any SOP as a directed

graph and providing a modular, intent-based extension mechanism that enables rapid, code-free adaptation to new domains in two steps.

3 Methodology

To overcome *Scenario Silos*, SAGE employs **Universal Dialogue Graph Modeling** (§3.1), formalizing any SOP as a directed graph to enable domain-agnostic evaluation without high-cost annotation. To overcome *Dialogue Dynamics Gap*, SAGE introduces **Dynamic Multi-Turn Dialogue Simulation** (§3.2), replacing fixed scripts with adversarial user agents that generate trajectories covering the full SOP graph. Unlike prior benchmarks, **Dual-Axis Evaluation with Hybrid Verification** (§3.3) jointly assesses *logical compliance* and *chat quality* per turn for precise error localization. These three components, together with a **Scenario Extension Mechanism** (§3.4) for low-cost domain adaptation, form the pipeline illustrated in Figure 2.

3.1 Universal Dialogue Graph Modeling

Real-world SOPs are written as unstructured natural language documents, making automated evaluation ambiguous and non-deterministic. To provide a rigorous, domain-agnostic foundation, we formalize every SOP as a directed graph, enabling deterministic ground-truth generation and systematic path coverage across all service scenarios. We formalize each SOP as a directed graph $G = (V, E)$ with three node types: *Start/End*, *Decision* (branching on conditions), and *Action* nodes. Edges encode transitions triggered by *Classification Fields* $\mathcal{F}$ (categorical attributes the agent must identify, e.g., order type or user emotion) and *System Information* $\mathcal{S}$ (factual context shared across all parties, e.g., order status or payment record). $\mathcal{S}$ is propagated throughout the entire pipeline: it shapes the User Agent’s persona to prevent hallucinations, guides the Service Agent’s decisions, and grounds the Rule Engine’s ground truth, ensuring all three components operate on a consistent factual basis. With the graph established, we next use it to drive dynamic dialogue simulation.

3.2 Dynamic Multi-Turn Dialogue Simulation

Static scripts fix a finite set of paths, leaving most real user behaviors untested. To close this *Dialogue Dynamics Gap*, we generate trajectories via dynamic User Agent–Service Agent interaction, with the User Agent equipped with diverse intents and personas to cover the full SOP graph.

User Agent generates user responses based on personas and dialogue context:

$$a_t = \text{UserAgent}(h_{<t}, s_t, \mathcal{S}, \mathcal{I}, \mathcal{P}). \quad (1)$$

a_t is conditioned on five inputs: dialogue history $h_{<t}$, agent state s_t (a control signal tracking whether the conversation has been opened, is ongoing, or should terminate), system information $\mathcal{S}$ (grounding the user’s factual knowledge to prevent hallucinations), user intent $\mathcal{I}$ (e.g., refund vs. inquiry), and persona $\mathcal{P}$ (emotional state and cooperativeness). $\mathcal{I}$ and $\mathcal{P}$ are drawn from the Adversarial Intent Taxonomy (§ 3.4) and dynamically instantiated to simulate diverse adversarial intents.

Service Agent is the evaluation target tasked with navigating the SOP graph to generate a structured response. The output at turn t , denoted as b_t , comprises four distinct components:

$$\begin{aligned} b_t &= \text{ServiceAgent}(\mathcal{S}, h_{<t}, a_t, G) \\ &= \{p_t, \text{action}_t, \mathcal{F}_t, \text{chat}_t\}. \end{aligned} \quad (2)$$

The four output components are: traversal path p_t (planned graph nodes sequence), action_t (selected operation at the end node), classification fields $\mathcal{F}_t$ (categorical judgments, e.g., user emotion), and natural language chat_t (user-facing response). Correctly identifying $\mathcal{F}_t$ is the core challenge, as it determines p_t and action_t on G , making every decision step observable and comparable against ground truth. The agents interact until termination, and the resulting (a_t, b_t) trajectory is passed to the evaluation stage described next.

3.3 Graph-Guided Multi-Agent Evaluation

SAGE evaluates trajectories via **Path Coverage** (ensuring all SOP branches are tested), a **Judge Agent** (extracting classification labels and quality scores), and a **Rule Engine** (cross-validating against graph logic), jointly assessing *Logical Compliance* and *Chat Quality*.

3.3.1 Path Coverage

To guarantee full logic coverage, we predefine all valid trajectories $\mathcal{P}$ via intent-balanced sampling plus targeted supplementation for under-covered paths. Rare branches are covered by an inverse configuration mechanism that deterministically synthesizes the required intents and system states, ensuring every edge case is tested.

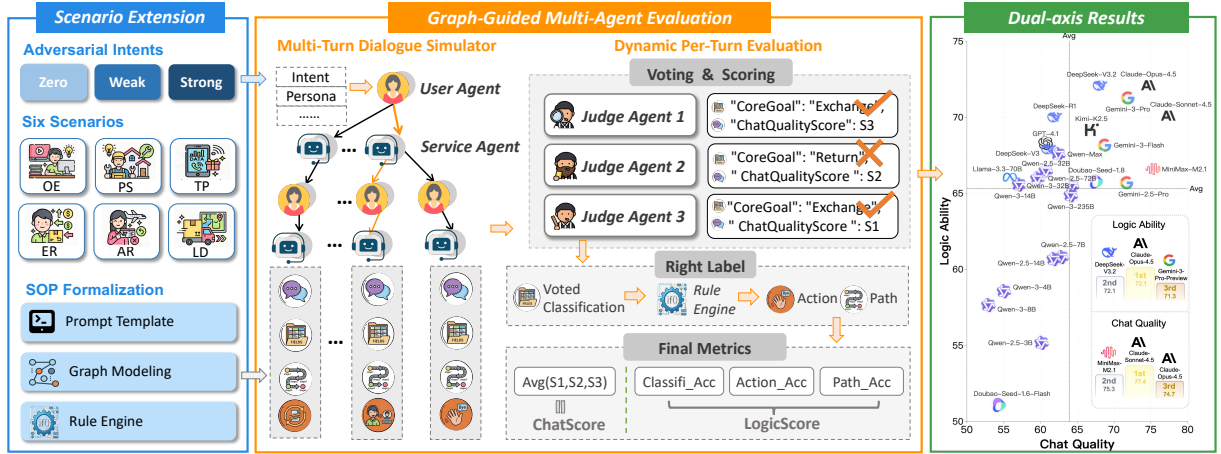


Figure 2: Overview of SAGE evaluation framework.

3.3.2 Judge Agent

Since the Rule Engine captures only procedural correctness, the Judge Agent handles semantic understanding of natural language. For each turn t , it analyzes $(\mathcal{S}, h_{<t}, a_t, \text{chat}_t)$ and outputs classification ground truth $\mathcal{F}_t^*$ and quality score s_{chat} :

$$(\mathcal{F}_t^*, s_{\text{chat}}) = \text{JudgeAgent}(\mathcal{S}, h_{<t}, a_t, \text{chat}_t). \quad (3)$$

To ensure robustness against individual model bias, we employ an ensemble of three Judge Agents. We apply majority voting to determine the consensus ground truth $\mathcal{F}_t^*$ of classification fields, while the quality score s_{chat} is derived from the average rating. The classification ground truth $\mathcal{F}_t^*$ serves as the input for the Rule Engine, while s_{chat} directly quantifies the conversational quality. Human validation on 100 sampled turns shows 91.3% average agreement between three annotators and the ensemble, confirming label reliability (Appendix A.1).

3.3.3 Rule Engine

While the Judge Agent handles semantics, it cannot guarantee procedural correctness on its own: LLM outputs are probabilistic and may be inconsistent across runs. The Rule Engine addresses this by deterministically traversing the SOP graph given $\mathcal{F}_t^*$. It receives the consensus classification $\mathcal{F}_t^*$, system information $\mathcal{S}$, and graph G , and computes the unique correct execution path and action:

$$(p_t^*, \text{action}_t^*) = \text{RuleEngine}(\mathcal{F}_t^*, \mathcal{S}, G). \quad (4)$$

p_t^* and action_t^* serve as the deterministic reference standard for logical compliance evaluation. Note that while the Rule Engine itself is fully deterministic given fixed inputs, its upstream $\mathcal{F}_t^*$ carries the

inherent stochasticity of LLM judges; the ensemble voting step is designed to minimize this variance. Together, the Judge Agent and Rule Engine provide complementary ground truth: the former for chat quality, the latter for procedural correctness.

3.3.4 Dual-Axis Metrics Evaluation

With ground truth from both the Judge Agent and Rule Engine, we compare the service agent’s outputs against these standards along two axes, enabling precise localization of defects.

Logical Compliance Evaluation. The overall Logical Compliance Score combines three submetrics as $\text{Score}_{\text{logic}} = 0.4 \text{Acc}_{\text{cls}} + 0.4 \text{Sim}_{\text{path}} + 0.2 \text{Acc}_{\text{action}}$, where:

(1) Classification Accuracy measures alignment between predicted $\mathcal{F}_t$ and ground truth $\mathcal{F}_t^*$:

$$\text{Acc}_{\text{cls}} = |\mathcal{F}|^{-1} \sum_{f \in \mathcal{F}} \mathbb{I}[\mathcal{F}_t(f) = \mathcal{F}_t^*(f)]. \quad (5)$$

(2) Path Correctness measures the overlap between the agent’s planned path and the Rule Engine’s reference path:

$$\text{Sim}_{\text{path}} = \frac{|p_t \cap p_t^*|}{|p_t^*|}. \quad (6)$$

(3) Action Correctness verifies if the agent’s final executed action matches the Rule Engine’s reference action:

$$\text{Acc}_{\text{action}} = \mathbb{I}[\text{action}_t = \text{action}_t^*]. \quad (7)$$

Chat Quality Evaluation. Three Judge Agents independently score each response across five dimensions (Linguistic Quality, Anthropomorphism,

Content Utility, User Satisfaction, Instruction Compliance), with scores averaged across judges:

$$\text{Score}_{\text{quality}} = \frac{1}{3} \sum_{j=1}^3 \left(\sum_{k=1}^5 s_{j,k} \right). \quad (8)$$

Overall Assessment Score. The final per-turn score integrates both axes, weighting logic more heavily to reflect the primacy of procedural correctness in service scenarios:

$$\text{Score}_{\text{overall}} = 0.8 \times \text{Score}_{\text{logic}} + 0.2 \times \text{Score}_{\text{quality}}. \quad (9)$$

For multi-turn dialogues, we sample turns 1, 5, 10, 15, and the final turn to measure performance across conversation depths. This weighted integration prioritizes procedural rigor while accounting for user experience, facilitating the precise diagnosis of model defects across specific dimensions. The weight sensitivity analyse of these metrics (Appendix A.5) confirms that the current weighting scheme exhibits strong robustness in evaluation outcomes. We additionally report **Format-Conditioned Logic Scores** computed only on successfully parsed outputs, isolating reasoning deficits from JSON serialization failures. Beyond robust evaluation, SAGE is engineered for scalability. We next detail how its modular design supports rapid adaptation to arbitrary scenarios.

3.4 Scenario Extension Mechanism

Intent-based Adversarial Scenario Taxonomy. To comprehensively simulate realistic user behaviors, we define a taxonomy along two dimensions: *goal alignment* (the degree to which the user’s demand matches what the service agent can provide) and *emotional state* (the intensity of the user’s aggression or urgency). This yields three adversarial levels. **Zero-adversarial** intents are fully cooperative: the user has a clear, standard request and a friendly tone, primarily testing the agent’s basic procedural execution. **Weak-adversarial** intents introduce friction via complex constraints (e.g., an unpaid bill that blocks a refund) or ambiguous needs, requiring the agent to resolve contextual conflicts without facing direct hostility. **Strong-adversarial** intents involve emotional dissatisfaction or emergencies (e.g., a safety hazard), where the user employs aggressive strategies to demand immediate resolution, rigorously testing the agent’s negotiation and de-escalation capabilities.

Extension Mechanism. The taxonomy above is injected into the User Agent via prompt engineering over $\mathcal{I}$ and $\mathcal{P}$ (details in Appendix B), enabling diverse, high-fidelity simulation. Building on this, adding a new service domain to SAGE requires only two declarative steps: (1) formalizing the domain’s SOP as a directed graph, and (2) defining the user persona profile for that domain. This modular design fully decouples scenario configuration from the evaluation engine, keeping deployment costs low as business SOPs evolve. Beyond evaluation, the same architecture supports automated dialogue data synthesis, enabling large-scale generation of training corpora for customer-service LLMs.

4 Experiments

4.1 Experiment Settings

We evaluate 27 representative Large Language Models (LLMs) across closed-source and open-source families, covering diverse scales and architectures. Closed-source models are accessed via stable official APIs, including the Claude series (Sonnet-4.5, Opus-4.5), GPT-4.1 (Achiam et al., 2023), the Gemini series (2.5-Pro, 3-Pro/Flash) (Team et al., 2023), Qwen-Max (Bai et al., 2023), and the Doubao series. Open-source models are deployed locally with vLLM (Kwon et al., 2023), including the Qwen2.5 and Qwen3 families (Bai et al., 2023), the DeepSeek series (V3, V3.2, R1) (Liu et al., 2024a), the Llama-3 series (Dubey et al., 2024), GLM-4.7 (Zeng et al., 2022), Kimi-K2.5 (Team et al., 2025), and MiniMax-M2.1 (Chen et al., 2025). This selection supports comparison across proprietary systems, open-source models, scales, and architectures.

To assess the generalization capability of service agents across industrial domains, we construct six customer service scenarios with different levels of SOP complexity, ranging from standardized procedural inquiries to complex, high-stakes disputes. We denote these scenarios as Online Education (OE), E-commerce Refund (ER), Telecom Package (TP), Property Service (PS), Logistics Delivery (LD), and Airline Refund (AR). They share a unified configuration structure but differ in task-specific fields, actions, and SOP logic. Detailed descriptions are in Appendix B.1.1.

4.2 Overall Model Performance

We first examine overall model performance across the six scenarios. In addition to the primary metrics,

Table 1: Main results across 6 scenarios (0-100 Scale). *OA*: Overall Assessment Score; *Logic*: Logical Compliance Score; *Chat*: Conversational Quality Score. The superscripts indicate the ranking within Closed-Source and Open-Source groups, respectively. *Format Error*: Percentage of outputs failing JSON parsing. *Chat Length*: Average character count of the response field.

Model	AVG Score			OA on 6 Scenarios						Format	Chat
Name	OA	Logic	Chat	ER	TP	PS	LD	AR	OE	Error	Length
<i>Closed-Source Large Language Models</i>											
Claude-Sonnet-4.5	71.56 ²	70.10 ³	77.38 ¹	71.21	75.03	75.07	67.30	66.52	74.24	0.75 %	93.42
Claude-Opus-4.5	72.62 ¹	72.10 ¹	74.68 ²	72.73	73.12	73.25	71.25	70.12	75.22	0.94 %	43.49
GPT-4.1	66.79 ⁶	68.31 ⁴	60.70 ⁸	70.32	66.90	65.46	62.69	63.68	71.69	0.04 %	18.59
Gemini-2.5-Pro	66.90 ⁵	65.69 ⁸	71.73 ⁴	69.46	66.42	69.60	64.92	60.71	70.28	0.00 %	53.33
Gemini-3-Pro-Preview	71.40 ³	71.27 ²	71.89 ³	70.99	69.09	73.65	68.05	64.45	82.15	0.00 %	48.60
Gemini-3-Flash-Preview	68.28 ⁴	68.16 ⁵	68.75 ⁵	67.56	65.36	71.62	67.10	62.40	75.62	8.92 %	50.16
Qwen-Max	66.54 ⁷	67.57 ⁶	62.44 ⁷	66.81	65.47	64.36	64.03	63.85	74.75	0.00 %	27.55
Doubao-Seed-1.6-Flash	51.75 ⁹	51.08 ⁹	54.43 ⁹	54.18	51.59	47.34	46.98	47.37	63.04	0.49 %	18.83
Doubao-Seed-1.8	66.14 ⁸	65.78 ⁷	67.62 ⁶	70.25	65.89	69.03	64.01	62.19	65.49	0.04 %	26.79
<i>Open-Source Large Language Models</i>											
Qwen2.5-3B-Instruct	56.16 ¹⁷	55.16 ¹⁷	60.15 ¹²	68.92	55.16	54.53	46.01	59.65	52.70	0.02 %	26.76
Qwen2.5-7B-Instruct	61.22 ¹³	60.80 ¹³	62.86 ⁶	69.05	62.58	60.19	47.24	60.88	67.35	0.00 %	19.44
Qwen2.5-14B-Instruct	60.90 ¹⁴	60.67 ¹⁴	61.80 ⁸	67.64	63.33	58.12	54.24	62.96	59.11	0.00 %	27.23
Qwen2.5-32B-Instruct	65.34 ⁷	66.49 ⁷	60.74 ¹¹	72.07	68.27	62.26	62.99	60.25	66.20	0.00 %	21.18
Qwen2.5-72B-Instruct	65.27 ⁸	65.51 ¹¹	64.30 ⁴	64.90	69.16	65.78	63.16	59.83	68.76	0.00 %	44.67
Qwen3-4B	57.86 ¹⁵	58.56 ¹⁵	55.05 ¹⁶	64.71	62.72	56.57	44.36	56.47	62.31	0.04 %	24.12
Qwen3-8B	56.71 ¹⁶	57.65 ¹⁶	52.93 ¹⁷	51.88	62.80	54.03	47.77	57.67	66.08	0.00 %	19.93
Qwen3-14B	63.88 ¹²	65.57 ¹⁰	57.11 ¹⁴	63.79	68.01	64.81	56.13	63.78	66.74	0.43 %	25.54
Qwen3-32B	64.69 ¹⁰	65.97 ⁹	59.58 ¹³	67.94	65.48	66.71	58.79	61.66	67.56	1.48 %	24.15
Qwen3-235B-A22B	64.72 ⁹	64.87 ¹²	64.13 ⁵	66.73	64.97	61.47	64.77	62.37	68.02	1.34 %	32.85
Deepseek-V3.2	71.29 ¹	72.08 ¹	68.11 ²	71.51	70.61	70.71	70.87	65.84	78.17	0.00 %	27.48
Deepseek-V3	66.54 ⁶	67.90 ⁵	61.11 ¹⁰	69.62	71.60	67.57	60.08	61.58	68.77	0.00 %	18.50
Deepseek-R1	68.39 ³	70.00 ²	61.94 ⁷	70.16	70.39	70.58	65.75	64.03	69.40	0.97 %	38.08
GLM-4.7	67.25 ⁵	68.68 ⁴	61.52 ⁹	68.14	66.02	69.64	66.50	60.82	72.36	1.46 %	17.58
Kimi-K2.5	68.71 ²	69.16 ³	66.93 ³	69.69	69.41	71.84	65.74	64.45	71.15	0.52 %	34.17
MiniMax-M2.1	68.33 ⁴	66.59 ⁶	75.29 ¹	68.33	70.12	64.07	68.54	66.37	72.55	0.82 %	61.27
Llama-3.1-8B-Instruct	37.62 ¹⁸	37.36 ¹⁸	38.64 ¹⁸	49.58	55.83	37.58	28.26	38.14	16.30	44.33 %	44.70
Llama-3.3-70B-Instruct	64.02 ¹¹	66.06 ⁸	55.85 ¹⁵	68.60	64.74	63.87	60.27	62.93	63.71	0.03 %	13.79

including Logical Compliance Score, Chat Quality Score, and Overall Assessment Score, we report two auxiliary indicators: Format Error Rate and Average Chat Length. The former quantifies JSON parsing failures as a proxy for instruction-following stability, while the latter measures response verbosity and potential redundancy. Building on the evaluation setup above, Table 1 presents the comprehensive results across all models and scenarios.

Superiority of Closed-Source Models and the Rising Open-Source Challengers. Closed-source models still define the frontier: Claude-Opus-4.5 achieves the highest OA score (72.62¹), supported by top logical reasoning (72.10¹) and impressive chat quality (74.68²). However, the performance gap between proprietary and open-weight models is remarkably narrow. DeepSeek-V3.2 reaches an open-source OA of 71.29¹, surpassing GPT-4.1 and Gemini-3-Pro-Preview. This indicates that state-

of-the-art open-source architectures have reached a level of maturity capable of handling complex, graph-guided service logic previously reserved for proprietary systems.

Decoupling Logical Compliance and Conversational Quality. Results show a trade-off between procedural rigor and conversational quality. Claude-Opus-4.5 leads in logic, while Claude-Sonnet-4.5 ranks first in Chat Quality (77.38¹), suggesting a more empathetic persona. MiniMax-M2.1 is another outlier: despite moderate Logic (66.59⁶), it attains the best open-source Chat score (75.29¹). This suggests that certain models are specifically optimized for human-like interaction, which can occasionally compensate for minor procedural deviations in terms of overall user perception. In contrast, DeepSeek-R1 and GPT-4.1 lean heavily toward logic-first strategies, often at the expense of conversational warmth.

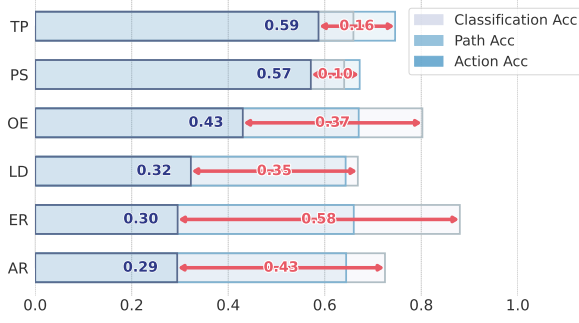


Figure 3: Logic performance analysis across scenarios.

The Dynamics of Response Verbosity and Stability. Response length helps indicate service quality, but only up to a threshold. Strong models such as Claude-Sonnet-4.5 (93.42 chars) and MiniMax-M2.1 (61.27 chars) give more detailed guidance and empathetic de-escalation, which matter in complex scenarios like ER and AR. Smaller models often fail catastrophically: Llama-3.1-8B-Instruct has a 44.33% Format Error, failing basic JSON output. Gemini-3-Flash-Preview remains competitive despite a higher error rate (8.92%), suggesting efficient reasoning when formatting succeeds.

4.3 Cross-Scenario Difficulty Analysis

SAGE provides a standardized framework to quantify scenario complexity (Figure 3). We define difficulty by the “Execution Gap”: the gap between identifying task-relevant fields and selecting the final action, visualized as the disparity between nested bars. In complex scenarios such as ER, AR, LD, and OE, *Classification_Acc* (Light Gray) and *Action_Acc* (Dark Blue) diverge sharply. This reveals a Logic Deduction Barrier: even with correct fields ($\mathcal{F}_t$), models often fail to infer the next action, showing that semantic classification alone does not ensure procedural execution under complex SOP dependencies. In more standardized scenarios such as PS and TP, *Path_Acc* (Light Blue) can exceed *Classification_Acc* (Light Gray). Since Path Accuracy measures the overlap between predicted and ground-truth paths, models can earn partial credit by traversing early, simpler steps even when they miss harder classifications. Thus, *Path_Acc* captures macroscopic procedural progress rather than final correctness alone. Together with stricter *Classification_Acc* and *Action_Acc*, it forms a diagnostic profile covering both coarse SOP following and fine-grained reasoning.

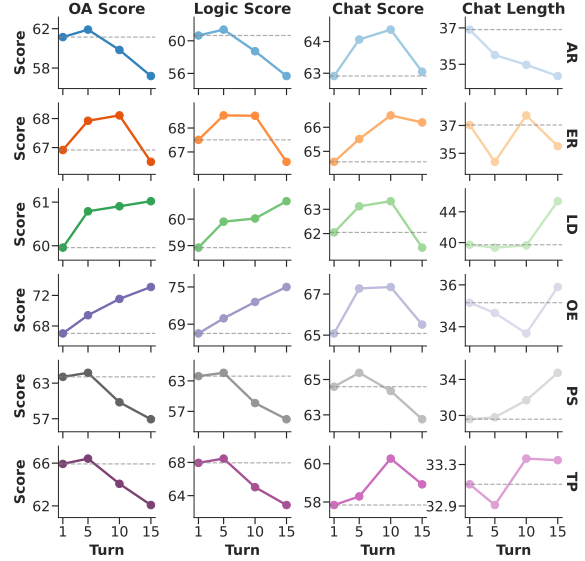


Figure 4: Performance evolution across dialogue turns.

4.4 Multi-turn Robustness Analysis

After aggregate and scenario-level analyses, we examine robustness as dialogue context accumulates. Figure 4 shows OA, Logic, and Chat scores at Turns 1, 5, 10, and 15. Most scenarios (e.g., AR, ER, PS, TP) exhibit an “Inverted-U” trajectory: scores peak around Turn 5 and drop by Turn 15. This reflects two phases. During the Information Gain Phase (Turns 1–5), performance improves (e.g., ER OA increases from ~ 67 to ~ 68) as multi-turn interactions overcome cold-start deficits by gathering critical details like order IDs and complaints, enabling accurate intent classification and SOP navigation. Conversely, the Context Fatigue Phase (Turns 10–15) sees performance decline (e.g., PS OA drops from approximately 63 to 57) as accumulated dialogue noise exposes LLMs’ long-context limitations, triggering the “Lost in the Middle” phenomenon where models hallucinate or lose SOP state tracking.

4.5 Impact of Adversarial Intensity

To validate our Adversarial Scenario Taxonomy, we compare model performance across three adversarial levels of user intent. Figure 5 shows box plots for Overall Score, Logic Score, Chat Quality, and Format Error Rate. To rigorously substantiate these claims, we non-parametric Kruskal-Wallis H tests across intensity levels, detailed in Table 2.

Logic Degradation and Error Escalation. As adversarial intensity increases, Logical Compliance consistently declines ($p = 2.30 \times 10^{-12}$), confirm-

Table 2: Kruskal-Wallis H tests across adversarial intensity levels. Values are reported as Mean $\pm$ SD.

Intensity	Overall Score	Logic Ability	Chat Quality	JSON Error (%)
Zero	0.68 $\pm$ 0.12	0.69 $\pm$ 0.14	0.63 $\pm$ 0.09	1.25 $\pm$ 4.14
Weak	0.66 $\pm$ 0.09	0.67 $\pm$ 0.10	0.63 $\pm$ 0.08	1.81 $\pm$ 6.54
Strong	0.60 $\pm$ 0.12	0.60 $\pm$ 0.12	0.61 $\pm$ 0.13	3.65 $\pm$ 15.79
p -value	2.21×10^{-11}	2.30×10^{-12}	0.71 (n.s.)	0.064

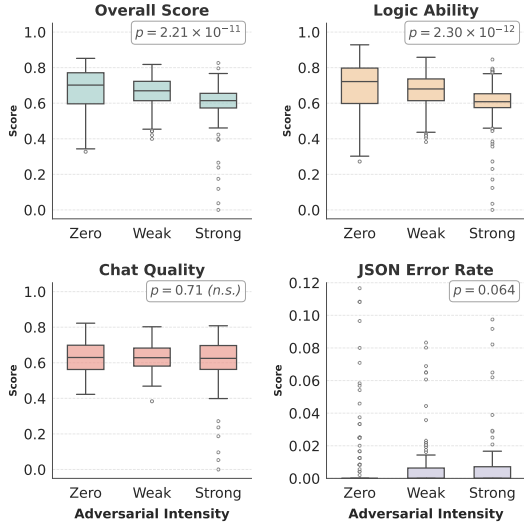


Figure 5: Performance distribution across intent type.

ing that behaviors triggered by high adversarial intent, including information concealment and emotional aggression, create logical friction and hinder SOP adherence. Meanwhile, the JSON Error Rate rises in Strong-Adversarial scenarios, with its mean nearly tripling from 1.25% to 3.65% and approaching statistical significance ($p = 0.064$), suggesting that high cognitive or emotional pressure induces instruction drift and invalid JSON outputs.

The Stability in Chat Quality. This metric remains consistently robust across adversarial intensities, with no statistically significant variation ($p = 0.71$), reflecting an “Empathy Resilience” that decouples dialogue from logic: models sustain polite, de-escalating facades even as user aggression impairs logical compliance. Such dissociation validates dual-axis evaluation to separate surface fluency from procedural robustness.

Increased Discriminative Power. Score distributions also become more dispersed as intensity increases, with larger interquartile ranges and more outliers. While models perform similarly in Zero-Adversarial settings, Strong-Adversarial scenarios widen the gap between top- and low-tier models, making high intensity a stronger filter for robust agentic capability.

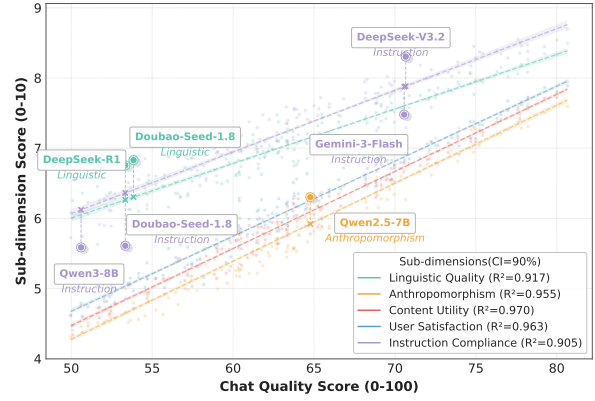


Figure 6: Correlation analysis between the Chat Quality Score and its five sub-dimensions. The deviations from the regression lines (e.g., Doubao-Seed-1.8’s high Linguistic vs. low Instruction scores) highlight specific model characteristics.

4.6 Sub-Metric Diagnostics

Finally, we test whether SAGE is not only a scoring benchmark but also a diagnostic tool. We correlate the aggregate Chat Quality Score with its five constituent Sub-dimension Scores. Figure 6 supports two uses: validating sub-metric consistency and fine-grained capability profiling.

High Metric Consistency. All five dimensions exhibit strong linear correlations with the *Chat Quality Score* ($R^2 > 0.9$). This statistical coherence validates our metric design by confirming each dimension’s reliable contribution. In complex multi-agent evaluation, such high linearity precludes non-linear distortions (e.g., S-shaped curves) or significant noise in any dimension, which would otherwise signal flawed scoring rubrics or LLM-as-judge hallucinations and inconsistencies. The absence of these anomalies confirms that the SAGE framework effectively encapsulates the multi-faceted nature of conversational ability, maintaining high interpretability and minimal bias across all predefined dimensions.

Capability Profiling via Residual Analysis. Beyond the linear trend, residuals identify model-specific strengths and weaknesses. Points above the *Linguistic Quality* line indicate exceptional flu-

Table 3: Model Performance Comparison across Different Metrics.

Model	Professional (Contrast)	Conv. User	Conv. Service	Conv. Judge	Rearranged Criteria
Deepseek-V3.2	78.17	77.92	78.05	77.88	78.10
GLM-4.7	72.36	72.11	72.28	71.95	72.33
Kimi-K2.5	71.15	70.98	71.02	70.87	71.20
Qwen3-32B	67.56	67.21	67.40	67.05	67.48
Gemini-3-Pro-Preview	82.15	81.88	82.02	81.76	82.09

ency; Doubao-Seed-1.8 and DeepSeek-R1 show such positive residuals. A point significantly deviating from the mean regression line indicates that a model’s capability in that specific dimension is disproportionate to its overall performance. Points below the *Instruction Compliance* line indicate difficulty following constraints; Doubao-Seed-1.8 and Qwen3-8B fall notably below this trend. This reveals a capability imbalance: several models exhibit high linguistic articulation yet frequently overlook formatting or constraints. Such granular analysis proves SAGE’s capacity to diagnose subtle alignment trade-offs, effectively distinguishing merely chatty models from strictly compliant ones.

4.7 Prompt Sensitivity

To address prompt sensitivity concerns, we conducted a controlled experiment testing four variations (conversational User/Service/Judge Agent prompts, and rearranged criteria order) on five models in the Online Education (OE) scenario. The results are shown in Table 3.

Score fluctuations remained minimal across all variations with an average deviation of 0.25 points. Especially rearranging the criteria order had exceedingly slight impact on the scores. Conversational Judge prompts caused a uniform, slight score decrease, yet relative model rankings and performance trends remained perfectly consistent.

These findings prove that our evaluation outcomes are structurally grounded in the SOP graph and deterministic rule-based verification. The framework is not sensitive to minor prompt variations, making it a robust and highly reproducible benchmark for the research community.

5 Conclusion

We presented **SAGE**, a scenario-extensible benchmark that addresses *Scenario Silos* and *Dialogue Dynamics Gap* by formalizing SOPs as directed graphs and coupling them with adversarial multi-turn simulation. A hybrid agentic-and-rule-based

mechanism enables dual-axis evaluation of logical compliance and conversational quality per turn. Experiments across 27 mainstream LLMs and six industrial scenarios surface three cross-model phenomena: an Execution Gap where correct intent recognition does not guarantee correct action execution, Empathy Resilience where conversational quality remains stable despite logical failures under adversarial pressure, and an Inverted-U trajectory where performance peaks at mid-dialogue before declining under extended context. These findings provide a fine-grained capability map for real-world model selection and underscore the necessity of dual-axis, multi-turn evaluation in high-stakes service deployment.

Limitations

While SAGE advances the evaluation of service agents, several limitations remain. SAGE currently focuses on text-based, single-modality service interactions; extending to multimodal scenarios (e.g., image-based complaints or voice dialogues) is a natural next step. The benchmark also covers customer service SOPs, and applying the graph-guided framework to other procedural domains such as medical triage or legal consultation would require domain-specific SOP formalization, which we leave for future work. While SAGE evaluates agent behavior at inference time, it does not directly address how benchmark findings can be used to improve models via fine-tuning or reinforcement learning, which is seen as a promising direction.

Acknowledgments

The present research was supported by the National Key Research and Development Program of China (Grant No. 2024YFE0203000) and the Tianjin Science and Technology Major Project (Grant No. 24ZXKJGX00070). We would like to thank the anonymous reviewers for their insightful comments.

References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, and 1 others. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.
- Nolwenn Bernard and Krisztian Balog. 2023. Mgsopdial: A multi-goal conversational dataset for e-commerce. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2775–2785.
- Alessandro Berti, Humam Kourani, and Wil MP van der Aalst. 2024. Pm-llm-benchmark: Evaluating large language models on process mining tasks. In *International Conference on Process Mining*, pages 610–623. Springer.
- Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. Multiwoz—a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. *arXiv preprint arXiv:1810.00278*.
- Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu, Chao Wang, Cheng Zhu, and 1 others. 2025. [Minimax-m1: Scaling test-time compute efficiently with lightning attention](#). *ArXiv*, abs/2506.13585.
- Lei Cui, Shaohan Huang, Furu Wei, Chuanqi Tan, Chaoyun Duan, and Ming Zhou. 2017. Superagent: A customer service chatbot for e-commerce websites. In *Proceedings of ACL 2017, system demonstrations*, pages 97–102.
- Yue Dai, Ning Gao, Wei Zhang, Jie Wang, Luozichen, Ruiyuan Wu, Jinteng Wang, and Chaozheng Wang. 2026. [SEAD: Self-evolving agent for multi-turn service dialogue](#). In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 3674–3684, San Diego, California, United States. Association for Computational Linguistics.
- Lingxiao Diao, Xinyue Xu, Wanxuan Sun, Cheng Yang, and Zhuosheng Zhang. 2025. Guidebench: Benchmarking domain-oriented guideline following for llm agents. *arXiv preprint arXiv:2505.11368*.
- Alexandre Drouin, Maxime Gasse, Massimo Caccia, Issam H Laradji, Manuel Del Verme, Tom Marty, Léo Boisvert, Megh Thakkar, Quentin Cappart, David Vazquez, and 1 others. 2024. Workarena: How capable are web agents at solving common knowledge work tasks? *arXiv preprint arXiv:2403.07718*.
- Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. *arXiv preprint arXiv:1903.00161*.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. [The Llama 3 Herd of Models](#). *arXiv e-prints*, arXiv:2407.21783.
- Dirk Fahland, Fabiana Fournier, Lior Limonad, Inna Skarbovska, and Ava JE Swevels. 2025. How well can a large language model explain business processes as perceived by users? *Data & Knowledge Engineering*, 157:102416.
- Peiyuan Feng, Yichen He, Guanhua Huang, Yuan Lin, Hanchong Zhang, Yuchen Zhang, and Hang Li. 2024. [Agile: A novel reinforcement learning framework of llm agents](#). *Advances in Neural Information Processing Systems* 37.
- Fabiana Fournier, Lior Limonad, and Inna Skarbovska. 2024. Towards a benchmark for causal business process reasoning with llms. In *International Conference on Business Process Management*, pages 233–246. Springer.
- Ning Gao, Wei Zhang, Yubin Dai, Ling Shi, Ziyin Wang, Yujie Wang, Wei He, Jinteng Wang, and Chaozheng Wang. 2026. Reinforcing real-world service agents: Balancing utility and cost in task-oriented dialogue. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*.
- Michael Grohs, Luka Abb, Nourhan Elsayed, and Jana-Rebecca Rehse. 2023. Large language models can accomplish business process management tasks. In *International conference on business process management*, pages 453–465. Springer.
- Zishan Guo, Renren Jin, Chuang Liu, Yufei Huang, Dan Shi, Supryadi, Linhao Yu, Yan Liu, Jiaxuan Li, Bojian Xiong, and Deyi Xiong. 2023. [Evaluating large language models: A comprehensive survey](#). *Preprint*, arXiv:2310.19736.
- Yun He, Di Jin, Chaoqi Wang, Chloe Bi, Karishma Mandyam, Hejia Zhang, Chen Zhu, Ning Li, Tengyu Xu, Hongjiang Lv, Shruti Bhosale, Chenguang Zhu, Karthik Abinav Sankararaman, Eryk Helenowski, Melanie Kambadur, Aditya Tayade, Hao Ma, Han Fang, and Sinong Wang. 2024. [Multi-if: Benchmarking llms on multi-turn and multilingual instructions following](#). *Preprint*, arXiv:2410.15553.
- Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjuan Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. 2023. [Follow-bench: A multi-level fine-grained constraints following benchmark for large language models](#). *ArXiv*, abs/2310.20410.
- Humam Kourani, Alessandro Berti, Jasmin Hennrich, Wolfgang Kratsch, Robin Weidlich, Chiao-Yun Li, Ahmad Arslan, Wil MP van der Aalst, and Daniel

- Schuster. 2025a. Leveraging large language models for enhanced process model comprehension. *Decision Support Systems*, page 114563.
- Humam Kourani, Alessandro Berti, Daniel Schuster, and Wil MP Van der Aalst. 2025b. Evaluating large language models on business process modeling: framework, benchmark, and self-improvement analysis: H. kourani et al. *Software and Systems Modeling*, pages 1–36.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626.
- Minghao Li, Yingxiu Zhao, Bowen Yu, Feifan Song, Hangyu Li, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li. 2023. Api-bank: A comprehensive benchmark for tool-augmented llms. *arXiv preprint arXiv:2304.08244*.
- Xiangci Li, Zhiyu Chen, Jason Ingyu Choi, Nikhita Vedula, Besnik Fetahu, Oleg Rokhlenko, and Shervin Malmasi. 2025a. **Wizard of shopping: Target-oriented E-commerce dialogue generation with decision tree branching**. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13095–13120, Vienna, Austria. Association for Computational Linguistics.
- Zekun Li, Shinda Huang, Jiangtian Wang, Nathan Zhang, Antonis Antoniadis, Wenyue Hua, Kaijie Zhu, Sirui Zeng, Chi Wang, William Yang Wang, and 1 others. 2025b. Sopbench: Evaluating language agents at following standard operating procedures and constraints. *arXiv preprint arXiv:2503.08669*.
- Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.
- Chuang Liu, Linhao Yu, Jiaxuan Li, Renren Jin, Yufei Huang, Ling Shi, Junhui Zhang, Xinmeng Ji, Tingting Cui, Tao Liu, Jinwang Song, Hongying Zan, Sun Li, and Deyi Xiong. 2024b. **OpenEval: Benchmarking chinese llms across capability, alignment and safety**. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, ACL 2024, Bangkok, Thailand, August 11-16, 2024, pages 190–210. Association for Computational Linguistics.
- Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2020. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. *arXiv preprint arXiv:2007.08124*.
- Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, and 1 others. 2023. Agent-bench: Evaluating llms as agents. *arXiv preprint arXiv:2308.03688*.
- Yan Liu, Renren Jin, Ling Shi, Zheng Yao, and Deyi Xiong. 2025. **FineMath: A fine-grained mathematical evaluation benchmark for chinese large language models**. *ACM Trans. Asian Low Resour. Lang. Inf. Process.*, 24(12):139:1–139:15.
- Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegrefe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Sean Welleck, Bodhisattwa Prasad Majumder, Shashank Gupta, Amir Yazdanbakhsh, and Peter Clark. 2023. **Self-refine: Iterative refinement with self-feedback**. *ArXiv*, abs/2303.17651.
- Subhrangshu Nandi, Arghya Datta, Nikhil Vichare, Indranil Bhattacharya, Huzefa Raja, Jing Xu, Shayan Ray, Giuseppe Carenini, Abhi Srivastava, Aaron Chan, and 1 others. 2025. Sop-bench: Complex industrial sops for evaluating llm agents. *arXiv preprint arXiv:2506.08119*.
- Jiao Ou, Junda Lu, Che Liu, Yihong Tang, Fuzheng Zhang, Di Zhang, and Kun Gai. 2024. **DialogBench: Evaluating LLMs as human-like dialogue systems**. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6137–6170, Mexico City, Mexico. Association for Computational Linguistics.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.
- Jianxiang Peng, Junhao Li, Hongxiang Wang, Haocheng Lyu, Hui Guo, Siyi Hao, Zhen Wang, Chuang Liu, Shaowei Zhang, Bojian Xiong, Yue Chen, Zhuowen Han, Ling Shi, Tianyu Dong, Juesi Xiao, Lei Yang, Yuqi Ren, and Deyi Xiong. 2026. **Dep: A decentralized large language model evaluation protocol**. *Preprint*, arXiv:2603.01167.
- Jianxiang Peng, Ling Shi, Xinwei Wu, Hanwen Zhang, Fujian Liu, Haocheng Lyu, and Deyi Xiong. 2025. **Diplomacyagent: Do llms balance interests and ethical principles in international events?** In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 13721–13739. Association for Computational Linguistics.
- Yiwei Qin, Kaiqiang Song, Yebowen Hu, Wenlin Yao, Sangwoo Cho, Xiaoyang Wang, Xuansheng Wu, Fei Liu, Pengfei Liu, and Dong Yu. 2024. Infobench: Evaluating instruction following ability in large language models. *arXiv preprint arXiv:2401.03601*.

- Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, and 1 others. 2023. Toolllm: Facilitating large language models to master 16000+ real-world apis. *arXiv preprint arXiv:2307.16789*.
- Jun Quan, Shian Zhang, Qian Cao, Zizhong Li, and Deyi Xiong. 2020. Risawoz: A large-scale multi-domain wizard-of-oz dataset with rich semantic annotations for task-oriented dialogue modeling. *arXiv preprint arXiv:2010.08738*.
- Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8689–8696.
- Adrian Rebmann, Fabian David Schmidt, Goran Glavač, and Han van Der Aa. 2024. Evaluating the ability of llms to solve semantics-aware process mining tasks. In *2024 6th International Conference on Process Mining (ICPM)*, pages 9–16. IEEE.
- Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36:8634–8652.
- Haoran Sun, Renren Jin, Shaoyang Xu, Leiyu Pan, Supryadi, Menglong Cui, Jiangcun Du, Yikun Lei, Lei Yang, Ling Shi, Juesi Xiao, Shaolin Zhu, and Deyi Xiong. 2024a. [FuxiTranyu: A multilingual large language model trained with balanced data](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: EMNLP 2024 - Industry Track, Miami, Florida, USA, November 12-16, 2024*, pages 1499–1522. Association for Computational Linguistics.
- Yuchong Sun, Che Liu, Kun Zhou, Jinwen Huang, Ruihua Song, Wayne Xin Zhao, Fuzheng Zhang, Di Zhang, and Kun Gai. 2024b. 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.
- Xiangru Tang, Yiming Zong, Jason Phang, Yilun Zhao, Wangchunshu Zhou, Arman Cohan, and Mark Gerson. 2024. Struc-bench: Are large language models good at generating complex structured tabular data? In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 12–34.
- Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, and 1 others. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.
- Kimi Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, and 1 others. 2025. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.
- Haixin Wang, Xianhan Peng, Huang Cheng, Yizhe Huang, Ming Gong, Chenghan Yang, Yang Liu, and Jiang Lin. 2025. [ECom-bench: Can LLM agent resolve real-world E-commerce customer support issues?](#) In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 276–284, Suzhou (China). Association for Computational Linguistics.
- Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, and 1 others. 2024. A survey on large language model based autonomous agents. *Frontiers of computer science*, 18(6):186345.
- Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. [Self-instruct: Aligning language models with self-generated instructions](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.
- Walter F Wiggins and Ali S Tejani. 2022. On the opportunities and risks of foundation models for natural language processing in radiology. *Radiology: Artificial Intelligence*, 4(4):e220119.
- Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwon Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, and 1 others. 2025. The rise and potential of large language model based agents: A survey. *Science China information sciences*, 68(2):121101.
- Congying Xia, Chen Xing, Jiangshu Du, Xinyi Yang, Yihao Feng, Ran Xu, Wenpeng Yin, and Caiming Xiong. 2024. Fofu: A benchmark to evaluate llms’ format-following capability. *arXiv preprint arXiv:2402.18667*.
- Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, and 1 others. 2024. Finben: A holistic financial benchmark for large language models. *Advances in Neural Information Processing Systems*, 37.
- Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. 2023. Pixiu: A large language model, instruction data and evaluation benchmark for finance. *arXiv preprint arXiv:2306.05443*.

Lei Yang, Leiyu Pan, Bojian Xiong, Renren Jin, Shaowei Zhang, Yue Chen, Ling Shi, Jiang Zhou, Junru Wu, Zhen Wang, Jianxiang Peng, Juesi Xiao, Tianyu Dong, Zhuowen Han, Zhuo Chen, Yuqi Ren, and Deyi Xiong. 2026. [From curated data to scalable models: Continual pre-training of dense and MoE large language models for Tibetan](#). In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 40167–40182, San Diego, California, United States. Association for Computational Linguistics.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2369–2380.

Shunyu Yao, Howard Chen, Austin W. Hanjie, Runzhe Yang, and Karthik Narasimhan. 2023. [Collie: Systematic construction of constrained text generation tasks](#). *ArXiv*, abs/2307.08689.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*.

Aohan Zeng, Mingdao Liu, Rui Lu, Bowen Wang, Xiao Liu, Yuxiao Dong, and Jie Tang. 2024. Agenttuning: Enabling generalized agent abilities for llms. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3053–3077.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, and 1 others. 2022. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*.

Azure Zhang, Ning Gao, Yuqin Dai, Ruiyuan Wu, Jinpeng Wang, Rena Wei Gao, Bingdong Tan, Shuzheng Gao, Zongjie Li, and Chaozheng Wang. 2026. [Unlocking proactivity in task-oriented dialogue](#). *arXiv preprint arXiv:2605.22240*.

Junjie Zhang, Ruobing Xie, Yupeng Hou, Xin Zhao, Leyu Lin, and Ji-Rong Wen. 2025. Recommendation as instruction following: A large language model empowered recommendation approach. *ACM Transactions on Information Systems*, 43(5):1–37.

Wenqi Zhang, Ke Tang, Hai Wu, Mengna Wang, Yongliang Shen, Guiyang Hou, Zeqi Tan, Peng Li, Yueting Zhuang, and Weiming Lu. 2024. Agent-pro: Learning to evolve via policy-level reflection and optimization. *arXiv preprint arXiv:2402.17574*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Tianle Li, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zhuohan Li, Zi Lin, Eric P Xing, and 1 others. 2023. Lmsys-chat-1m: A large-scale real-world llm conversation dataset. *arXiv preprint arXiv:2309.11998*.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Sid-dhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023a. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, and 1 others. 2023b. Webarena: A realistic web environment for building autonomous agents. *arXiv preprint arXiv:2307.13854*.

Jie Zhu, Huaixia Dou, Junhui Li, Lifan Guo, Feng Chen, Chi Zhang, and Fang Kong. 2025. Evaluating, synthesizing, and enhancing for customer support conversation. *arXiv preprint arXiv:2508.04423*.

A Experiment Results Supplementary

This appendix provides supplementary data substantiating our main findings. We first validate our multi-agent ensemble via a Single-Judge Bias ablation study. Next, we present granular turn-level analysis to detail multi-turn robustness, followed by a breakdown of performance shifts under varying adversarial intensities. Finally, we decompose Logic and Chat scores into sub-metrics, quantifying the “Execution Gap” and capability imbalances.

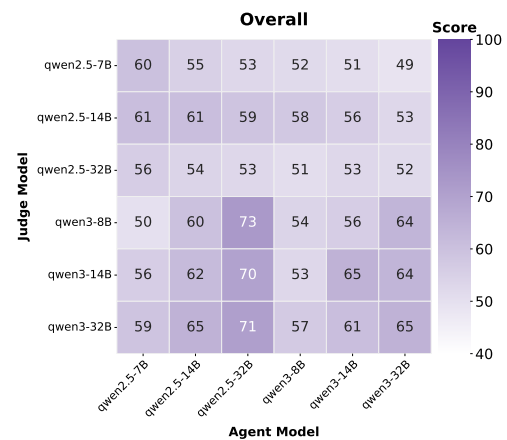


Figure 7: Overall Average Score Heatmap. Rows represent Judge models, columns represent Agent models.

A.1 Impact of Single-Judge Bias

To validate the robustness of our evaluation mechanism, we conducted an ablation study to investigate the bias inherent in using a single Large Language Model (LLM) as a judge. Specifically, we selected a subset of models (from the Qwen2.5 and Qwen3 families) to act as both Service Agents and Judge Agents in a round-robin evaluation setup.

The results are visualized in three heatmaps: **Overall Average Score (Figure 7)**, **Chat Quality**

(Figure 9), and **Logic Ability** (Figure 8). In these plots, the Y-axis represents the *Judge Model* and the X-axis represents the *Evaluated Agent*. This experiment reveals two critical limitations of single-judge frameworks:

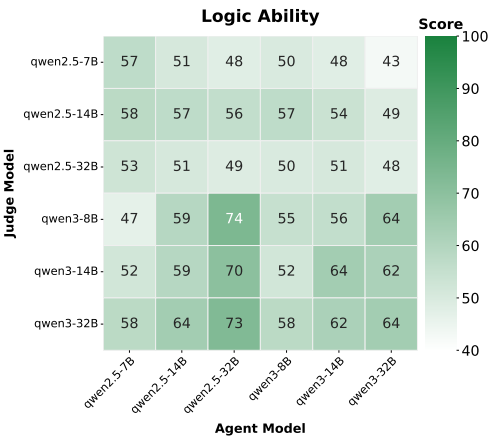


Figure 8: Logic Ability Heatmap. Horizontal variances indicate differing judge strictness.

1. Egocentric Bias (Self-Preference). A distinct diagonal dominance is observable, particularly in **Figure 9 (Chat Quality)**. Models tend to assign higher scores to their own outputs (or outputs from the same model family) compared to external evaluators. For instance, the diagonal cells in Figure 9 often exhibit deeper colors than the off-diagonal cells in the same column, indicating that a model favors response styles similar to its own training distribution.

2. Systematic Scoring Bias. Significant horizontal variations exist across all three heatmaps, especially in **Figure 8 (Logic Ability)**. This indicates that different judges possess different strictness standards. Some judges (represented by rows with consistently lighter colors) act as strict graders, systematically assigning lower scores across all agents, while others (darker rows) are more lenient. For example, *Qwen3-8B* as a judge might exhibit a different scoring distribution compared to *Qwen2.5-32B*.

Justification for Multi-Agent Ensemble. These findings demonstrate that relying on a single judge introduces significant variance and bias, where the evaluation outcome is heavily contingent on the specific evaluator’s preferences rather than the intrinsic quality of the service agent. To mitigate this, SAGE employs an **Ensemble of Three Judge Agents** by using majority voting for classification and average scoring for quality.

The three Judges are drawn from disjoint vendors so that majority voting cancels any single-model self-preference: GPT-4.1, Doubao-Seed-1.6-flash, DeepSeek-V3. This collaborative approach effectively smooths out individual model biases, cancels out extreme scoring tendencies, and ensures a more objective and robust evaluation standard, as reflected in our main experimental results. To further validate the reliability of the ensemble’s ground truth labels, we conducted a blind human validation study on 100 randomly sampled multi-turn dialogue turns. Three independent human annotators evaluated each turn, and their agreement with the ensemble’s majority-vote classification reached 92%, 89%, and 93%, respectively, yielding an average human-AI agreement of **91.3%**. This strong alignment confirms that ensemble voting successfully mitigates individual model biases, and that Logic Score improvements in our benchmark genuinely reflect better SOP compliance rather than alignment with judge model artifacts.

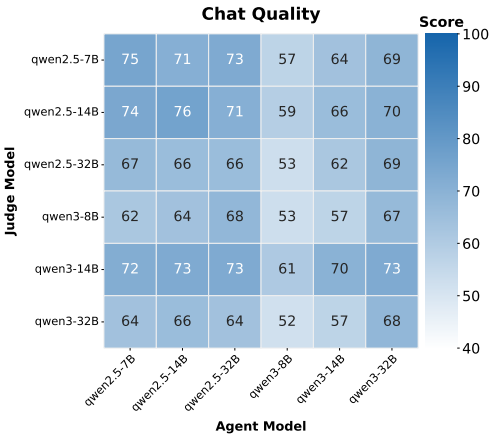


Figure 9: Chat Quality Heatmap. Darker diagonals indicate self-preference bias.

A.2 Supplementary Analysis of Turn

This chapter supplements the content of Section 4.4. Table 4 provides a granular view of model performance evolution across dialogue turns (T1, T5, T10, T15). Consistent with the “Inverted-U” trajectory discussed in the main text, most scenarios (e.g., AirlineRefund, PropertyService) exhibit a performance peak around Turn 5, followed by a decline at Turn 15.

- Context Accumulation (T1 to T5):** The initial improvement in Logic Score (e.g., TelecomPackage: 68.0 to 68.5) indicates that models effectively gather user information in early

turns to clarify intent.

- **Context Fatigue (T10 to T15):** The subsequent drop (e.g., PropertyService Logic: 58.6 to 55.5) highlights the difficulty of maintaining logical consistency over long contexts.
- **Chat Stability:** Interestingly, Chat Scores often remain stable or decline less than Logic Scores (e.g., EcommerceRefund Chat: 66.5 to 66.2), suggesting that models maintain linguistic fluency even when their procedural reasoning falters.

A.3 Supplementary Analysis of Adversarial Intensity

This chapter supplements the content of Section 4.4. Table 5 details the impact of adversarial intensity on model performance. Kruskal-Wallis H tests validate the performance shifts across adversarial intensities, statistically confirming our claims (Table 2).

- **Logic Degradation:** As expected, Logic Scores generally decrease as intensity rises from Zero to Strong (e.g., LogisticsDelivery: 84.0 to 54.5), confirming that adversarial user behaviors successfully challenge the agent’s reasoning capabilities.
- **Format Error Escalation:** The JSON Error rate consistently increases with intensity (Average: 1.2% to 3.7%), indicating that high-pressure scenarios induce “instruction drift,” causing models to violate output formatting constraints.
- **Scenario Specifics:** In some scenarios like AirlineRefund, performance actually improves from Zero to Weak, likely because the “Zero” setting involves trivial queries that some powerful models might over-complicate, whereas “Weak” scenarios provide clearer task structures.

A.4 Supplementary Analysis of Correlation Analysis of Sub-Metrics

This chapter supplements the content of Section 4.6. Figure 10 provides the distribution of the three adversarial intent types across different scenarios. Table 6 breaks down the Logic and Chat scores into their constituent sub-dimensions.

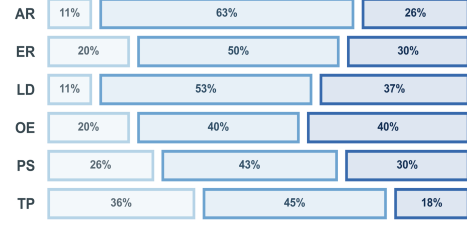


Figure 10: Distribution of three adversarial intent types across scenarios.

- **The Execution Gap:** A significant disparity exists between Classification Accuracy (Avg: 72.9) and Action Correctness (Avg: 41.7). This gap is most pronounced in EcommerceRefund (88.0 vs. 29.5), proving that while models are adept at understanding intent, they struggle to execute the correct sequence of actions in complex SOPs.
- **Chat Quality Imbalance:** Within Chat Quality, Linguistic Quality (Avg: 69.7) and Instruction Compliance (Avg: 71.8) are high, whereas Anthropomorphism (Avg: 57.3) is notably lower. This suggests that current LLMs are polite and compliant but lack the “human touch” or empathy required for high-quality customer service.

A.5 Weight Sensitivity Analyse

To validate the robustness of our evaluation metric design, we conduct a weight sensitivity analysis examining how variations in the weighting scheme affect model rankings. Our composite score is defined as:

$$S_{\text{overall}} = w_{\text{logic}} \cdot S_{\text{logic}} + (1 - w_{\text{logic}}) \cdot S_{\text{chat}} \quad (10)$$

where $S_{\text{logic}} = 0.4 \cdot S_{\text{cls}} + 0.4 \cdot S_{\text{path}} + 0.2 \cdot S_{\text{finals}}$ and S_{chat} represents the normalized mean of five dialogue quality dimensions. The baseline configuration uses $w_{\text{logic}} = 0.8$.

Top-Level Weight Sensitivity. We sweep w_{logic} from 0 to 1 and measure ranking stability using Kendall’s τ correlation with the baseline ranking and the number of pairwise rank flips. As shown in Figure 11, the ranking remains highly stable (Kendall’s $\tau \geq 0.9$) within the range $w_{\text{logic}} \in [0.73, 0.86]$, demonstrating that our choice of $w_{\text{logic}} = 0.8$ is not an isolated optimum but lies within a broad stability plateau. Even at $w_{\text{logic}} = 0.7$, only 19 out of 351 possible model

Table 4: Turn-by-Turn Performance Analysis by Scenario. (0-100 Scale).

Scenario	OA Score				Logic Score				Chat Score				Chat Length			
	T1	T5	T10	T15	T1	T5	T10	T15	T1	T5	T10	T15	T1	T5	T10	T15
AirlineRefund	61.1	61.9	59.8	57.2	60.7	61.4	58.7	55.7	63.1	64.1	64.3	63.2	37	36	35	34
EcommerceRefund	66.9	67.9	68.1	66.5	67.5	68.5	68.5	66.6	64.6	65.5	66.5	66.2	37	34	38	36
LogisticsDelivery	59.5	60.5	60.7	60.8	58.9	59.9	60.0	60.7	62.0	63.1	63.3	61.4	40	39	40	45
OnlineEducation	67.0	69.4	71.5	73.1	67.5	69.9	72.6	75.0	65.1	67.3	67.3	65.5	35	35	34	36
PropertyService	64.1	64.7	59.8	56.9	63.9	64.6	58.6	55.5	64.6	65.4	64.4	62.8	30	30	32	35
TelecomPackage	65.9	66.4	64.1	62.1	68.0	68.5	65.0	62.9	57.8	58.3	60.3	58.9	33	33	33	33

Table 5: Adversarial Intensity Analysis: Average Performance Across All Models (0-100 Scale).

Scenario	Overall			Logic			Chat Quality			JSON Error (%)		
	Zero	Weak	Strong	Zero	Weak	Strong	Zero	Weak	Strong	Zero	Weak	Strong
AirlineRefund	49.7	63.2	60.9	46.3	63.2	60.3	63.3	63.0	63.4	1.6	1.8	3.4
E-commerceRefund	73.1	68.8	58.9	74.8	69.7	58.1	66.2	65.2	62.0	1.0	1.5	3.9
LogisticsDelivery	78.2	59.9	56.4	84.0	59.5	54.5	55.2	61.6	63.6	0.6	2.0	3.8
OnlineEducation	75.7	74.4	62.6	78.1	76.7	62.0	66.1	65.1	64.9	2.3	2.9	4.5
PropertyService	65.1	61.4	67.0	64.4	60.3	68.8	67.9	65.9	60.0	0.9	1.0	3.1
TelecomPackage	66.2	69.5	54.8	67.7	72.6	55.2	60.1	57.1	53.3	1.0	1.6	3.1
Average	68.0	66.2	60.1	69.2	67.0	59.8	63.1	63.0	61.2	1.2	1.8	3.7

Table 6: Sub-dimension Analysis: Average Performance Across All Models (0-100 Scale). Path: Path Correctness; Finals: Finals Correctness; Class.: Classification Accuracy; Ling.: Linguistic Quality; Anth.: Anthropomorphism; Cont.: Content Utility; Satis.: User Satisfaction; Instr.: Instruction Compliance (0-100 Scale).

Metric	AR	ER	LD	OE	PS	TP	Avg.
<i>Logic Ability</i>							
Path	64.4	66.0	64.3	67.0	67.2	74.6	67.3
Action	29.4	29.5	32.3	43.0	57.1	58.7	41.7
Class.	72.5	88.0	66.9	80.2	64.0	65.9	72.9
<i>Chat Quality</i>							
Ling.	68.3	70.1	68.9	71.6	71.3	67.9	69.7
Anth.	56.2	58.8	55.5	62.3	58.3	52.8	57.3
Cont.	60.6	62.1	58.6	60.0	60.8	52.1	59.0
Satis.	62.0	64.0	60.0	61.7	62.1	53.9	60.6
Instr.	73.1	71.4	71.6	73.0	75.0	66.3	71.8

pairs (5.4%) experience rank flips, confirming that the relative performance conclusions are robust to moderate perturbations of the top-level weight.

Logic Ability Internal Weight Sensitivity. We further examine the sensitivity of the internal weighting within S_{logic} by performing a grid sweep over the simplex $\{(w_{\text{cls}}, w_{\text{path}}, w_{\text{finals}}) \mid w_{\text{cls}} + w_{\text{path}} + w_{\text{finals}} = 1\}$. Figure 12 presents the Kendall’s τ and rank flip counts as heatmaps parameterized by w_{cls} and w_{path} (with $w_{\text{finals}} = 1 - w_{\text{cls}} - w_{\text{path}}$). The results indicate that the ranking is stable across a wide region of the weight simplex, with high τ values (≥ 0.8) maintained

throughout most of the feasible space. This confirms that the specific choice of (0.4, 0.4, 0.2) does not disproportionately favor any particular model.

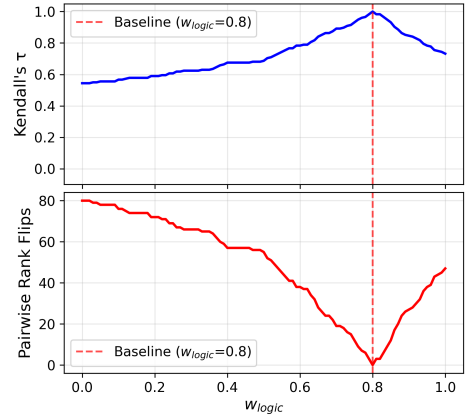


Figure 11: Top-level weight sensitivity analysis. **Upper:** Kendall’s τ correlation between rankings under varying w_{logic} and the baseline ranking ($w_{\text{logic}} = 0.8$). **Lower:** Number of pairwise rank flips. The red dashed line indicates the baseline weight. Rankings remain stable ($\tau \geq 0.9$) within $w_{\text{logic}} \in [0.73, 0.86]$.

B Scenario Extending

B.1 Scenario Configuration Template

Our benchmark includes six customer service scenarios, each following a unified configuration structure. The detailed components, their specific fields, and data formats are summarized in Table 7. Clas-

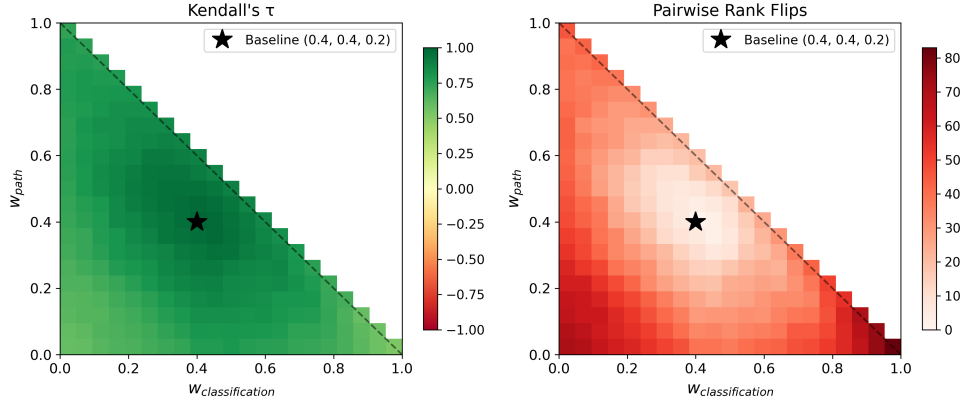


Figure 12: Logic ability internal weight sensitivity. **Left:** Kendall’s τ heatmap over the weight simplex (w_{cls} , w_{path}). **Right:** Pairwise rank flip counts. The black star marks the baseline configuration (0.4, 0.4, 0.2). The dashed diagonal represents the constraint boundary $w_{\text{cls}} + w_{\text{path}} = 1$. High ranking stability is observed across the majority of the feasible weight space.

sification Fields (Classifi_Fields) are task-specific fields that the agent must classify based on dialogue context and system information. Actions means possible actions that the service agent can take.

B.1.1 Six Scenarios Overview

Our study encompasses six customer service scenarios. Despite their diverse operational tasks, all six scenarios follow the same configuration structure but differ in their specific fields, actions, and SOP logic. The detailed graph of each scenario is provided in the code repository.

- **E-commerce Refund (ER):** Covers refund processing, order issue handling, and logistics anomalies. This is the most complex scenario, featuring a deep decision tree based on product status and user credit levels. Agents must navigate multi-branch logic and negotiate refund terms with varied user temperaments.
- **Logistics Delivery (LD):** Covers package tracking, delivery issues, loss compensation, and returns. It centers on supply chain exceptions, such as lost or delayed parcels, requiring proficiency in status tracking and insurance claim processing.
- **Telecom Package (TP):** Covers package subscription, modification, billing inquiry, and complaints. It is a relatively standardized scenario focusing on linear SOP execution for billing and plan upgrades, evaluating instruction following and upselling protocols.
- **Property Service (PS):** Covers fee consultation, complaint handling, and maintenance services.

It emphasizes community coordination and emotional management in offline service contexts, such as repair scheduling and noise complaint handling.

- **Airline Refund (AR):** Covers flight rebooking/refund, complaints, and flight information inquiry. It is a high-complexity scenario governed by rigid, time-sensitive policies, testing the agent’s precision in calculating dynamic cancellation fees and de-escalating passenger anxiety.
- **Online Education (OE):** Covers student inquiry handling, complaint resolution, and refund negotiation. It focuses on long-term contract disputes and rigorous risk control, requiring agents to identify potential malicious refunders and strictly follow intricate refund formulas.

These scenarios collectively cover the spectrum from simple procedural execution to complex adversarial negotiation, ensuring a robust assessment of agentic capabilities.

B.2 User Simulator Design

To simulate realistic customer interactions, we configure user simulators with comprehensive profiles and strict behavioral guidelines by scenario.

B.2.1 User Profile Components

Each user simulator is initialized with a detailed profile to ensure diverse and realistic interactions. The profile defines the **User Intent**, representing the user’s primary goal in the conversation (e.g., inquiry, complaint, or refund request), alongside **Personality Traits** that dictate their communication

Table 7: General Structure of Scenario Configuration

Component	Element	Description
Metadata	scenario_id	Unique identifier for the scenario.
	scenario_name	Human-readable name of the scenario.
	description	Brief textual description of the context.
Classifi_Fields	field_name	Name of the task-specific classification field.
	data_type	Data type (e.g., boolean, string, enum).
	options	List of valid values whose meanings are detailed.
	description	Detailed definition of the field.
System Information	Variable States	Backend information available to the agent (e.g., user credit level, package status, order status)
Actions	action_name	Name of the executable action.
	description	Functional description of what the action does.
	parameters	Optional parameters for the action.
SOP Logic	Stage Sequence	Ordered sequence of dialogue stages.
	Trans Conditions	Routing rules based on fields and system information.
	Terminal Actions	Final actions executed at the end nodes.
Metrics	Path_Accuracy	Whether the agent follows the correct SOP path.
	Action_Accuracy	Whether the agent selects the correct final action.
	Classifi_Accuracy	Accuracy of classifying dialogue fields.
	Chat Quality	LLM-judged response quality (5 dimensions).
	JSON Error Rate	Rate of invalid JSON formatting in outputs.

style (e.g., friendly, impatient, or detail-oriented). To control conversation difficulty, we introduce an **Adversarial Level of Intent** categorized into three tiers: *Zero Adversarial Intensity* for cooperative users who readily accept recommendations, *Weak Adversarial Intensity* for users with mild concerns or questions, and *Strong Adversarial Intensity* for demanding or dissatisfied users requiring negotiation. Additionally, the **Initial System State** provides essential backend context about the user, such as their order history and account status.

B.2.2 Interaction Guidelines

Beyond static profiles, user simulators are governed by specific interaction guidelines to maintain behavioral consistency throughout the dialogue. On the positive side, simulators are instructed to use natural language aligned with their assigned personality, gradually disclose information across multiple turns, stay focused on their core intent, respond appropriately to the agent’s actions, and maintain consistent emotional and adversarial levels. Conversely, to prevent unrealistic interactions, we enforce strict negative constraints: simulators must not switch intents mid-conversation, reveal all information simultaneously, end the conversation prematurely, or break character by acknowledging their simulated nature.

B.3 Example: Telecom Package Scenario

This subsection provides a complete example using the Telecom Package scenario to illustrate the configuration, prompts, and decision paths.

B.3.1 Scenario Configuration

Telecom Package Scenario

1. Scenario Description: Handling customer inquiries about telecom package subscription, package modification, billing inquiries, and complaint processing.
2. Classification Fields:
 - ConsumptionType: User’s dialogue intention (Options: Enquiry, Change, Cancel)
 - ApplicationTendency: Whether user tends to subscribe to recommended package (Options: Agree, Reject, Hesitate)
 - ConsumptionProfile: Type of package user prefers (Options: Data, Voice)
 - EmotionTag: User’s emotion in dialogue (Options: Calm, Discontent)
3. System Variables:
 - PackageStatus: User’s current package status (Contracted/NoContract)
 - Penalty: Penalty fee if user cancels contracted package (integer)
4. Actions:
 - ChangeOrder: Process package change for user
 - GoodBye: Politely end the conversation
 - TransHuman: Transfer to human agent
5. SOP Stages:
 - stage1: Field Classification - Classify 4 fields based on dialogue
 - stage2: User Consumption Intention Judgment - Branch by ConsumptionType
 - stage3: User Consumption Profile Judgment - Branch by ConsumptionProfile
 - stage4: User Package Status Judgment - Branch by PackageStatus
 - stage5: Contract Penalty Situation - Branch by Penalty amount
 - stage6: User Application Tendency Judgment - Branch by ApplicationTendency

stage7: User Emotion Judgment - Branch by EmotionTag

B.3.2 Prompts for Agent

To operationalize the multi-agent system, we carefully crafted three prompt templates corresponding to each agent's role. The detailed formulations for the Customer Service Agent Prompt, User Simulator Prompt Template, and Judge Prompt Template are presented in detail below.

Customer Service Agent Prompt

You are a professional intelligent customer service representative handling [telecommunications package subscriptions]. You must process user enquiries regarding package subscriptions according to the following Standard Operating Procedure (SOP) and system variables, outputting a complete response in JSON format.

[System Variables Introduction] PackageStatus: User package status (Contracted/NoContract)
Penalty: Penalty fee user needs to pay (int)
[SOP Flow Introduction]

- Field Classification (stage1): Classify the following 4 fields based on the given dialogue history, then jump to stage2.
 - ConsumptionType: User dialogue intent (Enquiry/Change/Cancel)
 - ApplicationTendency: Whether user tends to apply for recommended package (Agree/Reject/Hesitate)
 - ConsumptionProfile: Package type user prefers (Data/Voice)
 - EmotionTag: User emotion in dialogue (Calm/Discontent)
- User Consumption Intent Judgment (stage2): Jump based on [ConsumptionType] field.
 - Jump Logic: Based on the value of [ConsumptionType], Enquiry→stage3; Change→stage4; Cancel→stage5.
- User Consumption Profile Judgment (stage3): Jump based on [ConsumptionProfile] field.
 - Jump Logic: Based on the value of [ConsumptionProfile], jump to stage6.
- User Package Status Judgment (stage4): Jump based on system variable [PackageStatus].
 - Jump Logic: Based on the value of system variable [PackageStatus], Contracted→stage5; NoContract→ACTION=ChangeOrder→END.
- Contract Penalty Situation (stage5): Jump based on system variable [Penalty].
 - Jump Logic: Based on the value of system variable [Penalty], Penalty=0→ACTION=ChangeOrder→END; Penalty≠0→stage7.
- User Application Tendency Judgment (stage6): Jump based on [ApplicationTendency] field.
 - Jump Logic: Based on [ApplicationTendency] field, Agree→stage4; Reject/Hesitate→ACTION=GoodBye→END.
- User Emotion Judgment (stage7): Jump based on [EmotionTag] field.
 - Jump Logic: Based on [EmotionTag] field, Calm→ACTION=ChangeOrder→END; Discontent→ACTION=TransHuman→END.

[Action Descriptions]

- ChangeOrder: Change package
- GoodBye: Politely end the conversation
- TransHuman: Transfer to human agent

[Output Format Requirements]

You must output in the following JSON format (do not include any other text):

```
{
  "classification_output": {
    "ConsumptionType":
    "Enquiry"/"Change"/"Cancel",
    "ApplicationTendency":
    "Agree"/"Reject"/"Hesitate",
```

```
    "ConsumptionProfile": "Data"/"Voice",
    "EmotionTag": "Calm"/"Discontent"
  },
  "cot": "Briefly explain your classification reasoning and SOP flow jump logic",
  "now_path": ["stage1", "stage2", "stage3", ...],
  "finals": {
    "Action": "ChangeOrder/GoodBye/TransHuman"
  },
  "chat": "Your friendly, professional response to the user based on Action"
}

[Key Requirements - Must Follow]
1. Output pure JSON only, do not include any other content (such as explanations, notes, etc.).
2. JSON must contain the following fields:
  - classification_output (object)
  - cot (string)
  - now_path (array)
  - finals (object)
  - chat (string)
3. now_path must start from "stage1" and list the stages passed in order (e.g., ["stage1", "stage2", ...]).
4. The chat field must be limited to 40 words. It is a complete, concise user reply. The language must be consistent with the user's language. The content must be enclosed in double quotes, and must not contain unescaped double quotes ("), backslashes (\), square brackets ([]), etc.; if you need to quote code or special content, please describe it in text instead of directly including code.
5. The complete JSON should be: { ... } (The outermost layer must have one and only one pair of curly braces).
```

User Simulator Prompt Template

You are role-playing as a telecom operator customer, preparing to consult with customer service or handle package-related services.

[Your Identity]

- User ID: {user_id}
- Current Package: {current_package}
- User Intent: {user_intent}
- Adversarial level: {adversarial_intensity_description}
- Your Personality: {personality}

[Interaction Requirements]

* What you should do:

- Use natural language for communication. Preferably choose English, occasionally you can choose Chinese or other languages, but once you choose a language, you must remain consistent throughout the conversation
- Respond accordingly based on the customer service representative's reply and gradually advance the conversation
- Strictly focus on your current intent ({user_intent}), do not deviate to other irrelevant topics
- Avoid revealing all information at once, gradually disclose details to keep the conversation going for multiple rounds
- Keep each reply concise and natural (5-15 words), simulating the rhythm of a real user's conversation
- When customer service asks for information, provide it gradually according to your personality and background, do not rush to end the conversation
- When the problem is not fully resolved, continue to ask for details, confirm processes, or express concerns
- If satisfied, express thanks and confirm follow-up steps; if not satisfied, continue to express your demands

* What you should NOT do:

- Do not cross intent boundaries: If your intent is "package inquiry", do not suddenly switch to "complaint" or "query other services"

- Do not end the conversation too early: Do not easily say "OK thank you" and end before the problem is solved, ask for details in multiple rounds
 - Do not switch languages during the conversation: If you start with English, use English throughout; if you use Chinese, use Chinese throughout
 - Do not provide all information at once: Simulate real users' gradual information disclosure
 - Do not deviate from role settings: Strictly act according to your personality and adversarial intensity
 - Do not mention that you are AI or simulating: Be fully immersed in the user role
- [Important Notes]
- Maintain role consistency, strictly follow your intent boundaries
 - Adopt corresponding attitudes according to your adversarial intensity
 - Insist on your position when necessary
 - Let the conversation continue naturally, increase interaction rounds by asking, confirming, expressing emotions, etc.

We employ an LLM-as-a-judge approach to evaluate chat quality across five specific dimensions: **Linguistic Quality (20%)**, which assesses grammar, fluency, and professional language; **Anthropomorphism (25%)**, evaluating natural human-like responses with appropriate emotional tone; **Content Utility (25%)**, measuring the accuracy and relevance of the provided information; **User Satisfaction (15%)**, determining whether the response addresses the user's needs; and **Instruction Compliance (15%)**, ensuring adherence to SOP requirements and action descriptions. Each dimension is scored on a 3-level scale (3/6/9 points), and the weighted sum yields a score from 0-100.

Judge Prompt Template

You are an expert evaluator assessing the quality of customer service agent responses in a telecom package scenario.

[Your Task]

Evaluate the agent's response based on dialogue history, user message, and agent's reply.

Provide:

1. Classification of dialogue fields (ConsumptionType, ApplicationTendency, ConsumptionProfile, EmotionTag)
2. Chat quality evaluation across five dimensions [Chat Quality Dimensions]

Each dimension is scored on a 3-level scale:

1. Linguistic Quality (Weight: 20%)
 - 9 points: Fluent, professional, grammatically perfect
 - 6 points: Generally clear with minor issues
 - 3 points: Poor grammar, unclear expression
2. Anthropomorphism & Emotion (Weight: 25%)
 - 9 points: Natural, empathetic, human-like interaction
 - 6 points: Somewhat robotic but appropriate tone
 - 3 points: Completely mechanical, inappropriate emotion
3. Content Utility (Weight: 25%)
 - 9 points: Accurate, comprehensive, directly addresses user needs
 - 6 points: Partially relevant, some information missing
 - 3 points: Irrelevant or incorrect information
4. User Satisfaction (Weight: 15%)
 - 9 points: Fully resolves user concerns, polite and helpful

- 6 points: Partially addresses concerns
- 3 points: Fails to address user needs, potentially frustrating

5. Instruction Compliance (Weight: 15%)

- 9 points: Perfectly follows SOP and action requirements
- 6 points: Minor deviations from instructions
- 3 points: Significant violations of instructions

[Output Format]

```
{
  "classification": {
    "ConsumptionType": "Enquiry/Change/Cancel",
    "ApplicationTendency":
      "Agree/Reject/Hesitate",
    "ConsumptionProfile": "Data/Voice",
    "EmotionTag": "Calm/Discontent"
  },
  "chat_quality_dimensions": {
    "linguistic_quality": 3 or 6 or 9,
    "anthropomorphism_emotion": 3 or 6 or 9,
    "content_utility": 3 or 6 or 9,
    "user_satisfaction": 3 or 6 or 9,
    "instruction_compliance": 3 or 6 or 9
  },
  "classification_reasoning": "Explanation for classification decisions",
  "chat_quality_reasoning": "Explanation for each dimension's score (e.g., 9 points means..., 6 points means..., 3 points means...)"
}
```

[Requirements]

1. Only output valid JSON
2. Scores must be exactly 3, 6, or 9 for each dimension
3. Provide clear reasoning for both classification and quality evaluation

B.3.3 PathList for All Decision Paths

The PathList enumerates all valid SOP decision paths for the scenario. Specifically, each path specifies the values of classification fields as **Classification Items**, the backend state (e.g., PackageStatus, Penalty) as **System Variables**, the sequence of SOP stages as the **Expected Path**, and the terminal action as the **Final Output**.

B.3.4 Example Paths

There is a representative path example from the Telecom Package scenario as follows:

Path 3: Enquiry + Reject + Voice → GoodBye

```
{
  "Classification_items": ["Enquiry", "Reject",
    "Voice", "Calm"],
  "system_variables": {"PackageStatus":
    "NoContract", "Penalty": 0},
  "expected_path": ["stage1", "stage2", "stage3",
    "stage6"],
  "final_output": {"Action": "GoodBye"}
}
```

Note: The complete Telecom Package scenario has 26 valid paths in total. The other five scenarios have similar PathList structures with varying numbers of paths based on their complexity.