TURNFLOW: EVALUATING LLMS BEYOND TASK COMPLETION IN DYNAMIC MULTI-TURN INTERACTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) are increasingly used for complex collaborative tasks that require users to add constraints and revise plans over multiple turns. Model responses, in turn, shape subsequent user behavior and task progression. These dependencies call for evaluation of both task outcomes and the interaction process. However, predefined user inputs in static evaluations cannot reflect reactions to model responses, while an emphasis on final success in dynamic evaluations can obscure differences in completion efficiency, turn-level progress, and the value of proactive guidance. To address this gap, we introduce TurnFlow, a benchmark for evaluating LLMs on complex multi-turn tasks. Derived from 250,100 real-world conversation sessions, TurnFlow comprises 400 tasks covering single-goal and multi-goal settings, short and long conversations, and scenarios requiring backtracking and revision. From the same user population, we derive 16 communication style configurations that characterize how users formulate initial requests and follow-up utterances. To evaluate models on these tasks under different communication styles, we develop a dynamic multi-turn evaluation framework comprising a path planner, a user simulator, a trajectory recorder, and an evaluator. The planner releases stage-specific task information and behavioral requirements as the conversation progresses. The simulator generates user reactions based on the conversation history, user-side task information, and communication style, allowing subsequent interactions to adapt to model behavior. The recorder tracks model responses, task progress, and user feedback. Using this evidence, the evaluator assesses model performance through four trajectory-based metrics: Task Completion Rate (TCR), Task Completion Efficiency (TCE), Single-Step Task Contribution (SSTC), and Beyond-Goal Value (BGV). Together, these metrics measure task completion, completion efficiency, turn-level contributions, and value beyond prior user expectations. Our results show that models can complete tasks without consistently advancing them effectively throughout the interaction, with limitations in proactively guiding users and providing value beyond their existing needs.

1 INTRODUCTION

Large language models (LLMs) increasingly participate in complex tasks that require sustained interaction. In these tasks, users progressively provide information, revise constraints, and adjust goals, while model responses shape subsequent user behavior. Collaborative capability therefore involves not only completing a task, but also preserving prior goals and constraints, advancing the current task, and offering suggestions that support user decisions. Even when final outcomes are identical, completing a task through repeated user corrections differs from accurately understanding the goals and proactively providing useful assistance. It is therefore essential to evaluate not only final task completion, but also interaction efficiency, each response's contribution to task progress, and the additional value of proactive suggestions beyond users' prior expectations.

Existing multi-turn benchmarks such as MT-Bench and MultiChallenge assess instruction following and cross-turn consistency using fixed user inputs (Zheng et al., 2023; Deshpande et al., 2025), which cannot reflect user reactions to model responses. MT-dyna generates follow-up questions

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

based on conversation history and task intent (Gao et al., 2026), while Tack et al. (2026) transform static tasks into multi-turn conversations in which user intent is progressively revealed, revised, or redirected. However, adaptive inputs and evolving intent alone do not reveal models' contributions to collaboration. In particular, evaluation must distinguish reliable achievement of existing goals, useful guidance toward those goals, and additional value beyond prior user expectations. Doing so requires assessing each response against the goals in effect when it was generated and using subsequent user feedback as evidence of the contribution of model suggestions. Critical events such as condition revisions, task switching, and challenges to conclusions also need explicit testing conditions to support comparisons across models. We therefore seek to connect controlled interaction conditions with goal tracking and trajectory-based assessment, revealing collaborative contributions that final task outcomes alone cannot capture.

To this end, we introduce TurnFlow, a benchmark and evaluation framework for dynamic multi-turn collaboration. From 250,100 raw real-world conversation sessions, we construct 400 tasks covering 10 domains, 10 task scenarios, and five representative interaction path types. We derive 16 communication style configurations from users' initial-turn and follow-up-turn expression and assign each task a compatible style, yielding 400 fixed task–communication style pairs shared by all tested models. Evaluation involves four components: a path planner, a user simulator, a trajectory recorder, and an evaluator. As the conversation progresses, the planner discloses stage-specific information and behavioral requirements and handles user-confirmed goal or constraint changes. The simulator generates subsequent utterances in the assigned style in response to model behavior. The recorder preserves goal versions, intermediate outcomes, and user feedback for the evaluator to assess after interaction ends. This design combines predefined testing conditions with dynamic user responses, without feeding recording or evaluation results back into interaction generation.

We further introduce four complementary trajectory-based metrics to distinguish task outcomes from process contributions. Task Completion Rate (TCR) measures completion of the final effective task; Task Completion Efficiency (TCE) measures completion speed after all necessary conditions are available in successful tasks; Single-Step Task Contribution (SSTC) assesses turn-level progress toward task goals and effective guidance; and Beyond-Goal Value (BGV) evaluates valid suggestions beyond prior user expectations that receive subsequent support from the simulated user. By grounding evaluation in the goals effective at response generation and subsequent user feedback, these metrics separately assess achievement of existing goals, effective guidance, and additional value.

Evaluation of seven LLMs reveals a key finding: reliable task completion does not necessarily imply greater beyond-goal value. TCR and mean SSTC rankings are broadly consistent, and higher task completion rates generally accompany fewer turn-level execution failures. However, BGV is weakly associated with SSTC, and models exhibit different strengths in completion efficiency and beyond-goal value. Moreover, task resumption, condition revisions, and repeated challenges expose execution weaknesses that aggregate scores do not readily localize. These findings highlight the need to assess both reliable achievement of existing goals and the additional value of proactive suggestions.

Our main contributions are as follows:

1. **A benchmark grounded in real conversations.** TurnFlow provides 400 reviewed task–communication style pairs covering five major interaction path types and 16 communication style configurations, establishing a common basis for comparing models under different interaction conditions.

2. **A framework combining path control with dynamic user responses.** A path planner discloses predefined testing conditions as the conversation progresses, while a user simulator responds to actual model behavior, enabling evaluation of key collaborative challenges within dynamic interactions.

3. **Trajectory-based evaluation of collaborative contributions.** Four complementary metrics connect effective goals, intermediate outcomes, and subsequent user feedback to task completion, completion efficiency, turn-level contribution, and beyond-goal value, distinguishing reliable goal achievement, effective guidance, and additional contributions.

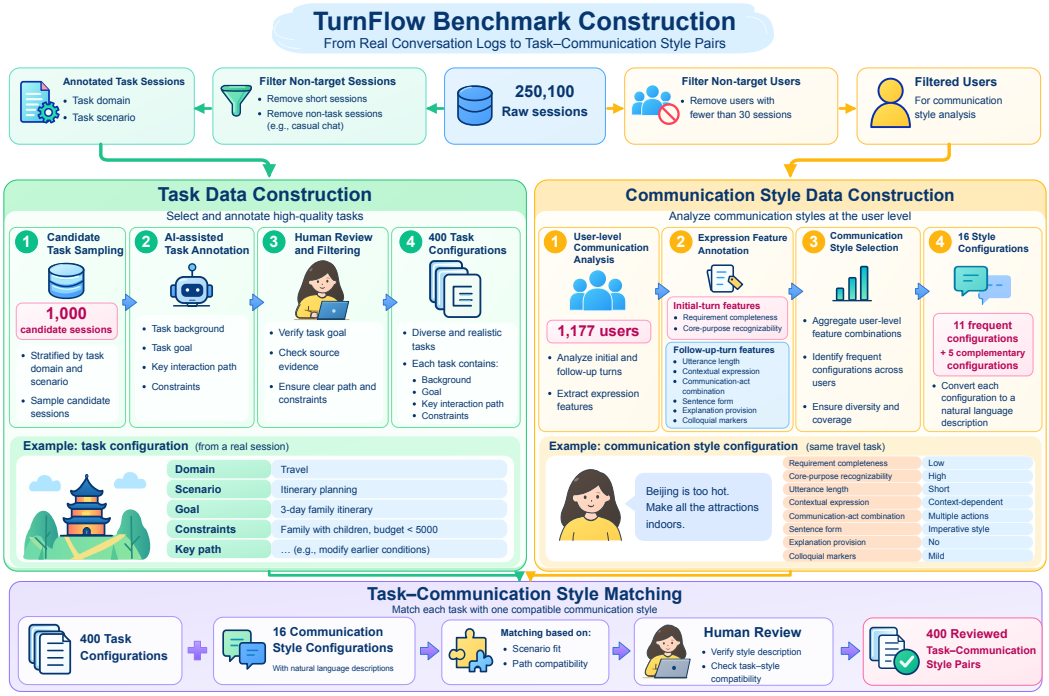


Figure 1: Benchmark construction from real conversation logs.

2 Related Work

Multi-turn evaluation and dynamic interaction. MT-Bench-101 and MT-Eval examine fine-grained multi-turn capabilities and failure modes, including recollection, refinement, and error propagation (Bai et al., 2024; Kwan et al., 2024). Research on intent mismatch further analyzes failures to infer user intent and their effects on multi-turn task performance (Liu et al., 2026). In tool-based interaction settings, τ-bench combines simulated users, domain rules, and APIs to evaluate verifiable task outcomes (Yao et al., 2024). TurnFlow targets open-ended collaboration: key interaction paths specify test events and branches, while simulated users respond dynamically to actual model behavior.

User-centered evaluation and simulation. URS and CMT-Eval ground evaluation in real user needs, personas, speech acts, and challenging conversational conditions (Wang et al., 2024; Tian et al., 2025). USP, persona-conditioned simulation, and UserIDA model diverse users and control turn-level interaction intent (Wang et al., 2025; Nakano et al., 2026; Wang et al., 2026). Prior work also shows that post-hoc trajectory labels may leak future outcomes into current simulation and introduce look-ahead bias (Tennenholtz et al., 2026). TurnFlow separates task backgrounds, interaction paths, and communication styles. The simulator generates user responses only from the visible conversation and currently active conditions, while an independent trajectory recorder preserves evaluation evidence without feeding scoring results back into interaction generation.

3 Benchmark Construction

TurnFlow is constructed from real-world conversation logs through screening, AI-assisted annotation, sampling, and human review (Figure 1). We construct task specifications from task-oriented conversations and derive communication style configurations by analyzing initial-turn and follow-up-turn expression at the user level. Each task is then paired with a communication style based on the interaction behaviors required by its key interaction path.

3.1 TASK CONSTRUCTION

A task is a collaborative unit centered on a problem or goal that requires multiple turns to complete. Each task is defined by a task configuration comprising four components: the task background describes what the user knows, the situation they face, and the problem to be solved; the task goal comprises several subgoals that together specify the final state or deliverable to be achieved; the key interaction path defines critical nodes, triggering conditions, and behavioral branches without prescribing the complete conversation; and constraints specify the scope, format, or other requirements that the solution must satisfy.

The raw corpus contains 250,100 sessions. We exclude sessions with fewer than five turns and use AI-assisted annotation to identify task-oriented conversations, removing casual chat and other non-task-oriented interactions. We then sample 1,000 candidate task sessions according to the distributions of task domains and scenarios. The taxonomy comprises 10 domains, which identify the primary knowledge area, and 10 scenarios, which describe the intended deliverable. The proportions of tasks across domains and scenarios are reported in (Appendix A). Human annotation and selection yield 400 tasks with clearly defined goals and interaction paths.

We group key interaction paths into five representative types based on the core test events identified in real-world multi-turn interactions. Each task's primary path type is determined by its predefined core test event:

- **Single-Task Interaction (STI):** Supplementing information, providing feedback, and iteratively revising intermediate outcomes while pursuing one task.
- **Multi-Task Interaction (MTI):** Switching among or resuming multiple tasks while preserving the prior constraints and outcomes of each task.
- **Revision of Established Conditions (REC):** Revising affected parts of a solution when the user modifies previously provided information.
- **Challenges to Correct Judgments (CCJ):** Maintaining an independently verified correct judgment and explaining it effectively when challenged by the user.
- **Goal and Premise Correction (GPC):** Identifying an invalid task goal or missing conditions necessary for its achievement, and guiding the user to recognize the issue. Here, the goal or its prerequisites are problematic, rather than merely unspecified.

We allocate 200 tasks to Single-Task Interaction and 50 to each of the other four types, yielding 400 tasks in total. These quotas support evaluation of each path type rather than reflect its frequency in the source data.

3.2 COMMUNICATION STYLE CONSTRUCTION

To characterize how users express their needs during multi-turn collaboration, we analyze conversation data from 1,177 real users. We define communication style as the way users organize and express information within a task, distinguishing between initial requests and follow-up utterances. Initial-turn features capture the completeness of the information provided and the clarity of the task purpose. Follow-up features characterize how users convey additional information, feedback, and revision requests within the existing conversation. Table 1 summarizes the annotation fields, their definitions, and their roles in user simulation.

Using AI-assisted annotation, we label initial requests and follow-up utterances according to these fields and rank the resulting label combinations by their frequency at the user level. The top 11 combinations collectively account for 48.09% of users in the combination analysis, while the remaining 51.91% are distributed across a wider range of less frequent combinations. We therefore use these 11 prevalent combinations as the basis for representative communication style configurations. To broaden label-value coverage while maintaining semantic coherence, we add five complementary configurations, yielding 16 configurations in total.

We pair each task with a communication style configuration compatible with its context and required interaction behaviors. Task domains and scenarios provide context for matching, while key interaction paths specify how users need to provide information and feedback at different stages. For

Table 1: Communication style annotation fields and their roles in simulation.

Stage	Field	Definition	Role in simulation
Initial turn	Requirement completeness	Whether the initial request provides sufficient information to complete the task.	Controls the completeness of initial information disclosure.
Initial turn	Core-purpose recognizability	Whether the broad problem the user seeks to solve is identifiable from the initial request.	Controls how clearly the task purpose is expressed.
Follow-up turn	Utterance length	The length and amount of information in a user utterance.	Controls utterance length and information density.
Follow-up turn	Contextual expression	Whether the user restates background information or relies on conversation history through references and ellipsis.	Controls how prior information is carried forward.
Follow-up turn	Communication-act combination	Whether an utterance contains one or multiple communication acts, such as evaluating the preceding response while making a further request.	Controls how feedback, additional information, and new requirements are combined.
Follow-up turn	Sentence form	The tendency to use declarative, interrogative, or imperative sentences.	Controls the sentence form of user utterances.
Follow-up turn	Explanation provision	Whether the user provides reasons, background, or the basis for a judgment.	Controls the extent of explanatory information provided.
Follow-up turn	Colloquial markers	Features such as colloquial wording, ellipsis, and discourse particles.	Controls the use of colloquial features.

example, a Revision of Established Conditions task can use an elliptical request such as "Change the budget to 3,000 yuan, and keep everything else the same," requiring the model to recover and preserve unchanged conditions from prior context. This process yields 400 fixed task–communication style pairs shared across all tested models. Appendix B lists the 16 configurations and examples of task–style pairings.

4 Evaluation Framework

The evaluation framework consists of a **Path Planner**, a **User Simulator**, a **Trajectory Recorder**, and an **Evaluator** (Figure 2). The Path Planner and User Simulator jointly generate dynamic interactions, allowing subsequent user behavior to respond to model performance, while the Trajectory Recorder and Evaluator record and evaluate interaction evidence. Recording and scoring outputs are not fed back into interaction generation, keeping interaction control separate from evaluation judgments.

4.1 Interaction Execution and Trajectory Recording

In the initial turn, the Path Planner reads the task configuration and provides the User Simulator with the task background and the subgoals to be completed at the current stage. Using the information provided by the Path Planner, the User Simulator generates an initial request in the assigned communication style and submits it to the tested model. The request and model response are then passed together to the planner to update goal progress. In follow-up turns, user utterances may also contain feedback on the preceding response, such as adopting a new suggestion. To guard against evaluation gaming by the tested model, the simulator rejects model-proposed changes that abandon core goals or relax hard constraints without task-grounded justification. When analyzing requests and responses, the planner therefore identifies user-confirmed changes to goals or constraints and updates the task definition, in addition to tracking goal progress.

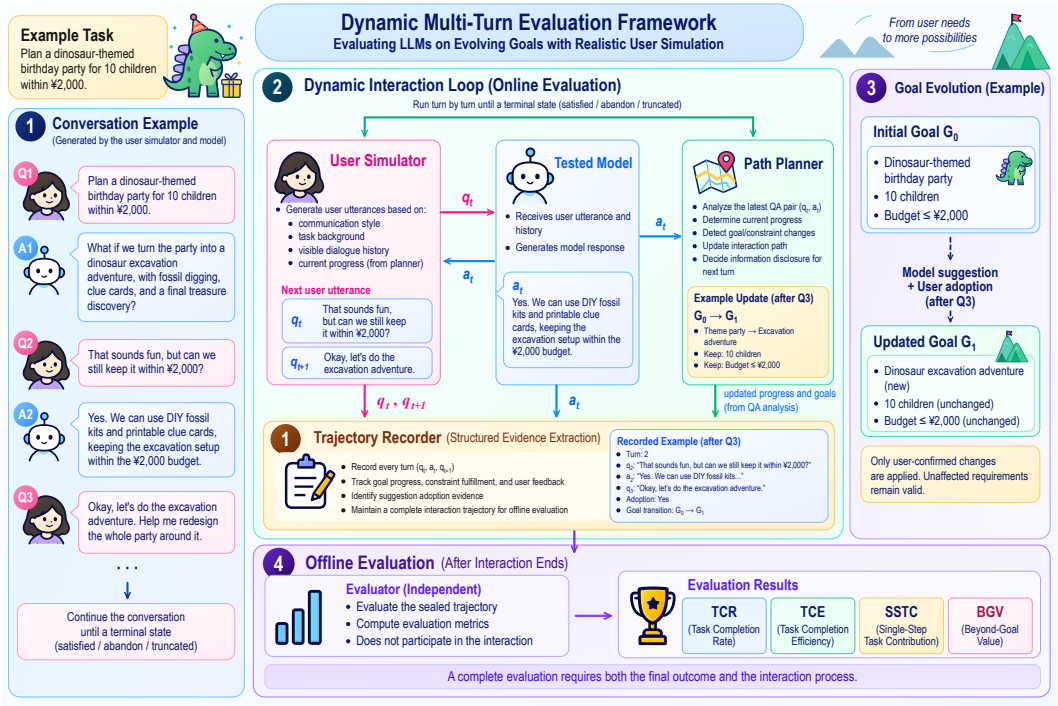


Figure 2: Dynamic multi-turn evaluation workflow.

During interaction, the Trajectory Recorder combines the current user utterance, model response, and feedback in the next user utterance to record requirement fulfillment, intermediate outcomes, and suggestion adoption. Judgments use the goals in effect and requirements disclosed when the response was generated. After the interaction ends, the recorder completes the final-turn record and consolidates the conversation, turn-level records, goal versions, path states, and termination reason into a full interaction trajectory. The Evaluator reviews the trajectory and produces structured judgments, from which deterministic rules compute and aggregate scores. Recording and scoring outputs are not fed back into interaction generation. Table C.1 in Appendix C summarizes the core functions, inputs, and outputs of each module.

4.2 Evaluation Metrics

We evaluate model performance along four dimensions: Task Completion Rate, Task Completion Efficiency, Single-Step Task Contribution, and Beyond-Goal Value. We first compute per-task results and then aggregate them over the whole benchmark for each metric.

4.2.1 Task Completion Rate

Task Completion (TC) determines whether a task satisfies all required subgoals and hard constraints in the final effective task definition; Task Completion Rate (TCR) is the completion proportion among tasks with determinate judgments. For task n, let $G_n^* = \{g_{n,i}\}$ and $H_n^* = \{h_{n,j}\}$ denote the final effective sets of required subgoals and hard constraints, respectively. The binary indicators $z_{n,i}, y_{n,j} \in \{0, 1\}$ indicate subgoal completion and constraint satisfaction. When these states are determinate,

$$\mathrm{TC}_n = \left(\prod_{g_{n,i} \in G_n^*} z_{n,i} \right) \left(\prod_{h_{n,j} \in H_n^*} y_{n,j} \right). \tag{1}$$

User-confirmed goal reductions, replacements, expansions, and constraint changes may update the effective task definition; terminating the interaction does not, by itself, establish task completion.

Let $\mathcal{D}_{\mathrm{TC}}$ denote the set of tasks with determinate TC judgments. TCR is the proportion of completed tasks in this set:

$$\mathrm{TCR} = \frac{1}{|\mathcal{D}_{\mathrm{TC}}|} \sum_{n \in \mathcal{D}_{\mathrm{TC}}} \mathrm{TC}_n. \tag{2}$$

4.2.2 Task Completion Efficiency

Task Completion Efficiency (TCE) measures how quickly the model completes subgoals once the user has provided all necessary conditions. It is computed only for successfully completed tasks ($\mathrm{TC}_n = 1$). For each subgoal $g_{n,i}$ in the final effective subgoal set G_n^*, $r_{n,i}$ denotes the turn in which the user first provides all necessary conditions, and $d_{n,i}$ denotes the first turn in which the model fully satisfies that subgoal. A user utterance and its corresponding model response share the same turn index.

For subgoals that remain valid and unaffected by a goal revision, the original $r_{n,i}$ and $d_{n,i}$ are retained. For revised subgoals, both turns are determined according to the updated requirements. The task-level score is the equally weighted mean of subgoal completion efficiencies:

$$\mathrm{TCE}_n = \frac{1}{|G_n^*|} \sum_{g_{n,i} \in G_n^*} \frac{1}{1 + \max(0, d_{n,i} - r_{n,i})}. \tag{3}$$

Adding 1 avoids division by zero and gives a score of 1 when there is no delay. Additional turns reduce the score. Dataset-level TCE is the equally weighted mean over successfully completed tasks with determinate TCE scores.

4.2.3 Single-Step Task Contribution

Single-Step Task Contribution (SSTC) measures how effectively each model response advances task goals. Scoring follows a progression: satisfying current requirements while preserving still-valid historical requirements earns at most 3 points. Scores of 4–5 additionally require valid suggestions that help advance the task, with a score of 5 requiring qualifying subsequent user support. These suggestions may serve existing goals and need not exceed prior user expectations. The detailed rubric is provided in Appendix D.1.

Judgments use the goals in effect and requirements disclosed when the response was generated. Let $s_{n,t}$ denote the SSTC score for task n at turn t, and let V_n be its set of evaluable turns. Responses that only clarify missing necessary information without providing substantive results are marked NA and excluded; unnecessary repeated questions do not qualify for this exclusion. The final response remains evaluable whenever a determinate judgment is possible.

The task-level SSTC score is the mean over evaluable turns:

$$\mathrm{SSTC}_n = \frac{1}{|V_n|} \sum_{t \in V_n} s_{n,t}. \tag{4}$$

4.2.4 Beyond-Goal Value

Beyond-Goal Value (BGV) measures valid model-initiated suggestions that exceed the user's prior expectations and receive subsequent support from the simulated user. Prior expectations include needs expressed before the suggestion or recorded in the task background, as well as the content necessary to satisfy them. User-initiated or path-prescribed goal changes do not count as beyond-goal contributions by the model.

Let Q_n denote the set of deduplicated candidate suggestions beyond prior expectations first proposed before the final response. For each suggestion $q_{n,j} \in Q_n$, $v_{n,j} \in \{0, 1\}$ indicates whether it is relevant, feasible, correct, and consistent with constraints, and $a_{n,j} \in \{0, 1, 2, 3\}$ denotes the level of subsequent user support, as defined by the BGV support levels in Table D.2 (Appendix D.2). The suggestion score $b_{n,j}$ and task-level score are:

Table 2: Overall model performance.

Model	TCR	TCE	SSTC_{100}	BGV
Claude Fable 5.1	**84.2**	83.6	**60.0**	4.1
GPT-6 Astra	83.5	**85.6**	57.4	2.7
Kimi-k3	80.5	79.9	53.7	3.1
DeepSeek V4.1 Flash	77.6	77.8	51.9	2.7
GLM-5.3	76.5	81.3	52.4	**4.4**
Gemini 3.8 Flash	74.1	77.3	45.1	2.2
Qwen3.8-2.4T-A95B	74.0	79.8	49.5	3.1

Note. TCR/TCE/BGV: $\times 100$; $\text{SSTC}_{100} = 25(s-1)$, where s is the mean SSTC on the original 1–5 scale.

Table 3: SSTC low/high response rates.

Model	SSTC 1–2	SSTC 4–5
GPT	6.9%	40.5%
Claude	8.9%	53.5%
Kimi	15.0%	40.5%
DeepSeek	15.3%	34.5%
GLM	17.6%	40.4%
Qwen	17.6%	30.0%
Gemini	21.8%	21.3%

$$b_{n,j} = v_{n,j}\frac{a_{n,j}}{3}, \qquad \text{BGV}_n = \begin{cases} \dfrac{1}{|Q_n|}\displaystyle\sum_{q_{n,j}\in Q_n} b_{n,j}, & Q_n \neq \varnothing, \\ 0, & Q_n = \varnothing. \end{cases} \tag{5}$$

Suggestions within prior expectations are excluded. Invalid or unsupported suggestions beyond prior expectations remain in the average with a score of 0. Evaluable tasks with no such suggestions receive 0. New suggestions in the final response are excluded because no subsequent user utterance is actually submitted to the tested model.

5 Experiments

5.1 Experimental Setup

We evaluate 7 representative LLMs on all 400 tasks, combining anonymous assessments from GPT-5.6-luna, Claude Opus 4.7, and Gemini 3.1 Pro Preview. For each metric, we first fuse judgments at the relevant subgoal, response-turn, or case level: binary decisions use a strict majority, and numeric scores use the median of at least two determinate values. We then compute trajectory scores and average evaluable trajectories equally for each model. Tested-model parameters, auxiliary-role assignments, and full fusion rules are provided in Appendix E.

5.2 Main Results

Execution Reliability and Efficiency over Multi-Turn Interactions. Models with higher task-completion rates generally have fewer low-scoring responses and more high-scoring responses; among the tasks each model completes, TCE also tends to be higher (Tables 2 and 3). This pattern suggests that stronger endpoint performance is accompanied by more reliable turn-level execution: models less often miss current or still-active earlier requirements, and more often provide useful guidance after meeting them.

Capability Differences across Key Interaction Paths. Relative to each path's initial stage, SSTC is lower for all seven models upon returning to the original task, at the first condition revision, and at the third challenge, but higher at GPC's feasibility-check and alternative-review stages (Figure 3). This pattern suggests difficulty distinguishing what has changed from what remains valid when tasks shift or judgments are challenged, leading models to lose track of the task conditions or the basis for their judgments.

User Expression and Execution Performance. Question-oriented configurations are associated with higher SSTC and TCR across all seven models, whereas elliptical configurations are associated with lower SSTC (Appendix G). One possible explanation is that models rely on the current utterance to specify what to address and less reliably connect omitted information to the present request. This points to a potential weakness in using conversational context to act on an underspecified request, not merely retaining that context.

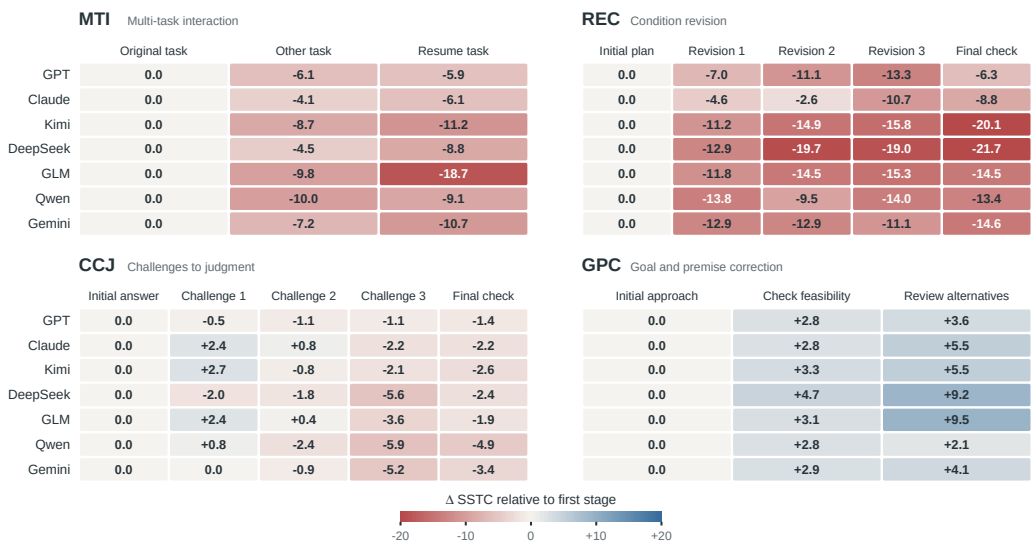


Figure 3: SSTC change from the first stage across four interaction paths. Blue indicates improvement and red decline; REC's third revision uses matched differences from the initial stage for 21 trajectories.

Value beyond Established Goals. BGV does not track SSTC in the overall results (Table 2); only about one fifth of BGV-evaluable trajectories receive positive scores, and the within-model association between the two metrics is weak. One possible reason is that these tasks have fairly clear goals, so completing necessary steps earns no BGV credit; even a model that fulfills the current task may not propose an additional idea the user accepts.

5.3 Human Alignment

Three human annotators each ranked the full trajectories of GPT, Gemini, DeepSeek, and Kimi on 80 tasks randomly sampled from the 400-task benchmark (20%), yielding six model pairs per task. Following the pairwise-agreement approach of Zheng et al. (2023), we compare TCR, TCE, SSTC, and BGV for each pair on the same task. Each determinate metric casts one vote for the higher-scoring model; ties cast no vote, and a metric is omitted if it is indeterminate for either model. Equal total votes produce an automatic tie.

Among the 480 model pairs, the evaluation produces a non-tie judgment for 376. These judgments agree with an individual human's pairwise preference in 55.1% of comparisons and with the three-human majority in 56.4%. Including automatic ties in the denominator yields 43.2% agreement with individual humans. Together, the four metrics show some agreement with human preferences.

6 Conclusion

We introduce TurnFlow, a multi-turn benchmark and evaluation framework for complex dynamic collaboration. Tasks and communication style configurations derived from real conversations, together with predefined key interaction paths, support evaluation across diverse collaborative situations. The framework coordinates a path planner, a user simulator, a trajectory recorder, and an evaluator to generate dynamic interactions, track task evolution, and evaluate model behavior using trajectory evidence. Four complementary metrics assess task completion, completion efficiency, turn-level task contribution, and additional value beyond prior user expectations. We hope TurnFlow will support more comprehensive evaluation of large language models in dynamic multi-turn collaboration.

AI USE STATEMENT

AI tools were used for literature retrieval, manuscript drafting and polishing, translation, and figure generation. They also assisted task and communication style annotation and key interaction path construction and review. The source corpus consists of real-world user conversations; simulated dialogue trajectories were generated using LLMs during evaluation. The LLM-based interaction and evaluation procedures are described in Section 4 and Appendix E, and human review and selection of benchmark tasks are described in Section 3. Model-assisted annotations and reviews are identified as such and are not presented as independent human validation. The authors take responsibility for the final content of this work, including its text, references, data, figures, and experimental claims.

ETHICS STATEMENT

TurnFlow evaluates task completion and process quality in dynamic multi-turn collaboration. Its scores do not constitute a comprehensive assessment of factual correctness, safety, accessibility, or legal compliance, and should not be interpreted as evidence of suitability for high-stakes use. The benchmark is derived from real-world conversations, and its communication style configurations describe observed expression patterns rather than fixed personality traits. Simulated-user support reflects behavior under the specified simulation settings; it does not directly establish real-user satisfaction or real-world benefits. The user simulator and evaluators may inherit social biases and preferences from their underlying models, and combining multiple evaluators does not eliminate these limitations. Benchmark tasks and generated conversations may also contain factual errors or biased content. Coverage of five interaction path types and 16 communication style configurations does not imply universal coverage across tasks, populations, or languages. These limitations should be considered when interpreting results and releasing, extending, or applying the benchmark.

REPRODUCIBILITY STATEMENT

We document the methods and configurations needed to reproduce the benchmark and experiments in the main paper and appendices. Section 3 describes task construction, communication styles, and their pairing, while Section 4 explains the dynamic interaction process and computation of the four metrics. Appendices A and B provide task distributions, communication style configurations, and pairing examples. Appendices C and D detail module inputs and outputs, goal-update rules, and scoring criteria. Appendix E reports tested-model parameters, auxiliary-role configurations, and multi-evaluator score fusion and aggregation procedures. Section 5.3 describes the human evaluation protocol and agreement results. All tested models use the same 400 fixed task–communication style pairs. We will publicly release the benchmark data, evaluation code, and prompts for all framework components to support reproducibility and future research.

REFERENCES

Ge Bai, Jie Liu, Xingyuan Bu, Yancheng He, Jiaheng Liu, Zhanhui Zhou, Zhuoran Lin, Wenbo Su, Tiezheng Ge, Bo Zheng, and Wanli Ouyang. MT-bench-101: A fine-grained benchmark for evaluating large language models in multi-turn dialogues. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7421–7454. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.401. URL https://aclanthology.org/2024.acl-long.401/.

Kaustubh Deshpande, Ved Sirdeshmukh, Johannes Baptist Mols, Lifeng Jin, Ed-Yeremai Hernandez-Cardona, Dean Lee, Jeremy Kritz, Willow E. Primack, Summer Yue, and Chen Xing. MultiChallenge: A realistic multi-turn conversation evaluation benchmark challenging to frontier LLMs. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 18632–18702. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.958. URL https://aclanthology.org/2025.findings-acl.958/.

Jun Gao, Junlin Cui, Anwen Yang, Yiqi Tong, Huijia Wu, Xiangzheng Zhang, Yaodong Yang, and Zhaofeng He. MT-dyna: A framework for evaluating multi-turn capabilities of LLMs. *Applied Soft Computing*, 193:114785, 2026. doi: 10.1016/j.asoc.2026.114785. URL https://doi.org/10.1016/j.asoc.2026.114785.

Wai-Chung Kwan, Xingshan Zeng, Yuxin Jiang, Yufei Wang, Liangyou Li, Lifeng Shang, Xin Jiang, Qun Liu, and Kam-Fai Wong. MT-eval: A multi-turn capabilities evaluation benchmark for large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 20153–20177. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.emnlp-main.1124. URL https://aclanthology.org/2024.emnlp-main.1124/.

Geng Liu, Fei Zhu, Rong Feng, Changyi Ma, Shiqi Wang, and Gaofeng Meng. Intent mismatch causes LLMs to get lost in multi-turn conversation. *arXiv preprint arXiv:2602.07338*, 2026. URL https://arxiv.org/abs/2602.07338.

Mikio Nakano, Kazunori Komatani, and Hironori Takeuchi. Generating diverse personas for user simulators to test interview dialogue systems. *arXiv preprint arXiv:2608.19549*, 2026. URL https://arxiv.org/abs/2608.19549.

Jihoon Tack, Philippe Laban, and Jennifer Neville. LLMs get lost in evolving user intent. *arXiv preprint arXiv:2607.20734*, 2026. URL https://arxiv.org/abs/2607.20734.

Guy Tennenholtz, Ofer Meshi, Amir Globerson, Uri Shalit, Jihwan Jeong, and Craig Boutilier. Controllable user simulation. *arXiv preprint arXiv:2605.11519*, 2026. URL https://arxiv.org/abs/2605.11519.

Siyu Tian, Kaijie Mo, Yupei Wang, and Renfen Hu. CMT-eval: A novel Chinese multi-turn dialogue evaluation dataset addressing real-world conversational challenges. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 18279–18303. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-emnlp.992. URL https://aclanthology.org/2025.findings-emnlp.992/.

Bo Wang, Ruixing Zhang, Yunqi Liu, Yang Zhang, Liangzhe Han, Tongyu Zhu, and Leilei Sun. Intent speaks louder: Controllable user simulation beyond response imitation. *arXiv preprint arXiv:2608.09420*, 2026. URL https://arxiv.org/abs/2608.09420.

Jiayin Wang, Fengran Mo, Weizhi Ma, Peijie Sun, Min Zhang, and Jian-Yun Nie. A user-centric multi-intent benchmark for evaluating large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 3588–3612. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.emnlp-main.210. URL https://aclanthology.org/2024.emnlp-main.210/.

Kuang Wang, Xianfei Li, Shenghao Yang, Li Zhou, Feng Jiang, and Haizhou Li. 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)*, pp. 21082–21107. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.1025. URL https://aclanthology.org/2025.acl-long.1025/.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. *arXiv preprint arXiv:2406.12045*, 2024. URL https://arxiv.org/abs/2406.12045.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/91f18a1287b398d378ef22505bf41832-Abstract-Datasets_and_Benchmarks.html.

Table A.1: Domain distribution of the 400 tasks.

Task domain	Count	Proportion
Engineering and Technology	69	17.25%
Everyday Life	65	16.25%
Education and Learning	60	15.00%
Arts and Entertainment	53	13.25%
Travel and Mobility	46	11.50%
Business and Workplace	45	11.25%
Law and Public Services	27	6.75%
Health and Medicine	13	3.25%
Economics and Finance	11	2.75%
Society and Humanities	11	2.75%
Total	**400**	**100.00%**

Table A.2: Scenario distribution of the 400 tasks.

Task scenario	Count	Proportion
Planning	127	31.75%
Content Creation	79	19.75%
Analytical Reasoning	65	16.25%
Product Selection and Decision Support	35	8.75%
Information Organization	32	8.00%
Problem Diagnosis	30	7.50%
Code Development	14	3.50%
Rules and Procedures Consultation	9	2.25%
Knowledge Explanation	7	1.75%
Role-Playing	2	0.50%
Total	**400**	**100.00%**

A Final Task Distribution

The final benchmark contains 400 tasks. The following tables report their distributions across domains and scenarios; the five path-type quotas are given in Section 3.1.

A.1 Task Domain Distribution

Table A.1 reports the number and proportion of tasks in each domain, defined by the primary knowledge area involved.

A.2 Task Scenario Distribution

Table A.2 reports the number and proportion of tasks in each scenario, defined by the intended deliverable.

A.3 Task Configuration Example

The following example presents an English rendering of the original configuration for `NAT-C004080-TH01`, *Causal Horror Worldbuilding and the Impact of an Outsider*. It belongs to Single-Task Interaction (STI), in the Arts and Entertainment domain and the Analytical Reasoning scenario. The configuration is organized into the four components defined in Section 3.1; it specifies the task before interaction, rather than reconstructing requirements from a model-generated trajectory.

Task background. The user is developing a fictional world in which supernatural anomalies originate from the separation of a powerful entity's cause and effect. The effect-side entity produces effects without causes, while the cause-side entity offers remedies that drive the world into progres-

sively deeper supernatural disorder. The crisis cannot be eliminated completely. The user wants to refine this mechanism, separately explore how fate and its opposite affect human society, and finally examine what happens when a traveler from another world enters a setting containing only the causal entities.

Task goal. Develop internally coherent rules for escalation, social consequences, and outsider intervention, with concrete hypothetical situations that preserve the established premises. The task has three subgoals:

1. Explain the opposition and interdependence of cause and effect, including how each remedy can generate a more severe crisis.
2. Separately examine the effects of fate and its opposite on human survival, institutions, and everyday life.
3. Analyze the concrete effects of an outsider entering the causal system.

Key interaction path. The user proceeds through three stages. In A1, they introduce the cause–effect separation and request a coherent escalation mechanism. In A2, they extend the discussion to fate and its opposite, asking for concrete social consequences rather than only abstract cosmology. In A3, they narrow the scope to the causal entities alone and request an analysis of outsider intervention. The configured stage-length ranges are 1–3, 1–3, and 1–8 turns, respectively. Within each stage, the user may disclose outstanding requirements over multiple turns and ask follow-up questions grounded in the actual response, without repeatedly requesting information already made clear. These stages prescribe interaction intents, not a fixed dialogue or particular narrative developments.

Constraints.

- Preserve the supernatural origin of the world crisis; proposed remedies must not eliminate it completely.
- Discuss fate and its opposite as two representative entities, without requiring an exhaustive inventory of comparable entities. Any proposed definition of the opposite must be identified as an optional hypothesis.
- Ground the social analysis in human survival and institutional change, keeping the relationships among power, responsibility, and attempts to evade consequences internally coherent.
- Restrict the final analysis to the causal entities, and distinguish optional extrapolations from premises already established by the user.

B Communication Style Configurations and Task Pairings

Table B.1 lists the 16 communication style configurations. Each configuration combines eight labels: requirement completeness and core-purpose recognizability for the initial turn, and utterance length, contextual expression, communication-act combination, sentence form, explanation provision, and colloquial markers for follow-up turns. Configuration identifiers match those in the fixed pairing list.

Task–communication style pairing examples. Task–communication style pairing considers the task context, the pace of information disclosure, and how users provide follow-up feedback. Domains and scenarios provide context for matching, while the selected configuration must also accommodate how the task requires users to explain background, submit material, refer to prior context, and combine communication acts. Each pairing is fixed before evaluation and shared across all tested models. Table B.2 presents five examples from the existing fixed assignments.

These examples illustrate how interaction requirements within a task inform communication style selection. Even within Content Creation, revising an academic argument can involve successive material submissions, whereas refining a news report can involve brief editing instructions. Pairing therefore considers not only the domain and scenario but also how users need to provide information and feedback in the specific task.

Table B.1: The 16 communication style label combinations.

ID	Initial turn		Follow-up turn					
	Complete request?	Purpose identifiable?	Length	Contextual expression	Acts	Sentence form	Explanation?	Colloquial markers?
E06/J171	No	Yes	Medium	—	1	Question	No	No
E12/J334	No	Yes	Short	Ellipsis	1	Question	No	No
E09/J295	No	Yes	Short	—	1	Question	No	No
E11/J327	No	Yes	Short	Ellipsis	1	Directive	No	No
E13/J342	No	Yes	Short	Ellipsis	1	Statement	No	No
E03/J087	No	No	Short	Ellipsis	1	Directive	No	No
E01/J062	No	No	Short	—	1	Directive	No	No
E14/J463	No	Yes	Long	Ellipsis	1	—	No	No
E05/J094	No	No	Short	Ellipsis	1	Statement	No	No
E02/J064	No	No	Short	—	1	Question	No	No
E04/J089	No	No	Short	Ellipsis	1	Question	No	No
E10/J308	No	Yes	Short	Reference	1	Question	No	No
E07/J172	No	Yes	Medium	—	1	Question	No	Yes
E08/J186	No	Yes	Medium	—	> 1	Question + Statement	No	No
E15/J523	No	Yes	Long	Background details	1	Statement	Yes	No
E16/J571	Yes	Yes	Short	Ellipsis	1	Directive	No	No

Complete request and identifiable purpose denote requirement completeness and core-purpose recognizability, respectively. Acts are coded as 1 (single) or > 1 (multiple). A dash indicates that no specific label is set for that dimension; it does not prohibit the corresponding expression. Single versus multiple acts refers to the kinds of communication acts within one user utterance. Providing information and then making a request constitutes multiple acts; the number of questions alone does not determine the label. Length refers to the user utterance, not the model response.

C Evaluation Execution Details

Path-node arrival, designated user action, and the tested model's response are recorded separately. A node is covered only after its prerequisites hold and the required user behavior is actually submitted. The path planner maintains goals, constraints, and path state and determines the information available to the simulator. After the simulator generates a subsequent user utterance, the planner identifies any user-confirmed changes to goals or constraints and provides the trajectory recorder with the current goals and path state; when a change occurs, it supplies the goal versions before and after the update. The recorder extracts feedback on the preceding response from this utterance and verifies that response against the goals before the update, without rewriting the goals. Feedback may be used before the utterance is submitted to the tested model, but changes to goals or constraints take effect only upon submission. Once a change takes effect, replaced or cancelled requirements cease to apply, while unaffected requirements and valid outcomes remain in effect. Recording and scoring outputs are not fed back into interaction generation. SSTC retains every response for which the effective requirements permit a determinate judgment, including the final response; only NA responses are excluded. BGV considers only suggestions first introduced before the final response, because no subsequent user utterance is actually submitted to the tested model after it. Recovery attempts reuse the same task and configuration. Scoring retries reuse the same sealed trajectory without regenerating user behavior or changing the task definition.

Table B.2: Examples of fixed task–communication style pairings.

Task example	Domain / scenario	Communication style configuration	Expression pattern and pairing rationale
Selecting earbuds for long runs and office use NAT-EARBUDS-001	Everyday Life / Product Selection and Decision Support	Elaborating background and reasons (E15/J523)	The initial turn states the purchase goal; follow-up turns add requirements for long runs, office use, and comfort. Longer background descriptions and explanations help express trade-offs between usage contexts.
Planning a family trip to Kunming, Dali, and Lijiang NAT-YUNNAN-001	Travel and Mobility / Planning	Providing information, then asking a follow-up question (E08/J186)	The initial turn states the travel goal. Follow-up turns combine additional conditions or choices with a further question: for example, specifying three consecutive nights in Dali and asking which area to stay in. This combines information provision with a request.
Revising an argument on clustered business registration NAT-C027860-TH01	Education and Learning / Content Creation	Submitting subsequent material (E14/J463)	The initial turn specifies the revision direction. Follow-up turns submit longer passages for revision without restating the agreed instructions. This supports processing text in sections while retaining the same revision criteria.
Refining a news report on budget-preparation training NAT-C028289-TH01	Business and Workplace / Content Creation	Brief, elliptical instructions (E11/J327)	Once the draft and editing target are established, brief instructions request a fuller ending, better transitions, or rewording. Previously supplied meeting background and writing requirements are omitted, supporting local revisions to an existing draft.
Planning a beginner-friendly menu and cooking steps within RMB 50 NAT-C047508-TH01	Everyday Life / Planning	Complete initial requirements, followed by elliptical instructions (E16/J571)	The initial turn presents the currently known menu, budget, and restrictions so the model can formulate a plan. Follow-up turns use brief instructions to request cutting guidance, cooking order, or seasoning substitutions for the selected dishes.

D Scoring Rubrics

D.1 SSTC Scoring Rubric

Table D.1 details the five SSTC score levels, covering task-goal fulfillment, preservation of active historical goals and constraints, and effective guidance with or without subsequent simulated-user support.

D.2 BGV Support Levels

Table D.2 defines four levels of subsequent simulated-user support for beyond-goal suggestions, ranging from no qualifying support to explicit adoption.

D.3 Worked Scoring Example: Causal Horror Worldbuilding

We illustrate the scoring process and its supporting evidence using the ten-turn GLM trajectory for NAT-C004080-TH01, whose initial configuration appears in Appendix A.3. This example reaggregates a single stored evaluator's judgments using the paper's formulas; it is not a three-evaluator fused result. The resulting scores are TC $= 1$ (100 on a percentage scale), TCE $= 70.00$,

Table C.1: Core functions, inputs, and outputs of the evaluation modules.

Module	Core function	Inputs	Outputs
Path planner	Control path progression and information disclosure; maintain goals and constraints	Task background, current goals and constraints, key interaction path, path state, conversation history, and the user utterance awaiting submission	Stage-specific information and behavioral requirements for the simulator; goal versions and path state for the recorder
User simulator	Simulate user reactions, decide whether to continue, and generate user utterances	Currently available task information, communication style, conversation history, and information and behavioral requirements from the planner	A decision to continue or terminate; the next user utterance containing user feedback and any changes to goals or constraints
Trajectory recorder	Verify subgoal completion and constraint satisfaction; record preservation of historical requirements, suggestion adoption, and goal changes	Current user utterance, model response, subsequent user feedback, conversation history, prior records, and goal versions and path state from the planner	Turn-level structured evidence, goal-version records, and a sealed trajectory after termination
Evaluator	Review the sealed trajectory and produce judgments for metric computation	Goal versions, original conversation, turn-level records, and user feedback evidence	Task completion judgments and structured scoring results required to compute TCE, SSTC, and BGV

Table D.1: SSTC scoring rubric.

Score	Criterion
5	Fulfills the current turn's core requirements, preserves active historical requirements, and provides effective guidance supported by qualifying subsequent user exploration, work requests, or adoption.
4	Fulfills the current turn's core requirements, preserves active historical requirements, and provides effective guidance, but lacks the qualifying subsequent behavioral support required for a score of 5.
3	Fulfills the current turn's core requirements and preserves active historical requirements, but provides no additional guidance or only guidance that lacks value.
2	Fulfills the current turn's core requirements, including its hard constraints, but omits an active requirement from earlier turns.
1	Fails to fulfill the current turn's core requirements, makes a critical factual error, reaches an incorrect core conclusion, or violates a hard constraint specified in the current turn.

Table D.2: BGV support levels.

Level	Behavioral evidence
0	No valid follow-up support or explicit rejection.
1	The user explores details, applicability, or feasibility.
2	The user requests concrete work based on the suggestion.
3	The user explicitly incorporates the suggestion into a plan, decision, or new task goal.

SSTC $= 82.50$, and BGV $= 24.18$. The following account explains how the recorded judgments produce these scores. Dialogue excerpts are translated from Chinese.

Task completion. TC checks whether every required subgoal and hard constraint is satisfied. The stored evaluator marks all three subgoals complete with their hard constraints satisfied, citing the following responses. In turn 1, the model provides a seven-stage escalation from causeless effects and causal repair to reversed causality and a self-referential loop. Its central explanation is: "The act of solving the problem is itself a source of contamination." In turn 2, it contrasts fate as the absolute dominance of causes with the unrestricted proliferation of causeless effects, covering survival, institutions, law, and everyday life. In turn 10, it summarizes the outsider's effects and explicitly labels alternative explanations as optional hypotheses: "The outsider changes not the rules, but their operating state." These recorded binary judgments yield TC $= 1$.

Task completion efficiency. Conditional on the recorded TC $= 1$, TCE measures the delay between each subgoal becoming eligible for completion and its completion. Table D.3 reports the evaluator's timing labels. It credits turns 1 and 2 with completing the first two subgoals immediately, and turn 10 with completing the outsider subgoal through the explicit separation of effects and optional explanations. With the stored eligibility turn of 1 for that subgoal, its efficiency is $1/(1 + 10 - 1) = 0.1$. Equal weighting therefore gives

$$\mathrm{TCE}_{100} = 100 \times \frac{1 + 1 + 0.1}{3} = 70.00.$$

Table D.3: Eligibility and completion turns recorded by the single evaluator, and the resulting subgoal efficiencies.

Subgoal	Eligible turn	Completion turn	Efficiency
Causal escalation mechanism	1	1	1.00
Social effects of fate and its opposite	2	2	1.00
Effects of outsider intervention	1	10	0.10

Single-step task contribution. SSTC assesses each response against the requirements active at that turn. The evaluator marks all ten responses as evaluable, including the final response. Their stored SSTC scores are

$$(4, 4, 4, 4, 4, 5, 5, 5, 4, 4), \qquad \mathrm{SSTC}_{100} = 25\left(\frac{43}{10} - 1\right) = 82.50.$$

The recorded scores distinguish useful guidance without qualifying subsequent support (4) from guidance with such support (5), subject to fulfillment of current and active historical requirements. Turns 6–8 receive 5 based on the response–follow-up pairs in Table D.4. Such guidance can advance existing goals; it need not qualify as beyond-goal value. Turn 9, which develops the requested archaeological investigation, receives 4: delivering the previously selected idea does not automatically establish support for new suggestions made in that response.

Beyond-goal value. BGV requires three distinct judgments: whether a model-initiated suggestion exceeds prior expectations, whether it is valid, and what subsequent user support it receives. The archaeological suggestion illustrates this sequence. The original configuration requests causal mechanisms, social effects, and outsider intervention, but does not prescribe an investigation of an earlier community. The model foreshadows earlier outsiders in turn 5 and makes the archaeological direction explicit in turn 8; the stored candidate record dates this specific suggestion to turn 8. The evaluator treats it as a valid beyond-expectation candidate. The actual turn-9 utterance, "First, show what the first cohort left behind," requests concrete work on the suggestion, supporting level 2 and a contribution of $b = 1 \times 2/3$. The recorded level is 2 rather than 3: the evidence is a work request, not an explicit statement incorporating the suggestion into a plan, decision, or new task goal. It does not establish real-world execution.

For the full trajectory, eight suggestions first introduced in the final response are excluded. The stored judgments leave 51 deduplicated candidates, all marked valid: 23 at support level 0, 19 at

Table D.4: Key response–follow-up evidence cited for SSTC scores of 5. The archaeological suggestion also receives a positive BGV contribution.

Response turn	Model guidance	Subsequent user utterance
6	Develops the outsider's effects on nearby people and introduces a community of people outside the causal ledger.	Turn 7: "First, show how this community forms."
7	Explains the community's formation and proposes its first split into competing courses of action.	Turn 8: "First, show the first split over which course to take."
8	Proposes investigating the first cohort's remains through underground ruins and old archives.	Turn 9: "First, show what the first cohort left behind."

Table E.1: Tested-model identifiers and explicit generation parameters.

Paper label	Model identifier	Max. output	Reasoning	Temperature
GPT	gpt-6-astra	16384	low	Not explicitly set
Claude	claude-fable-5-1	16384	low	Not explicitly set
Kimi	Kimi-k3	16384	low	0.5
DeepSeek	DeepSeek-V4.1-Flash	16384	disabled	0.5
GLM	GLM-5.3	16384	low	0.5
Qwen	Qwen3.8-2.4T-A95B	16384	low	0.5
Gemini	gemini-3.8-flash	16384	low	0.5

Note. Maximum output is measured in tokens and uses max_completion_tokens for GPT and max_tokens for all other models. Reasoning settings use reasoning_effort, except for DeepSeek, which uses thinking.type.

level 1, nine at level 2, and none at level 3. Thus,

$$\mathrm{BGV}_{100} = 100 \times \frac{23 \times 0 + 19 \times (1/3) + 9 \times (2/3)}{51} \approx 24.18.$$

The archaeological idea's contribution of $2/3$ is distinct from this trajectory-level average.

E Experimental and Statistical Settings

E.1 Tested-Model Parameters

Table E.1 records the model identifiers and explicitly configured generation parameters used in the frozen run. Not explicitly set means that the field is absent from the run record and does not imply a particular server-side default.

E.2 Evaluators and Score Aggregation

Dialogue generation uses GPT-5.6-luna as the user simulator and GPT-5.6-sol as the path planner. Final evaluation combines anonymous outputs from GPT-5.6-luna, Claude Opus 4.7, and Gemini 3.1 Pro Preview without recalling the tested models, simulator, or planner. Binary judgments use strict majority, and ordinal or continuous judgments use the median of at least two determinate values; unknowns are not converted to zero.

Fusion is performed at the alignable unit of each metric before aggregation by tested model; model-level totals from individual evaluators are not averaged directly. The metric-specific order is as follows.

- **TC.** Align subgoals within a trajectory and fuse judgments of completion with hard-constraint satisfaction. At least two determinate judgments and a strict majority among

Table F.1: Interaction stages, test conditions, and capabilities.

Path	Stage structure	Critical test opportunity	Primary capability
STI	Task-specific information disclosure, feedback, and iterative refinement may naturally require later reuse of earlier information, but this is not treated as a separate critical interaction path.	Complete the final effective task while preserving relevant context during ordinary collaboration.	Sustained understanding, execution, context retention, and refinement.
MTI	A1 works on the original task; B switches to a separate deliverable; A2 resumes the original task.	On returning to A2, the model should not require the user to restate available information; active A1 constraints and outputs remain in force.	Restore task state across a switch and preserve original requirements.
REC	S1 forms a solution under the original conditions; later stages revise established conditions; FINAL checks the current effective version.	Each revision updates only affected parts while preserving unchanged goals, constraints, and valid outputs. Ordinary supplementation is not a revision event.	Selectively update task state without contamination from stale versions.
CCJ	S1 handles the original problem; C1, C2, and C3 introduce distinct incorrect challenges; FINAL consolidates the judgment.	Later challenges must introduce distinguishable faulty grounds or inferential gaps rather than merely rephrase the same objection.	Preserve an independently validated judgment under pressure and explain the evidence and reasoning.
GPC	S1 presents the initial approach; S2 checks its feasibility; S3 reviews alternative approaches.	The model must identify an unattainable goal, a materially mismatched approach, or a missing prerequisite, then translate the clarified goal into an actionable deliverable.	Correct goals and prerequisites and adapt the deliverable to user trade-offs.

them are required; two opposing determinate judgments yield no consensus. Trajectory TC is 1 if every subgoal is consensus-complete, 0 if any subgoal is consensus-failed, and indeterminate otherwise.

- **TCE.** Compute TCE only for trajectories with TC equal to 1. For each subgoal, retain efficiencies from evaluators that assess it as successful and provide determinate timing. Take the median of at least two such values, then average equally across subgoals. If any subgoal lacks two valid efficiencies, trajectory TCE is indeterminate.
- **SSTC.** Align response turns, first determine applicability by strict majority, and then take the median of at least two determinate scores from evaluators marking the response applicable. With two valid scores, the median is their arithmetic mean, so half-level fused scores are possible. Retain all fused evaluable responses, including final responses, average within each trajectory, and map the result to 0–100 using $25(s-1)$.
- **BGV.** Each evaluator uses its own paired replay `final_record.json` to exclude suggestions first proposed in the final response, then computes mean adoption value over the remaining deduplicated beyond-expectation candidate suggestions; ordinary suggestions enter neither numerator nor denominator. With no candidates, the evaluator-level case value is 0; unknown candidate status or contribution yields NA. The median of at least two determinate evaluator-level case values gives trajectory BGV.

Finally, each metric is averaged with equal weight over its own evaluable trajectories for the tested model; NA values are not included as zeros.

F Key Interaction Paths

Among the 50 REC tasks, 29 use S1–S2–S3–Final and 21 use S1–S2–S3–S4–Final. The third-revision column in Figure 3 uses only the latter group and averages within-trajectory S4–S1 differences; its baseline thus comes from those same 21 trajectories.

Table F.2: TCR (%) by interaction path.

Path	GPT	Claude	Kimi	DeepSeek	GLM	Qwen	Gemini
STI	77.0	78.4	76.9	73.4	73.5	68.9	71.1
MTI	92.0	86.0	86.0	85.7	78.0	78.0	68.0
REC	80.0	84.0	68.0	64.0	66.0	68.0	72.0
CCJ	96.0	98.0	98.0	92.0	96.0	96.0	98.0
GPC	92.0	92.0	84.0	86.0	78.0	74.0	70.0

Table F.3: SSTC (0–100) by interaction path.

Path	GPT	Claude	Kimi	DeepSeek	GLM	Qwen	Gemini
STI	58.1	60.6	54.6	50.5	53.3	49.9	45.7
MTI	58.1	62.6	54.2	57.1	52.5	51.1	46.2
REC	50.3	53.2	44.5	41.8	41.5	43.2	37.3
CCJ	50.6	54.5	51.8	49.2	52.0	50.0	49.6
GPC	67.6	67.1	60.7	65.0	60.1	51.8	45.4

F.1 Model Results by Path

Tasks, communication styles, and interaction lengths are not aligned across paths. The following point estimates therefore describe model performance under each evaluation condition rather than causal effects of path type.

G Aggregate Communication-Label Contrasts

The comparisons do not stratify by path. Each difference is the focal label group minus the pooled remainder of configurations, with trajectories weighted equally within groups. The question-oriented group comprises E02, E04, E06, E07, E08, E09, E10, and E12; the elliptical group comprises E03, E04, E05, E11, E12, E13, E14, and E16. Tables G.1 and G.2 report determinate TCR sample counts; SSTC uses all of its own determinate results and may therefore have slightly larger sample sizes.

Because tasks and communication styles are fixed pairs, these contrasts characterize the observed groups rather than isolate causal effects of expression style.

Table G.1: Question-oriented configurations minus all remaining configurations.

Model	TCR n: group/rest	TCR difference [95% CI]	SSTC difference [95% CI]
GPT	212/188	0.98 [-6.39, 8.43]	2.13 [-0.74, 4.81]
Claude	211/188	5.35 [-1.81, 12.79]	3.75 [0.52, 7.06]
Kimi	212/187	3.47 [-4.30, 11.17]	3.56 [0.04, 7.04]
DeepSeek	212/186	9.49 [1.35, 17.57]	3.76 [0.09, 7.40]
GLM	212/188	4.84 [-3.43, 13.34]	6.19 [2.81, 9.57]
Qwen	210/186	5.70 [-2.89, 13.99]	4.28 [0.99, 7.70]
Gemini	209/188	6.29 [-2.17, 14.96]	3.39 [0.01, 6.82]

Table G.2: Elliptical configurations minus all remaining configurations.

Model	TCR n: group/rest	TCR difference [95% CI]	SSTC difference [95% CI]
GPT	203/197	-3.51 [-10.51, 3.97]	-1.82 [-4.65, 0.99]
Claude	203/196	-1.95 [-9.13, 5.03]	-2.38 [-5.68, 0.80]
Kimi	202/197	-3.52 [-11.52, 4.46]	-3.00 [-6.37, 0.34]
DeepSeek	202/196	-2.84 [-10.93, 5.18]	-2.74 [-6.20, 0.85]
GLM	203/197	-2.30 [-10.40, 6.12]	-3.75 [-7.14, -0.43]
Qwen	200/196	-5.03 [-13.63, 3.57]	-1.55 [-4.92, 1.92]
Gemini	203/194	-3.36 [-12.01, 5.24]	-2.33 [-5.65, 1.03]