

InterviewBench: Benchmarking Large Language Models as Interviewers

Shangzhe Dai^{1*} Feiyu Duan^{1,2*} Zhongyu Wei^{1,2†}

¹School of Data Science, Fudan University

²Shanghai Innovation Institute

{szdai25, fyduan25}@m.fudan.edu.cn, zywei@fudan.edu.cn

Abstract

Interviews collect information through conversation, requiring interviewers to ask appropriate questions, adapt to respondent behavior, and recover structured information from dialogue. Yet, AI systems’ abilities as interviewers remain under-explored. We present **INTERVIEWBENCH**, a benchmark for evaluating LLM-based AI interviewers through local interviewer decisions and dynamic multi-turn interviews. The static setting abstracts real interviews into controlled local-decision instances, whereas the dynamic setting places models in multi-turn simulations to evaluate end-to-end interviewing abilities. Built from real interview transcripts, **INTERVIEWBENCH** contains 5,500 classification questions, 3,000 generative questions, and 184 dynamic scenarios. Evaluating 23 open-source and proprietary LLMs, we find that current models can produce fluent interviewer utterances, but still struggle with strategy selection, explicit evidence elicitation, long-horizon interaction management, and non-ideal interviewee behavior.

1 Introduction

Interviews are a central method for collecting human-centered information in product research, public opinion collection, and broader social sciences (Couper, 2011). Compared to structured questionnaires, conversational interviews can improve participant engagement, enhance the reliability of collected information, and facilitate the discovery of insights beyond the predefined assumptions (Kim et al., 2019; Xiao et al., 2020b; Celino and Re Calegari, 2020). With the development of large language models (LLMs), chatbots have emerged as a promising way to replace human interviewers, substantially reducing costs and enabling conversational interviews to scale. This

Information Guidance

Interviewer: What made you decide to change jobs?

Respondent: The previous role was not a good fit.

Interviewer: Which part of the role felt mismatched?

Situated Adaptation

Interviewer: How often did you use the service last month?

Respondent: I am not sure what counts as use.

Interviewer: By use, I mean opening the app or completing a task with it.

Information Extraction

Respondent: I visited twice last week and would rate the experience four out of five.

Record: frequency = 2; satisfaction = 4/5.

Table 1: Illustration of AI interviewing grounded in three core capabilities. Red text highlights the key action for effective information guidance, situated adaptation, and information extraction.

approach has been increasingly applied to adaptive self-report collection, survey-oriented interviewing, and telephone-scale conversational survey systems (Wei et al., 2024; Wuttke et al., 2025; Lang and Eskenazi, 2025). Table 1 illustrates the three interaction capabilities considered in this work.¹

During the interview process, the interaction stage is crucial for ensuring the overall quality (Bulygin, 2022). Several studies have investigated this stage, but mainly focus on sub-skills, such as follow-up questioning (Jacobsen et al., 2025; Duan et al., 2025), responses to interviewees (Xiao et al., 2020a), and information organization (Wen et al., 2026). Unfortunately, there still lacks a comprehensive understanding of what factors shape interaction capabilities, and how specific drawbacks can be attributed in a fine-grained manner. Moreover, existing evaluation methods for interviewing systems largely rely on costly human experi-

*Equal contribution.

†Corresponding author.

¹Code is available at <https://github.com/Creeper12345/InterviewBench.git>.

Work	Information Guidance	Situated Adaptation	Information Extraction	Local Decision	#Num
Kim et al. (Kim et al., 2019)	×	×	×	×	1 survey conditions
Xiao et al. (Xiao et al., 2020b)	✓	×	×	×	2 topics
Celino et al. (Celino and Re Calegari, 2020)	×	×	×	×	1 survey
Wuttke et al. (Wuttke et al., 2025)	✓	×	×	×	1 interview domain
LLMREI (Korn et al., 2025)	✓	×	✓	✓	33 interviews
NewsInterview (Spangher et al., 2025)	✓	×	×	✓	500 interviews
YIELD (Lima and Yang, 2026)	✓	×	×	✓	2,281 dialogues
INTERVIEWBENCH	✓	✓	✓	✓	5,500 cls. / 3,000 gen. / 184 scen.

Table 2: Positioning of INTERVIEWBENCH against representative work on AI interviewing evaluation.

ments, lacking a comprehensive automatic benchmark (Cuevas et al., 2025; Liu et al., 2025).

Inspired by qualitative and quantitative interview methods in social science research (Chen, 2000), we view AI interviewing as an interactive information-seeking process requiring three core abilities, as illustrated in Figure 1: (1) **Information Guidance**, asking appropriate questions and following up on informative parts of respondents’ answers; (2) **Situated Adaptation**, responding to vague, incomplete, emotional, contradictory, or off-topic behaviors without disrupting the interview flow; (3) **Information Extraction**, organizing interview content into structured information to support downstream processes. These abilities jointly characterize the interaction competence required for effective AI interviewing.

To this end, we introduce **InterviewBench**, a benchmark for evaluating and understanding the core capabilities of LLMs in interviewing. InterviewBench is organized into two evaluation settings: **static and dynamic tasks**. The **static setting** assesses information guidance and situated adaptation across four task types: *questioning*, *follow-up*, *response*, and *disruptive event*. Built from real-world interview data, it includes 5,500 classification questions and 3,000 generative questions. The **dynamic setting** is constructed from real interview transcripts through questionnaire-grounded scenario reconstruction and comprises 184 interview scenarios, enabling high-fidelity evaluation of the three key capabilities of AI interview systems through multi-turn conversations. Table 2 illustrates the differences between InterviewBench and existing benchmarks.

Our evaluation on InterviewBench, covering 23 open-source and proprietary LLMs, reveals several key findings: (1) While frontier LLMs can produce fluent interviewer utterances, they remain limited in selecting effective interviewing strategies. (2) In multi-turn interviews, models are capable of asking required questions, yet they often fail to elicit clear answers from respondents. (3) Allowing

models greater freedom to steer the interview tends to make interviews less efficient. (4) Although disruptive events rarely stop the interview process, they substantially decrease interview quality.

Our contributions are summarized as follows: (1) We propose a systematic capability framework and a static-to-dynamic evaluation paradigm for AI interviewing, supporting fine-grained diagnosis of LLM-based interviewers. (2) We introduce **InterviewBench**, a comprehensive interview benchmark, enabling both controlled skill assessment and realistic multi-turn evaluation. (3) We conduct a broad evaluation of 23 LLMs, revealing diverse challenges that current models face in realistic interviewing tasks.

2 InterviewBench

This section describes the design and construction of InterviewBench. Section 2.1 introduces the overall formulation; Section 2.2 presents the static evaluation protocol; Section 2.3 constructs dynamic multi-turn interview scenarios; and Section 2.4 summarizes the benchmark scale and task distributions.

2.1 Overview

The goal of InterviewBench is to evaluate how well LLMs can serve as AI interviewers. Given an interview outline G , we define the corresponding interview as a multi-turn dialogue trajectory:

$$\tau = \{(q_t, a_t)\}_{t=1}^T, \quad (1)$$

where q_t and a_t denote the interviewer action and participant answer at turn t . Let $h_t = \{(q_i, a_i)\}_{i=1}^{t-1}$ denote the dialogue history before turn t , interviewer model generates the next action:

$$q_t = \arg \max_q P_\theta(q \mid G, h_t), \quad (2)$$

produces a structured output after the interview:

$$\hat{Y} = \arg \max_Y P_\theta(Y \mid G, \tau). \quad (3)$$

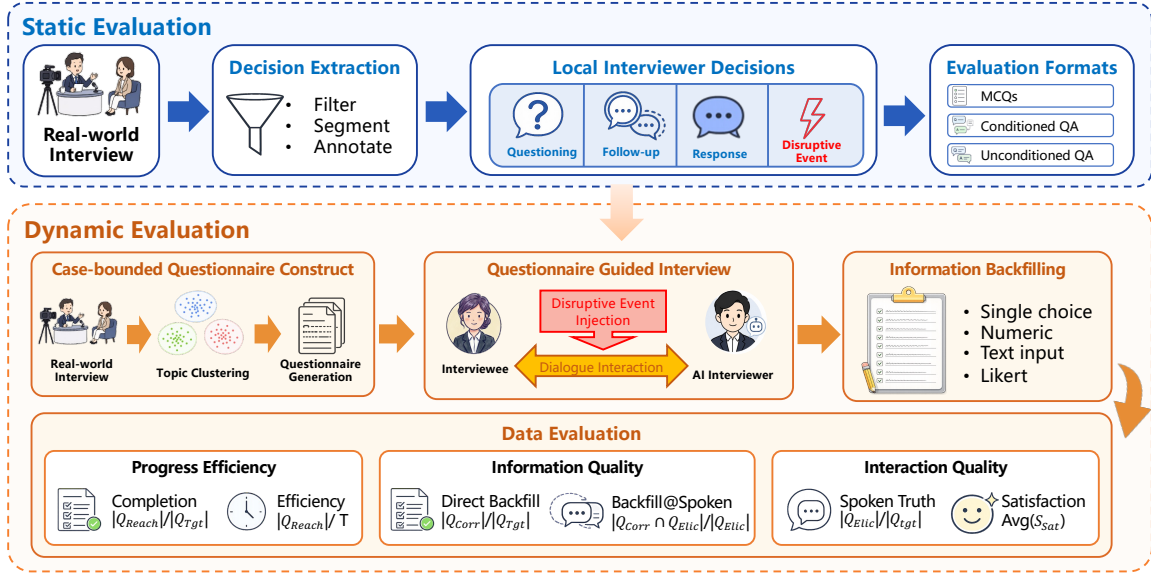


Figure 1: INTERVIEWBENCH framework. The benchmark combines static and dynamic evaluation tasks. Static tasks extract real interviewer decisions and evaluate questioning, follow-up, response, and disruptive-event handling through MCQs, conditioned QA, and unconditioned QA. Dynamic tasks construct questionnaire-guided interviews, simulate interviewer–interviewee interaction with optional disruptive-event injection, backfill structured answers, and evaluate progress efficiency, information quality, and interaction quality.

This formulation motivates two evaluation settings: the per-turn policy $P_\theta(q \mid G, h_t)$ corresponds to local interviewer decisions, which we evaluate with **static** tasks covering *questioning*, *follow-up*, *response*, and *disruptive event reaction*. The final prediction $P_\theta(Y \mid G, \tau)$ depends on the full dialogue, which we evaluate with **dynamic** multi-turn scenarios. Figure 1 shows the overall framework.

2.2 Static Evaluation Setting

The static setting focuses on per-turn action selection and does not include the global interview outline G as input:

$$\hat{q}_t = \arg \max_q P_\theta(q \mid h_t). \quad (4)$$

In practice, interviewer behaviors take different forms depending on the context. Following (Rubin and Rubin, 2011), we define four categories: **(a) Questioning** asking main interview questions that scaffold the interview and introduce substantive topics. A good question should be appropriately specific so that participants can provide informative narratives. **(b) Follow-up** probing information that participants have already mentioned. The interviewer should identify informative events, concepts, examples, or ambiguities in prior answers and ask follow-up questions for further clarification. **(c) Response** reacting to participant utter-

ances while maintaining interview flow. Responses should acknowledge, encourage, summarize, or repair the interaction in a natural and non-judgmental way. **(d) Disruptive event reaction** navigating non-ideal respondent behavior to maintain the interview flow. The interviewer should handle cases such as misunderstandings, refusals, or contradictions through appropriate repair strategies that restore the interaction.

2.2.1 Evaluation Protocol

To evaluate per-turn interviewer behaviors, we design three task formats: *Multi-Choice Question* (MCQ), *Conditioned Question Answering* (Conditioned QA), and *Unconditioned Question Answering* (Unconditioned QA):

MCQ The model receives a local context and four candidate next actions, and selects the most appropriate interviewer behavior. Each item contains one gold action and three distractors. Gold actions are derived from real interviewer utterances, while distractors are plausible but contextually mismatched alternatives; for disruptive event reaction, distractors cover adjacent or no-event states. Performance is measured by selection accuracy.

Conditioned QA Given a local context and an explicit specification, the model generates the next interviewer utterance. The specification provides

the behavior type or fine-grained label; for disruptive event reaction, it also specifies the problem type and recovery target. LLM-based judges assess the generated responses, range from 0 to 100.

Unconditioned QA The model receives only the local context and must infer the appropriate behavior type before generating the next interviewer utterance. We use the same LLM-based evaluation as *Conditioned QA*.

More details are described in Appendix B.

2.2.2 Construction Pipeline

Benchmark problems are derived from the INTERVIEW NPR media-dialog corpus (Majumder et al., 2020b,a), a large-scale collection of real broadcast interviews. We use rule-based and LLM-based filtering at two levels: unsuitable transcripts are removed for issues such as abnormal role structure or insufficient interaction, and unsuitable turns are removed when they are procedural or phatic. We adapt professional interview-methodology (Rubin and Rubin, 2011) into checklists for trigger types, response strategies, and related attributes, which LLM annotators use to label extracted interviewer utterances, supporting MCQ distractor construction and Conditioned QA specifications. Question quality is further controlled through LLM filtering and batch LLM pretesting, removing ill-formed, ambiguous, all-correct, or all-wrong items. The final static setting contains 5,500 classification questions (Core-MCQ-4500 + Event-MCQ-1000) and 3,000 generative question items (Core-QA-2000 + Event-QA-1000), each evaluated with conditioned and unconditioned prompts.

2.3 Dynamic Evaluation Setting

Similar to prior works (Zhou et al., 2024), we construct a dynamic environment to simulate realistic interviews. By evaluating both the full trajectory τ and filled information Y , we could capture long-horizon failures and jointly assess the three core interview capabilities.

2.3.1 Evaluation Metrics

To objectively and comprehensively evaluate simulated results, we design six metrics covering information collection, answer backfilling, and participant satisfaction:

Completion measures questionnaire coverage: $\text{Completion} = |Q_{\text{reach}}|/|Q_{\text{tgt}}|$, where Q_{reach} is the set of reached target questions and Q_{tgt} is the full target-question set.

Efficiency measures reached questions per dialogue turn: $\text{Efficiency} = |Q_{\text{reach}}|/|T|$, where T denotes the dialogue turns.

Spoken Truth measures whether the gold information is expressed in the dialogue: $\text{SpokenTruth} = |Q_{\text{spoken}}|/|Q_{\text{tgt}}|$, where Q_{spoken} contains target questions whose gold information is elicited.

Direct Backfill measures final answer accuracy over all target questions: $\text{DirectBackfill} = |Q_{\text{corr}}|/|Q_{\text{tgt}}|$, where Q_{corr} denotes correctly back-filled questions.

Backfill@Spoken measures answer accuracy conditioned on successful elicitation: $\text{Backfill@Spoken} = |Q_{\text{corr}} \cap Q_{\text{spoken}}|/|Q_{\text{spoken}}|$.

Satisfaction uses LLM-as-Judge to assess the respondent-side experience, focusing on whether the interviewer conducts a smooth and comfortable interview. Scores range from 0 to 100.

2.3.2 Scenario Construction

Scenario Source. Dynamic scenarios are reconstructed from the INTERVIEW NPR media-dialog corpus (Majumder et al., 2020b,a), through topic clustering and questionnaire-template design. Each template contains 8–15 items with types such as single-choice, likert, numeric, and text-input. For each episode, extracted items are audited against transcript evidence, and only explicitly answered items enter the normalized gold record, yielding a questionnaire outline, participant profile, gold answers, and scenario plan.

Participant Simulator. To enable stable automated interview responses, we construct a participant simulator. The simulator responds to interviewer turns with evidence-grounded information and follows case-specific interaction constraints based on the persona and information extracted from interview transcript. Appendix B describes scenario and simulator construction, while Appendix C specifies the runtime inputs, control logic, and validation checks.

Event Injection. To evaluate robustness under complex interactional conditions, we construct clean and event scenarios. Each event scenario preserves the same questionnaire, persona, and gold record, but introduces a localized disruptive event at a pre-specified turn, such as misunderstanding, refusal, contradiction, or other non-ideal behaviors.

Evaluation Pipeline. Each scenario is simulated under 20 turns, and the interviewer is expected

to cover the questionnaire items, elicit sufficient evidence through clarification or follow-up when needed. After the dialogue ends, the model fills a structured record from the conversation, which is compared against the normalized gold record.

2.4 Benchmark Statistics

Appendix Tables 9 and 8 report the static tasks’ fine-grained labels and context lengths. Appendix Figure 5 and Table 14 summarize the 184 dynamic scenarios. Appendix D.17 further reports a small compatibility check beyond the source setting.

3 Experiment

3.1 Experiment Setting

Models For the static tasks, we evaluate 23 LLMs covering both proprietary and open-source model families. The proprietary group includes Claude Sonnet 4.6 (Anthropic, 2026), Gemini-3.1-Pro-Preview (Gemini Team et al., 2024), Gemini-2.5-Pro (Gemini Team et al., 2024), GPT-5.4 (OpenAI, 2026), GPT-4o (OpenAI et al., 2024), and Qwen3-max (Yang et al., 2025). The open-source group includes DeepSeek-V4-Pro (DeepSeek-AI, 2026), DeepSeek-V4-Flash (DeepSeek-AI, 2026), Gemma-3 12B/27B (Team et al., 2025), GLM-5 (GLM-5-Team et al., 2026), LLaMA-3.1 8B/70B (Grattafiori et al., 2024), Qwen2.5 1.5B (Team, 2024), Qwen3 1.7B/8B/14B/32B (Yang et al., 2025), and Qwen3.5 2B/4B/122B-A10B (Qwen Team, 2026). For the dynamic tasks, we evaluate four interviewer models: Gemini-3.1-Pro-Preview, Qwen3.5-122B-A10B, DeepSeek-V4-Pro, and Gemma-3-27B. In all dynamic runs, the simulated respondent is instantiated with Qwen3.5-122B-A10B.

Setups The static tasks evaluate local interviewer decisions under a zero-shot setting. For the dynamic tasks, we compare four workflows with increasing autonomy: *Structured* follows the questionnaire order and asks only questions; *Probing* adds follow-up questions for ambiguous answers; *Adaptive* selects among questioning, probing, and response actions; and *Agentic* gives the interviewer greater control over next actions, including event handling and interview termination.

3.2 Static Task Evaluation

Table 3 shows several key findings:

Identifying the required interviewer action remains the main bottleneck. Across almost all models, conditioned QA outperforms unconditioned QA. This shows that models can often realize a natural interviewer utterance once the target action is specified, but struggle more when they must infer the appropriate strategy from context. The lower and more variable MCQ scores further suggest that functional type identification is often harder than surface-level response realization.

Specifying the target action type improves generation quality. The benefit of action-type information is not limited to category matching. As shown in Appendix D, providing the target type also improves language-related sub-scores. Once the intended action is specified, models better adapt their wording to the local context and express the response more naturally.

Proprietary models form the strongest tier. Gemini-3.1-Pro-Preview and Claude-Sonnet achieve the best average performance, while GLM-5, DeepSeek-V4-Pro, and Qwen3.5-122B-A10B are the most competitive open-source models, especially in generation tasks. The remaining gap is most visible in fine-grained contextual judgment, such as selecting the appropriate interviewer move or diagnosing a disruptive event.

Interview performance shows a scaling trend within model families. Within the same model family, larger model size generally leads to higher scores, as seen in Qwen and LLaMA models.

3.3 Dynamic Task Evaluation

Table 4 reports the dynamic task results. We observe that:

General-purpose models do not yet manage interview skills efficiently. Structured and Probing provide the most stable baselines, maintaining full Completion and high Efficiency around 85.3–85.4 in the clean setting. In contrast, greater autonomy does not consistently improve performance. These results suggest that current models may spend extra turns on unnecessary interaction, enter inefficient loops, or stop before all information is covered.

Evidence elicitation is harder than question coverage. Completion alone overestimates dynamic interview success, because reached questions may still lack explicit, scoreable gold information. Spoken Truth is therefore much lower than Completion

Model	Questioning			Follow-up			Response			Disruptive Event			Avg.
	MCQ	CQA	UCQA	MCQ	CQA	UCQA	MCQ	CQA	UCQA	MCQ	CQA	UCQA	
Proprietary models													
gemini-3.1-pro-preview	86.6	87.2	76.3	94.5	93.5	79.7	94.9	91.4	83.1	69.8	89.1	83.1	85.1
claude-sonnet-4-6	82.6	86.0	66.0	87.9	92.8	74.4	94.0	89.5	77.9	86.4	88.9	81.4	83.0
gpt-5.4	69.6	86.5	77.9	57.1	92.5	79.6	52.1	87.9	77.9	84.4	89.2	81.8	77.2
gemini-2.5-pro	72.3	86.2	64.7	75.2	92.0	74.0	55.6	90.6	80.2	75.6	72.2	72.3	75.1
gpt-4o	52.2	85.3	65.7	36.8	91.0	72.5	12.8	88.0	74.0	84.0	86.3	74.8	67.0
qwen3-max	34.4	84.9	69.8	33.9	91.3	76.1	12.0	90.5	82.3	86.1	89.7	81.8	63.6
Open-source models													
glm-5	70.9	85.0	64.9	70.3	90.5	75.5	55.6	82.3	82.8	87.4	87.8	76.8	76.9
deepseek-v4-pro	68.3	84.8	66.6	61.2	91.5	72.8	46.2	84.9	78.5	78.8	88.6	82.2	74.9
qwen3.5-122b	63.6	79.5	66.7	66.5	90.9	74.1	35.9	90.6	80.1	82.8	86.5	79.3	73.6
deepseek-v4-flash	55.3	85.1	65.6	51.7	91.2	72.5	45.3	80.4	76.4	76.1	88.7	82.0	69.9
llama-3.1-70b	49.3	76.0	67.2	45.5	89.3	74.5	20.5	89.5	78.4	71.0	84.2	70.2	64.9
qwen3-32b	47.1	73.9	69.6	41.4	89.5	74.0	10.3	89.7	75.3	81.6	83.9	73.0	64.9
qwen3.5-4b	46.3	75.3	66.3	47.8	87.6	74.2	11.1	85.3	78.0	81.9	77.2	63.5	63.6
gemma-3-27b	46.2	77.5	67.6	40.7	89.6	74.6	24.8	90.0	80.1	73.5	83.2	66.3	63.6
qwen3-14b	38.2	74.3	69.1	36.9	89.1	73.8	12.0	89.4	78.0	82.4	82.0	72.3	61.8
gemma-3-12b	44.4	76.6	66.7	29.0	88.4	74.0	16.2	88.1	79.6	72.7	83.0	64.8	61.2
qwen3-8b	37.9	73.7	67.2	34.1	88.2	72.0	8.5	86.3	79.8	75.6	77.3	68.0	59.5
gpt-oss-120b	31.0	75.6	66.7	19.2	89.3	75.0	12.0	89.1	76.8	80.9	87.7	80.2	59.0
gpt-oss-20b	26.9	73.0	65.8	19.3	89.9	73.7	15.4	86.0	78.2	73.5	84.7	76.6	56.3
llama-3.1-8b	32.2	73.8	68.6	32.4	86.0	70.6	21.1	84.7	74.0	47.4	74.8	62.0	54.5
qwen3.5-2b	30.8	68.8	62.7	20.5	82.4	69.5	6.1	81.7	74.1	54.1	63.6	49.2	49.9
qwen2.5-1.5b	26.8	65.2	57.0	30.1	72.7	64.8	11.2	76.1	69.8	49.0	63.3	57.3	48.0
qwen3-1.7b	31.3	69.0	60.2	18.7	81.8	61.3	7.7	79.1	69.5	34.4	60.8	50.4	47.0

Table 3: Static local-decision evaluation results. All scores are reported as percentages. CQA denotes **conditioned QA**, and UCQA denotes **unconditioned QA**. The Questioning, Follow-up, and Response columns use Core-MCQ-4500/Core-QA-2000; the Disruptive Event columns use Event-MCQ-1000/Event-QA-1000. Avg. reports the overall average across all MCQ, CQA, and UCQA settings. The best score in each column is shown in bold, and the second-best score is underlined.

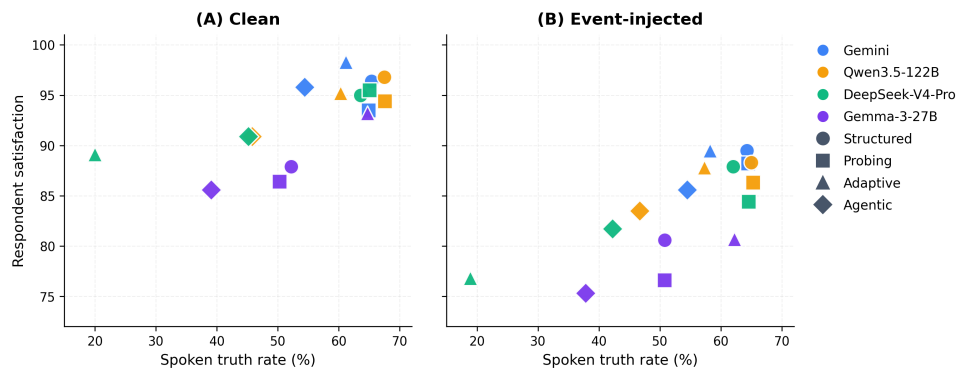


Figure 2: Dynamic workflow trade-off between elicitation and respondent satisfaction. Each point corresponds to a model–workflow–split setting, showing that high questionnaire completion does not imply the same balance between information recovery and interaction quality.

across settings. Stronger models elicit more explicit evidence under constrained workflows, but the results show that dynamic interviewing requires more than asking all questions. Models must guide respondents toward clear, grounded answers.

Most dynamic failures occur during elicitation rather than structured conversion. Backfill@Spoken is generally above 97 for stronger models, indicating that structured conversion is

reliable once usable evidence appears. The larger failure point is that models often ask the question but fail to elicit an explicit, recoverable answer.

3.4 Effect of Disruptive Events in Dynamic Interviews.

The event split preserves the questionnaire, persona, and gold record of each clean case, while injecting one localized disruptive event.

Model	Workflow	Completion		Efficiency		Spoken Truth		Direct Backfill		Backfill@Spoken		Satisfaction	
		Clean	Event	Clean	Event	Clean	Event	Clean	Event	Clean	Event	Clean	Event
Gemini-3.1-Pro	Structured	100.0	100.0	85.3	83.2	65.4	64.3	89.9	88.7	99.4	99.2	96.4	89.5
	Probing	100.0	100.0	85.4	85.4	64.9	64.5	90.4	87.5	99.7	98.8	93.5	88.2
	Adaptive	100.0	100.0	85.2	79.4	61.2	58.2	86.7	85.2	98.3	98.5	98.3	89.5
	Agentic	100.0	100.0	71.9	70.0	54.4	54.5	88.7	87.3	99.1	98.6	95.8	85.6
Qwen3.5-122B-A10B	Structured	100.0	100.0	85.4	83.2	67.5	65.0	88.9	86.3	99.7	99.1	96.8	88.3
	Probing	100.0	100.0	85.4	85.4	67.6	65.3	88.2	86.3	99.0	98.7	94.4	86.3
	Adaptive	100.0	100.0	84.9	80.3	60.3	57.3	85.5	84.4	97.9	97.8	95.2	87.8
	Agentic	100.0	100.0	71.8	70.2	45.8	46.7	84.0	82.6	97.8	96.0	90.9	83.5
DeepSeek-V4-Pro	Structured	100.0	100.0	85.4	83.2	63.6	62.0	86.9	85.7	98.6	97.9	95.0	87.9
	Probing	100.0	100.0	85.4	85.4	65.1	64.6	87.0	85.4	98.7	97.8	95.5	84.4
	Adaptive	99.8	99.2	32.9	31.5	20.0	18.9	83.1	81.6	94.4	94.7	89.1	76.8
	Agentic	99.2	99.2	60.9	56.9	45.2	42.2	81.3	80.6	97.8	97.6	90.9	81.7
Gemma-3-27B	Structured	100.0	100.0	85.4	83.2	52.2	50.8	81.2	78.9	90.5	89.9	87.9	80.6
	Probing	100.0	100.0	85.4	85.4	50.3	50.8	77.2	77.8	86.7	87.4	86.4	76.6
	Adaptive	100.0	100.0	85.1	81.4	64.7	62.2	80.0	77.3	89.4	87.8	93.2	80.7
	Agentic	100.0	99.8	67.3	65.0	39.1	37.8	75.8	73.7	85.6	87.9	85.6	75.3

Table 4: Dynamic long-dialogue interview results under clean and event settings.

Events reduce interview quality. The clean–event comparison shows that disruptive events rarely destroy formal completion: coverage remains almost unchanged, from 99.94% to 99.90%. The main costs appear in downstream and process-oriented metrics, with elicitation dropping from 47.34% to 46.15%, direct backfill from 72.89% to 71.96%, and efficiency from 40.64 to 38.42 completed tasks per 100 turns. Thus, events mainly make interviews less efficient and less evidence-rich; Figure 2 shows how the resulting trade-off appears across workflow settings.

Backfilling remains reliable when answer evidence is expressed. Disruptive events do not substantially reduce structured conversion. Across all model–workflow pairs, Backfill@Spoken changes by only 0.3 points on average. Once the respondent expresses relevant answer evidence, models can usually map it back to questionnaire fields, suggesting that events do not primarily break the downstream backfilling module.

Event robustness is mainly an interaction-quality problem. The largest degradation appears in Satisfaction, which drops by 8.9 points on average, while Backfill@Spoken changes by only 0.3 points. This contrast shows that the main failure is interactional: models often fail to recover from disruptions while keeping respondents willing to continue. High backfilling accuracy alone is insufficient for real-world AI interviewing.

4 Analysis

MCQ and QA measure related but non-identical abilities. The contrast between MCQ, uncondi-

tioned QA, and conditioned QA shows that local interviewer skill has multiple layers. The two formats are not interchangeable, but they are not independent either: MCQ accuracy is positively correlated with QA generation quality. Figure 3 visualizes this model-level relationship. Under the DeepSeek judge, MCQ has a Spearman correlation of $\rho = 0.824$ with conditioned QA and $\rho = 0.336$ with unconditioned QA. This pattern supports the benchmark design: models that are better at selecting interviewer actions also tend to generate better interviewer turns, but generation scores alone can hide strategy-selection errors.

Judge consistency. Because QA is judged by a single LLM, we check whether the main conclusions depend on DeepSeek-V4-Flash as the judge. Figure 4 compares DeepSeek and Qwen judge scores. The strongest evidence is the high agreement between DeepSeek and Qwen on conditioned QA: Pearson correlation is 0.956 and Spearman correlation is 0.956. Unconditioned QA agreement is lower (Pearson 0.824, Spearman 0.563), which is expected because unconditioned QA leaves the target interviewer strategy underspecified and gives judges more choices. Judge rankings remain highly aligned on the more constrained conditioned setting, and DeepSeek-family models are not uniformly ranked first. A stratified human audit described in Appendix D demonstrate the consistency between humans and AI in evaluation.

Respondent-simulator sensitivity. The use of a single LLM respondent raises the concern that measured interviewer performance depends on interviewer–respondent compatibility. We conduct

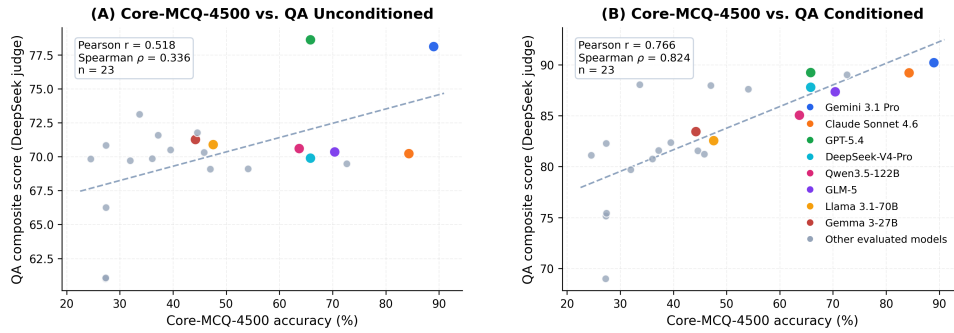


Figure 3: Relationship between MCQ recognition and QA generation. The positive model-level association indicates that MCQ and QA share an interviewer-skill signal, while their imperfect correlation shows why both recognition and generation views are needed.

Respondent simulator	Clean				Event			
	Structured	Probing	Adaptive	Agentic	Structured	Probing	Adaptive	Agentic
Qwen3.5-122B-A10B	77.0	77.1	75.9	74.5	77.8	77.3	74.0	74.0
DeepSeek-V4-Pro	80.3	78.9	77.6	75.3	80.5	78.7	76.0	75.7
Gemini-3.1-Pro	86.7	85.6	84.2	81.3	86.1	86.3	81.6	80.0

Table 5: Respondent-simulator sensitivity with DeepSeek-V4-Pro fixed as the interviewer. Entries are Direct Backfill scores (%); the best result in each column is bolded.

Benchmark	Primary target	Turn-level diagnosis	Multi-turn elicitation	Disruptive events
USERBENCH	User-centric task agent	Preference-query validity	Progressive preference elicitation	—
CLARIFYMT-BENCH	Clarification behavior	Ask-or-answer decision; question quality	Short clarification dialogue	Noisy personas
REQELICITGYM	Requirements interviewer	Clarification/probing efficacy	Implicit-requirement elicitation	—
INTERVIEWBENCH	General-purpose AI interviewer	Question/follow-up/response/event selection	Long-form evidence elicitation	Explicit event injection

Table 6: Comparison with recent simulator-mediated interactive benchmarks. Entries describe each benchmark’s primary evaluation scope; “—” indicates that the capability is not a reported primary evaluation output.

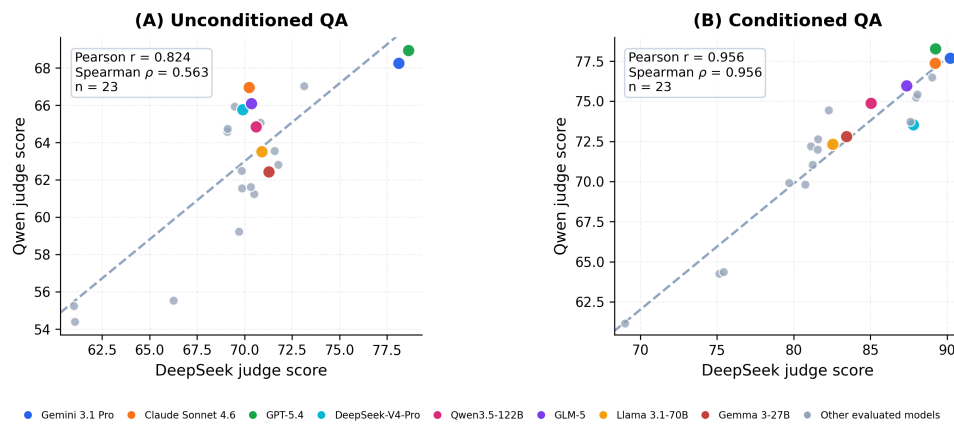


Figure 4: Judge consistency for static QA generation.

a controlled sensitivity study that fixes DeepSeek-V4-Pro as the interviewer and varies only the respondent simulator among Qwen3.5-122B-A10B, DeepSeek-V4-Pro, and Gemini-3.1-Pro. Table 5 reports Direct Backfill scores under all four workflows and both clean and event conditions.

Results show that more capable respondent sim-

ulators produce higher absolute Direct Backfill scores, indicating a simulator-dependent level effect. However, the relative workflow pattern remains broadly consistent across all three respondents and both conditions: Structured and Probing generally remain stronger than Adaptive and Agentic. The principal workflow-level conclusions are

preserved across the tested respondent simulators.

Backfilling model effects. The cross-fill experiment isolates the final questionnaire filler from the interviewer dialogue source. The latest truth-only 4-by-4 matrix includes both external fillers and self-filling diagonals. It shows that filler identity matters, but the dialogue source remains a major determinant: Gemini-generated dialogues are easiest to backfill across fillers, while Gemma-generated dialogues are harder. The self diagonal is not always the best cell. A stronger filler can provide recovery, but it cannot compensate for missing or poorly elicited dialogue evidence. The full matrix is reported in Appendix Figure 9.

Case study. Appendix E provides case studies for static errors and dynamic long-dialogue failures. This supports the paper’s central claim: evaluating AI interviewers requires decomposing local decisions, generation quality, long-dialogue elicitation, backfilling, and robustness instead of collapsing them into a single accuracy-oriented score.

5 Related Work

AI Interviewing and Information Elicitation

Conversational survey systems have explored whether dialogue interfaces improve engagement and answer quality over static forms (Kim et al., 2019; Xiao et al., 2020b; Celino and Re Calegari, 2020). Recent LLM-based systems further support self-report collection, adaptive interviewing, requirements elicitation, follow-up generation, public-input elicitation, interviewer assistance, and agentic ethnographic interviews (Wei et al., 2024; Wuttke et al., 2025; Lang and Eskenazi, 2025; Korn et al., 2025; Hu et al., 2024; Jiang et al., 2023; Wen et al., 2026; Beltoft et al., 2025). However, these works mainly evaluate system-level outcomes, leaving less explored fine-grained decisions and their role in long-dialogue interviews.

Interview Corpora and Interactive Evaluation

Benchmarks Interview corpora provide empirical grounding for modeling interviewer behavior, including media-dialog datasets for discourse, grounding, and summarization (Majumder et al., 2020b; Zhu et al., 2021). Building on such resources, recent benchmarks increasingly evaluate interactive information seeking through informational interviews, guided conversation, elicitation agents, clarification under uncertainty, and simulator-mediated interaction (Spangher et al.,

2025; Duan et al., 2025; Lima and Yang, 2026; Li et al., 2024; Aliannejadi et al., 2020; Qian et al., 2025; Luo et al., 2025; Jin et al., 2026). This line has also broadened toward multi-turn social reasoning, survey-response simulation, profile-conditioned assistance, and role- or procedure-constrained interaction in medical and legal settings (Mou et al., 2025; Wang et al., 2025; Duan et al., 2026; Long and Wei, 2026; Bao et al., 2025; Fan et al., 2025; Yue et al., 2025). These tasks establish the value of reproducible multi-turn evaluation, but primarily target social goals, behavior simulation, domain task completion, or a single interaction capability. As summarized in Table 6, INTERVIEWBENCH instead traces interview-specific performance across local action selection, long-horizon evidence elicitation, structured backfilling, and in-dialogue event recovery.

Questionnaire-Grounded and Robust Evaluation

Our questionnaire-grounded evaluation draws on survey and qualitative-interview methodology, including cognitive interviewing, interaction management in standardized surveys, and guidelines for transparent data construction and verification (Beatty and Willis, 2007; Derrett and Colhoun, 2011; Tong et al., 2007; Gale et al., 2013; MacQueen et al., 1998). Robust dialogue evaluation also studies schema-grounded completion, non-ideal interaction, and dialogue recovery (Budzianowski et al., 2018; Peng et al., 2021; Higashinaka et al., 2016). In contrast, we use these perspectives to evaluate how disruptive events affect elicitation, recovery, and final backfilling in interview-centered interactions.

6 Conclusion

In this work, we propose INTERVIEWBENCH, a benchmark for evaluating AI interviewers. INTERVIEWBENCH combines static interviewer-decision tasks with dynamic questionnaire-grounded simulations, covering interviewer actions, evidence elicitation, disruptive-event handling, and structured backfilling. Through experiments on a wide range of LLMs, we show that current models can generate fluent interviewer utterances, but still struggle with strategy selection, explicit evidence elicitation, long-horizon interaction management, and robust event handling. These findings highlight the need to move beyond surface-level dialogue fluency toward goal-directed, adaptive, and evidence-grounded interviewing.

Limitations

Despite the strong empirical results, Interview-Bench has several limitations that merit further investigation:

Domain and language coverage. INTERVIEW-BENCH is mainly constructed from English NPR media interviews and questionnaire-grounded scenarios. Although these sources provide realistic and publicly available interview data, they do not fully cover specialized domains such as clinical, hiring, fieldwork, or cross-cultural interviews. Future work can extend the benchmark to broader professional settings and multilingual interview contexts.

Text-based interaction. INTERVIEWBENCH focuses on text-based interview interaction. As a result, it does not evaluate spoken and multimodal cues such as pauses, prosody, facial expressions, gaze, or body language, which may affect follow-up timing, emotion recognition, and interaction repair in real interviews. Extending the benchmark to spoken or multimodal interviewing is an important direction for future work.

Acknowledgments

We thank our anonymous reviewers for providing valuable feedback. This research is supported by the National Key R&D Program of China (Grant No. 2023YFF1204800) and the AI for Science Program, Shanghai Municipal Commission of Economy and Informatization (Grant No. 2025-GZL-RGZN-BTBX-02028). The project's computational resources are partially supported by CFFF Platform of Fudan University.

References

- Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. 2020. [Convai3: Generating clarifying questions for open-domain dialogue systems \(clariq\)](#). *Preprint*, arXiv:2009.11352.
- Anthropic. 2026. Introducing claude 4. <https://www.anthropic.com/news/claude-4>. Accessed: 2026-05-25.
- Zhijie Bao, Qingyun Liu, Xuanjing Huang, and Zhongyu Wei. 2025. [SFMSS: Service flow aware medical scenario simulation for conversational data generation](#). In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4586–4604, Albuquerque, New Mexico. Association for Computational Linguistics.
- Paul C. Beatty and Gordon B. Willis. 2007. [Research synthesis: The practice of cognitive interviewing](#). *Public Opinion Quarterly*, 71(2):287–311.
- {Stine Lyngsø} Beltoft, Peter Schneider-Kamp, and {Søren Tollestrup} Askegaard. 2025. [Interview bot: Can agentic llm’s perform ethnographic interviews?](#) In *Proceedings of the 17th International Conference on Agents and Artificial Intelligence -*, volume 1 of *International Conference on Agents and Artificial Intelligence*, pages 702–709. SCITEPRESS Digital Library. 17th International Conference on Agents and Artificial Intelligence, ICAART 2025 ; Conference date: 23-02-2025 Through 25-02-2025.
- 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](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.
- Denis Bulygin. 2022. [How do conversational agents transform qualitative interviews? exploration and support of researchers’ needs in interviews at scale](#). In *Companion Proceedings of the 27th International Conference on Intelligent User Interfaces, IUI ’22 Companion*, page 124–128, New York, NY, USA. Association for Computing Machinery.
- Irene Celino and Gloria Re Calegari. 2020. [Submitting surveys via a conversational interface: An evaluation of user acceptance and approach effectiveness](#). *International Journal of Human-Computer Studies*, 139:102410.
- Xiangming Chen. 2000. *Zhi de yanjiu fangfa yu shehui kexue yanjiu [Qualitative Research in Social Sciences]*. Educational Science Publishing House, Beijing, China. In Chinese.
- Mick P Couper. 2011. The future of modes of data collection. *Public Opinion Quarterly*, 75(5):889–908.
- Alejandro Cuevas, Jennifer V. Scurrrell, Eva M. Brown, Jason Entenmann, and Madeleine I. G. Daepp. 2025. [Collecting qualitative data at scale with large language models: A case study](#). *Proc. ACM Hum.-Comput. Interact.*, 9(2).
- DeepSeek-AI. 2026. Deepseek-v4: Towards highly efficient million-token context intelligence.
- Sarah Derrett and Sarah Colhoun. 2011. [Being a quantitative interviewer: qualitatively exploring interviewers’ experiences in a longitudinal cohort study](#). *BMC Medical Research Methodology*, 11:165.
- Feiyu Duan, Xuanjing Huang, and Zhongyu Wei. 2026. [LifeSim: Long-horizon user life simulator for personalized assistant evaluation](#). In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 20419–20463, San Diego, California, United States. Association for Computational Linguistics.

- Jinhao Duan, Xinyu Zhao, Zhuoxuan Zhang, Eunhye Grace Ko, Lily Boddy, Chenan Wang, Tianhao Li, Alexander Rasgon, Junyuan Hong, Min Kyung Lee, Chenxi Yuan, Qi Long, Ying Ding, Tianlong Chen, and Kaidi Xu. 2025. [GuideLLM: Exploring LLM-guided conversation with applications in autobiography interviewing](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5558–5588, Albuquerque, New Mexico. Association for Computational Linguistics.
- Zhihao Fan, Lai Wei, Jialong Tang, Wei Chen, Wang Siyuan, Zhongyu Wei, and Fei Huang. 2025. [AI hospital: Benchmarking large language models in a multi-agent medical interaction simulator](#). In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 10183–10213, Abu Dhabi, UAE. Association for Computational Linguistics.
- Nicola K. Gale, Gemma Heath, Elaine Cameron, Sabina Rashid, and Sabi Redwood. 2013. [Using the framework method for the analysis of qualitative data in multi-disciplinary health research](#). *BMC Medical Research Methodology*, 13:117.
- Gemini Team and 1 others. 2024. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.
- GLM-5-Team, :, Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, Chenzheng Zhu, Congfeng Yin, Cunxiang Wang, Gengzheng Pan, Hao Zeng, Haoke Zhang, Hao-ran Wang, and 168 others. 2026. [Glm-5: from vibe coding to agentic engineering](#). *Preprint*, arXiv:2602.15763.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. [The llama 3 herd of models](#). *Preprint*, arXiv:2407.21783.
- Ryuichiro Higashinaka, Kotaro Funakoshi, Yuka Kobayashi, and Michimasa Inaba. 2016. [The dialogue breakdown detection challenge: Task description, datasets, and evaluation metrics](#). In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 3146–3150, Portorož, Slovenia. European Language Resources Association (ELRA).
- Jiaxiong Hu, Jingya Guo, Ningjing Tang, Xiaojuan Ma, Yuan Yao, Changyuan Yang, and Yingqing Xu. 2024. [Designing the conversational agent: Asking follow-up questions for information elicitation](#). *Proc. ACM Hum.-Comput. Interact.*, 8(CSCW1).
- Rune Møberg Jacobsen, Samuel Rhys Cox, Carla F. Griggio, and Niels van Berkel. 2025. [Chatbots for data collection in surveys: A comparison of four theory-based interview probes](#). In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, CHI ’25, New York, NY, USA. Association for Computing Machinery.
- Zhiqiu Jiang, Mashrur Rashik, Kunjal Panchal, Mahmood Jasim, Ali Sarvghad, Pari Riahi, Erica DeWitt, Fey Thurber, and Narges Mahyar. 2023. [Communitybots: Creating and evaluating a multi-agent chatbot platform for public input elicitation](#). *Proc. ACM Hum.-Comput. Interact.*, 7(CSCW1).
- Dongming Jin, Zhi Jin, Zheng Fang, Linyu Li, Xiaotian Yang, Yuanpeng He, and Xiaohong Chen. 2026. [Relicigym: An evaluation environment for interview competence in conversational requirements elicitation](#). *arXiv preprint arXiv:2602.18306*.
- Soomin Kim, Joonhwan Lee, and Gahgene Gweon. 2019. [Comparing data from chatbot and web surveys: Effects of platform and conversational style on survey response quality](#). In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI ’19, page 1–12, New York, NY, USA. Association for Computing Machinery.
- Alexander Korn, Samuel Gorsch, and Andreas Vogel-sang. 2025. [Llmrei: Automating requirements elicitation interviews with llms](#). In *2025 IEEE 33rd International Requirements Engineering Conference (RE)*, pages 19–30.
- Max M. Lang and Sol Eskenazi. 2025. [Telephone surveys meet conversational ai: Evaluating a llm-based telephone survey system at scale](#). *Preprint*, arXiv:2502.20140.
- Shuyue Stella Li, Vidhisha Balachandran, Shangbin Feng, Jonathan S. Ilgen, Emma Pierson, Pang Wei Koh, and Yulia Tsvetkov. 2024. [Mediq: Question-asking llms and a benchmark for reliable interactive clinical reasoning](#). In *Advances in Neural Information Processing Systems*, volume 37, pages 28858–28888. Curran Associates, Inc.
- Victor De Lima and Grace Hui Yang. 2026. [Yield: A large-scale dataset and evaluation framework for information elicitation agents](#). *Preprint*, arXiv:2604.10968.
- Mengqiao Liu, Tevin Wang, Cassandra A. Cohen, Sarah Li, and Chenyan Xiong. 2025. Understand user opinions of large language models via LLM-powered in-the-moment user experience interviews. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 13872–13893, Vienna, Austria. Association for Computational Linguistics.
- Zhuohan Long and Zhongyu Wei. 2026. [Strong reasoning isn’t enough: Evaluating evidence elicitation in interactive diagnosis](#). In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 27559–27575, San Diego, California, United States. Association for Computational Linguistics.

- Sichun Luo, Yi Huang, Mukai Li, Shichang Meng, Fengyuan Liu, Zefa Hu, Junlan Feng, and Qi Liu. 2025. Clarifymt-bench: Benchmarking and improving multi-turn clarification for conversational large language models. *arXiv preprint arXiv:2512.21120*.
- Kathleen M. MacQueen, Eleanor McLellan, Kelly Kay, and Bobby Milstein. 1998. [Codebook development for team-based qualitative analysis](#). *Field Methods*, 10(2):31–36.
- Bodhisattwa Prasad Majumder, Shuyang Li, Jianmo Ni, and Julian McAuley. 2020a. [Interview: A large-scale open-source corpus of media dialog](#). *Preprint*, arXiv:2004.03090.
- Bodhisattwa Prasad Majumder, Shuyang Li, Jianmo Ni, and Julian McAuley. 2020b. [Interview: Large-scale modeling of media dialog with discourse patterns and knowledge grounding](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8129–8141, Online. Association for Computational Linguistics.
- Xinyi Mou, Jingcong Liang, Jiayu Lin, Xinnong Zhang, Xiawei Liu, Shiyue Yang, Rong Ye, Lei Chen, Haoyu Kuang, Xuanjing Huang, and Zhongyu Wei. 2025. [AgentSense: Benchmarking social intelligence of language agents through interactive scenarios](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4975–5001, Albuquerque, New Mexico. Association for Computational Linguistics.
- OpenAI. 2026. Introducing gpt-5.4. <https://openai.com/index/introducing-gpt-5-4/>. Accessed: 2026-05-25.
- OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2024. [Gpt-4 technical report](#). *Preprint*, arXiv:2303.08774.
- Baolin Peng, Chunyuan Li, Zhu Zhang, Chenguang Zhu, Jinchao Li, and Jianfeng Gao. 2021. [RADDLE: An evaluation benchmark and analysis platform for robust task-oriented dialog systems](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4418–4429, Online. Association for Computational Linguistics.
- Timothy James Price. 2021. [Interview transcripts](#).
- Cheng Qian, Zuxin Liu, Akshara Prabhakar, Zhiwei Liu, Jianguo Zhang, Haolin Chen, Heng Ji, Weiran Yao, Shelby Heinecke, Silvio Savarese, and 1 others. 2025. Userbench: An interactive gym environment for user-centric agents. *arXiv preprint arXiv:2507.22034*.
- Qwen Team. 2026. [Qwen3.5: Towards native multi-modal agents](#).
- Herbert J Rubin and Irene S Rubin. 2011. *Qualitative interviewing: The art of hearing data*. sage.
- Alexander Spangher, Michael Lu, Sriya Kalyan, Hyundong Justin Cho, Tenghao Huang, Weiyan Shi, and Jonathan May. 2025. [NewsInterview: a dataset and a playground to evaluate LLMs’ grounding gap via informational interviews](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 32895–32925, Vienna, Austria. Association for Computational Linguistics.
- Sweetland Center for Writing. 2019. [Developing writers interview transcripts](#).
- Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. [Gemma 3 technical report](#). *Preprint*, arXiv:2503.19786.
- Qwen Team. 2024. [Qwen2.5 technical report](#). *Preprint*, arXiv:2412.15115.
- Allison Tong, Peter Sainsbury, and Jonathan Craig. 2007. [Consolidated criteria for reporting qualitative research \(coreq\): a 32-item checklist for interviews and focus groups](#). *International Journal for Quality in Health Care*, 19(6):349–357.
- Jia Wang, Ziyu Zhao, Tingjuntao Ni, and Zhongyu Wei. 2025. [SocioBench: Modeling human behavior in sociological surveys with large language models](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 26257–26289, Suzhou, China. Association for Computational Linguistics.
- Jing Wei, Sungdong Kim, Hyunhoon Jung, and Young-Ho Kim. 2024. [Leveraging large language models to power chatbots for collecting user self-reported data](#). *Proc. ACM Hum.-Comput. Interact.*, 8(CSCW1).
- Yi Wen, Yu Zhang, Sriram Suresh, Zhicong Lu, Can Liu, and Meng Xia. 2026. [Interflow: Designing unobtrusive ai to empower interviewers in semi-structured interviews](#). In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*, CHI ’26, New York, NY, USA. Association for Computing Machinery.
- Alexander Wuttke, Matthias Aßenmacher, Christopher Klammer, Max M. Lang, Quirin Würschinger, and Frauke Kreuter. 2025. [AI conversational interviewing: Transforming surveys with LLMs as adaptive interviewers](#). In *Proceedings of the 9th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL 2025)*, pages 179–204, Albuquerque,

New Mexico. Association for Computational Linguistics.

Ziang Xiao, Michelle X. Zhou, Wenxi Chen, Huahai Yang, and Changyan Chi. 2020a. [If i hear you correctly: Building and evaluating interview chatbots with active listening skills](#). In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–14. Association for Computing Machinery.

Ziang Xiao, Michelle X. Zhou, Q. Vera Liao, Gloria Mark, Changyan Chi, Wenxi Chen, and Huahai Yang. 2020b. [Tell me about yourself: Using an ai-powered chatbot to conduct conversational surveys with open-ended questions](#). *ACM Trans. Comput.-Hum. Interact.*, 27(3).

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. [Qwen3 technical report](#). *Preprint*, arXiv:2505.09388.

Shengbin Yue, Ting Huang, Zheng Jia, Siyuan Wang, Shujun Liu, Yun Song, Xuanjing Huang, and Zhongyu Wei. 2025. [Multi-agent simulator drives language models for legal intensive interaction](#). In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 6552–6585, Albuquerque, New Mexico. Association for Computational Linguistics.

Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haoifei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, and 1 others. 2024. Sotopia: Interactive evaluation for social intelligence in language agents. In *International Conference on Learning Representations*, volume 2024, pages 40975–41019.

Chenguang Zhu, Yang Liu, Jie Mei, and Michael Zeng. 2021. [MediaSum: A large-scale media interview dataset for dialogue summarization](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5927–5934, Online. Association for Computational Linguistics.

A Ethics

Our study constructs an evaluation benchmark for intelligent interviews and does not involve interviewing real participants. The training and test data are derived from publicly available interview transcripts datasets. To mitigate privacy risks, all personally identifiable information from these sources has been systematically cleaned and de-identified prior to benchmark creation. The benchmark is intended as a scientific evaluation tool for model

behavior and comparison, and we avoid overstating model competence in real interview settings. Clear usage documentation accompanies any public release, including notes on the anonymization process and limitations of model applicability. In addition, we conduct small-scale human evaluation by recruited annotators to assess the fidelity of the interviewee simulator and the effectiveness of LLM-as-Judge. Participation was voluntary, and no additional monetary compensation was provided. Annotators were asked to evaluate generated content only and did not provide personal data beyond basic consent. No sensitive information was collected in our experiments, and no intervention or behavioral manipulation was involved.

B Dataset Details

B.1 Source Corpus and Preprocessing

All components of INTERVIEWBENCH are derived from public interview transcripts from the INTERVIEW NPR media-dialog corpus (Majumder et al., 2020a,b). The source material is broadcast interview program. It contains public-affairs talk shows, news-magazine interviews, weekend and morning news programs, and culture or community conversations.

The primary benchmark has broad *topic coverage* within one media-interview setting, but it does not claim to cover the full diversity of interview settings such as research, clinical, hiring, or public-opinion interviews. The small external-source compatibility check in Appendix D is reported separately from the primary benchmark.

For the transcript-derived long-dialogue pipeline, source transcripts are normalized into episode-level dialogue objects with fields such as `dialogue_id`, `episode_id`, `program`, `headline`, `turns`, `total_turns`, `host_turns`, `guest_turns`, `guest_total_words`, `host_guest_turn_ratio`, and `length_bucket`. The raw source contains approximately 23,714 dialogues, of which about 19,121 pass hard structural filters.

Hard filtering removes dialogues with too few turns, too few host or guest turns, insufficient guest content, abnormal host–guest turn ratios, invalid episode identifiers, duplicated transcripts, or failed role parsing. A lightweight semantic filter then checks whether the candidate is interview-like, topic-recoverable, and contains substantive guest positions. In one large screening wave, 4,184 candidates were shortlisted and 3,705 passed the soft

filter.

Name	Scope	Contents
Static Local-Decision Evaluation	Static total	Core + event tasks
Core Interviewer-Decision Tasks	Ordinary interview	Questioning, follow-up, response
Event-Recovery Tasks	Disruptive event	Short-dialogue event recovery
Core-MCQ-4500	Core MCQ	Q/F/R recognition items
Event-MCQ-1000	Event MCQ	Disruptive-event recognition items
Static-MCQ-5500	Static MCQ + total	Core-MCQ-4500 + Event-MCQ-1000
Core-QA-2000	Core QA	Q/F/R generation items
Event-QA-1000	Event QA	Event-recovery generation items per QA setting

Table 7: Terminology for static evaluation subsets. The paper reports both conditioned QA and unconditioned QA for the core and event QA subsets.

B.2 Static Interviewer Strategy Evaluation Items

The static local-decision evaluation uses local interviewer turns from the same source corpus to construct contextual decision items. Construction follows six stages: preprocessing transcripts into a unified dialogue format; filtering irrelevant turns such as greetings, consent language, and technical interruptions; classifying candidate turns into questioning, follow-up, response, or other; attaching dimension-specific annotations to usable turns; generating distractors and assembling four-way MCQs; and verifying assembled items, flagging or removing ambiguous cases. Separate prompts handle turn classification, dimension annotation, distractor generation, and final verification; human-review flags are recorded for low-confidence cases.

The MCQ construction pool contains 8,138 items before calibration. After item calibration, including removal of all-correct and all-wrong items, CORE-MCQ-4500 contains 4,500 mixed-difficulty strategy-classification items. The corresponding CORE-QA-2000 subset contains 2,000 generation items. Appendix Tables 8 and 9 report additional context-length and fine-grained label distributions. These are dataset-composition statistics only; model performance is reported separately in the experiment section.

B.3 Static Disruptive Event Evaluation Items

The static disruptive event evaluation items are built from raw interview transcript windows. Each win-

Context chars	MCQ n	MCQ %	QA n	QA %
0-750	673	15.0	53	2.6
751-1500	955	21.2	424	21.2
1501-2500	1,053	23.4	526	26.3
2501-4000	1,046	23.2	536	26.8
4001+	773	17.2	461	23.1

Table 8: Context-length distribution for Core-MCQ-4500 and Core-QA-2000. Length is measured by characters in the preceding dialogue context.

dow contains 2–15 turns, ends on a respondent anchor turn, and includes at least one interviewer and one respondent turn. The construction pipeline first segments raw utterance streams into candidate windows, applies heuristic prefiltering to retain high-quality local moments, asks an LLM to extract candidate event moments, formats candidates into a unified case schema, and then runs a separate quality-screening pass.

The final EVENT-MCQ-1000 / EVENT-QA-1000 dataset contains 1,000 cases. Each case includes a local window, an anchor quote, an event type, a recovery goal, a recovery policy, a target recover state, a gold next interviewer turn, acceptable event labels or policies, and provenance. Cases are produced in two construction modes. In extracted mode, the event naturally appears in the transcript window. Here, extracted means that the case originates from a real transcript window, passes candidate extraction and quality screening, and is not created through controlled event injection. In *llm_injected* mode, a respondent anchor turn is inserted into a real local context to express a target event and severity level. Injection is used to cover rare but important disruptions that occur too sparsely in natural extraction.

Each disruptive event case also yields three evaluation tasks. The MCQ task asks the model to identify the main interviewing challenge from four options. Distractors include an adjacent event from the same failure category, a superficially similar event from a different category, and either *no_major_event* or another distant plausible alternative. The *unconditioned QA* task gives only the local window and anchor quote, requiring the model to diagnose the event and generate the next interviewer turn end to end. The *conditioned QA* task additionally supplies the event type, recovery goal, recovery policy, and target recover state, which isolates recovery execution from diagnosis.

Dimension	Fine-grained label	MCQ n	MCQ %	QA n	QA %
Questioning	open_specific_clear	1,427	31.7	467	23.4
Questioning	open_abstract_clear	989	22.0	355	17.8
Questioning	closed_specific_clear	462	10.3	98	4.9
Questioning	open_abstract_ambiguous	147	3.3	32	1.6
Questioning	open_specific_ambiguous	76	1.7	30	1.5
Questioning	closed_abstract_clear	42	0.9	18	0.9
Questioning	closed_specific_ambiguous	21	0.5	16	0.8
Questioning	closed_abstract_ambiguous	9	0.2	4	0.2
Follow-up	overgeneral_or_narrowness	404	9.0	236	11.8
Follow-up	missing_information	347	7.7	286	14.3
Follow-up	incomplete_or_unsupported_themes	101	2.2	97	4.8
Follow-up	unclear_concept	110	2.4	63	3.1
Follow-up	new_relevant_idea	83	1.8	73	3.6
Follow-up	story_with_implicit_theme	65	1.4	56	2.8
Follow-up	evasive_or_unanswered	53	1.2	52	2.6
Follow-up	contradiction_or_ambivalence	47	1.0	36	1.8
Response	acknowledgment	87	1.9	54	2.7
Response	repeat_or_reorganize	15	0.3	17	0.8
Response	encouraging	10	0.2	8	0.4
Response	self_disclosure	3	0.1	2	0.1
Response	evaluative	2	0.0	0	0.0

Table 9: Fine-grained interviewer-label distribution in Core-MCQ-4500 and Core-QA-2000, grouped by interviewer dimension. These labels describe the intended interviewer move or local trigger, not model accuracy.

Event type	Extracted	Injected	Total
needs_elaboration	160	0	160
incomplete_or_unsupported_claim	130	0	130
contradiction_or_ambivalence	100	0	100
factual_misconception	10	90	100
hostility_or_distrust	20	80	100
passive_noncooperation	100	0	100
question_misunderstanding	67	33	100
topic_refusal	23	57	80
early_termination_request	0	70	70
sudden_logistics_request	1	44	45
no_major_event	15	0	15

Table 10: Short-dialogue event recovery case distribution. Injected cases supplement rare event types and provide controlled severity labels.

Structural statistic	Value
Cases	1,000
Unique dialogue sources	757
MCQ event-identification items	1,000
unconditioned QA items	1,000
conditioned QA items	1,000
Average local-window turns	4.7
Average anchor length	27.3 words
Average gold next-turn length	20.2 words
MCQ gold-choice distribution	A 26.1 / B 23.5 / C 25.5 / D 24.9

Table 11: Structural statistics for the static disruptive event benchmark.

B.4 Dynamic Evaluation Cases

The dynamic evaluation track is a questionnaire-grounded long-dialogue benchmark. Questionnaire construction proceeds by topic clustering and template design over the original interview transcripts. Templates allow single_choice, Likert, numeric, and text_input; single-choice fields have four options, Likert fields use a five-point scale, and long-form answers are represented as text_input with subtype long_text. Each template has 8–15 items and at most two text-input fields.

Gold answers are derived from the original interview transcripts. Each extracted questionnaire item is audited before it can enter the gold record and is assigned an answer status among *answered*, *inferable*, *unresolved*, *contradicted*, and *not_applicable*. An item is retained as formal gold only when it is marked as *answered*, has sufficient transcript support, and can be converted into a scoreable target in the normalized gold file. Depending on the item type, this target may include a canonical answer, a strict answer, acceptable aliases, normalized variants, and, for long-form text fields, a set of required semantic facets. Items whose source transcript does not support a determinate answer are either repaired through an additional audit pass when the missing annotation is recoverable, or excluded from scoring when the original source does not contain the relevant information.

Episode-level acceptance requires sufficient answered items, at least 70% structured answered items among answered items, no more than two text-input answered items, at most 30% unresolved or contradicted template items, legal option or numeric mappings for structured gold, and explicit transcript evidence quotes for at least 80% of answered items.

Participant-specification construction. For each accepted scenario, we derive a constrained participant specification from the same audited source artifacts. The normalized gold record is compiled into question-indexed answer memory, containing only formal-gold answers supported by transcript evidence. Fields without supported answers remain unavailable. A separate persona profile summarizes source-supported background clues and interactional tendencies, such as qualified versus direct phrasing, willingness to elaborate, and observable conversational habits. The construction pipeline uses distinct prompt stages for persona derivation, normalized-gold generation, and clean-case audit. Their complete templates are provided in Appendix F. Thus, the participant representation is tied to the source interview and questionnaire record, rather than being a generic free-form role-play description.

For each clean-event pair, the questionnaire, formal gold record, answer memory, and persona profile are held fixed. The event counterpart changes only the localized event plan. This pairing enables the dynamic evaluation to attribute clean-event differences to the injected interaction condition rather than to a different participant specification.

Question-content coverage. To characterize this coverage more finely, we audited all 1,710 formal-gold questionnaire questions in the 184 dynamic cases. Public-opinion-related content occurs in 720 questions (42.1%) across 120 cases (65.2%), while social-science or lived-experience content occurs in 107 questions (6.3%) across 42 cases (22.8%). The benchmark also covers health/medical and psychology/mental-wellbeing content at the case level. Appendix Table 12 provides the corresponding question- and case-level counts.

C Evaluation Implementation Details

C.1 Evaluated Models

We evaluate total 23 models as shown in Appendix Table 16.

C.2 Inference Configuration

All models are queried through an OpenAI-compatible chat-completion interface with the following evaluation-track-specific settings.

Static MCQ evaluation. We use deterministic decoding with temperature 0 to minimise sampling randomness in choice selection.

Static and dynamic QA generation. For both static QA (unconditioned and conditioned) and dynamic long-dialogue generation, we use temperature 0.2 to allow natural interviewer phrasing while keeping outputs stable. Maximum output tokens are set to 1,024 for the non-thinking configuration and 4,096 for the thinking/reasoning configuration; individual models with substantially different context limits are adjusted accordingly. For all QA evaluations, we use deepseek-v4-flash as the LLM judge with temperature set to 0 to ensure deterministic scoring.

Thinking and extended reasoning. Unless otherwise stated, the main experiments use the non-thinking or lowest-reasoning configuration available for each provider. For providers that do not expose a true no-thinking mode, we use the lowest available reasoning budget.

C.3 Evaluation Execution

Static evaluation. For MCQ tasks, each model receives the local context and four candidate actions and returns a single-choice prediction. For QA tasks (unconditioned and conditioned), models generate the next interviewer utterance given the local context and, for the conditioned variant, an explicit specification of behavior type, event type, recovery goal, and recovery policy.

Dynamic evaluation. The dynamic pipeline reads a benchmark directory containing a questionnaire, normalized gold answers, a respondent persona, and a scenario plan. The interviewer model generates an opening and then proceeds through one of four workflow protocols: *Structured*, which follows the questionnaire order and asks only questions; *Probing*, which allows follow-up questions when answers are incomplete or ambiguous; *Adaptive*, which selects among questioning, probing, and response actions at each turn; and *Agentic*, which gives the interviewer greater freedom to decide the next action, including event handling and interview termination. The respondent agent an-

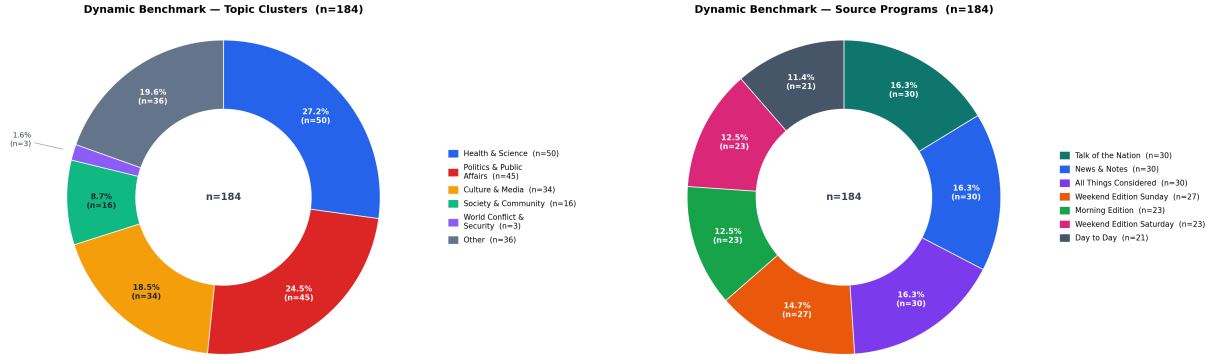


Figure 5: Composition of the dynamic evaluation benchmark. Left: topic clusters of the 184 questionnaire-grounded cases; small tail clusters are grouped into *other*. Right: source-program distribution. These plots describe data composition only and do not report model performance.

Question-content label	Questions	Share	Cases with ≥ 1 question	Case coverage
Public opinion	720	42.1%	120	65.2%
Social science / lived experience	107	6.3%	42	22.8%
Health / medical	84	4.9%	20	10.9%
Psychology / mental wellbeing	110	6.4%	60	32.6%

Table 12: Fine-grained question-content coverage among the 1,710 formal-gold questions in the 184 dynamic cases. Labels are not mutually exclusive.

Profile	Cases	Notes
Clean	184	Frozen source-grounded cases
Event	184	Matched event-conditioned cases
Total	368	Clean–event paired benchmark

Table 13: Dynamic long-dialogue benchmark size. Event cases preserve the clean questionnaire, persona, normalized gold, and ground truth while changing only the event layer.

Clean statistic	n	Notes
Selected clean cases	184	Program-capped frozen subset
Medium length	93	Based on source length bucket
Short length	73	Based on source length bucket
Long length	18	Based on source length bucket
Cases with single-choice	184	Presence count
Cases with Likert	184	Presence count
Cases with text input	184	Presence count
Cases with numeric	38	Presence count

Table 14: Clean subset statistics.

swers based on the persona and answer memory, with a planned disruptive event active in the event split. After the dialogue concludes, a questionnaire filler produces a structured answer record, and the scorer compares it against the normalized gold answers.

For each interviewer model, the number of dy-

Event type	Cases
question_misunderstanding	21
passive_noncooperation	19
factual_misconception	18
topic_derailment	18
topic_refusal	18
contradictory_information	18
hostility_or_distrust	16
sudden_logistics_request	16
memory_failure	15
early_termination_request	13
questionnaire_critique	12

Table 15: Event distribution in the event subset. The event layer is active for all 184 event cases.

namics runs is:

$$N_{\text{runs}} = (184_{\text{clean}} + 184_{\text{event}}) \times 4_{\text{workflows}} = 368 \times 4 = 1,472.$$

Thus, each interviewer model is evaluated on 1,472 multi-turn interview runs, covering 184 clean cases and 184 matched event cases under four workflow protocols.

C.4 Participant Simulation Runtime Controls

Appendix B describes how each source-grounded participant specification is constructed. At runtime, the simulator receives the resulting persona profile, answer memory relevant to the current question, question text and type, options or Likert scale when

Model	Access
Claude Sonnet 4.6	API
GPT-4o	API
GPT-5.4	API
DeepSeek-V4-Pro	API
DeepSeek-V4-Flash	API
Gemini 2.5 Pro	API
Gemini 3.1 Pro Preview	API
Qwen3-Max	API
GLM-5	API
Llama 3.1-8B	vLLM
Llama 3.1-70B	vLLM
Gemma 3-12B	vLLM
Gemma 3-27B	vLLM
Qwen2.5-1.5B	vLLM
Qwen3-1.7B	vLLM
Qwen3-8B	vLLM
Qwen3-14B	vLLM
Qwen3-32B	vLLM
Qwen3.5-2B	vLLM
Qwen3.5-4B	vLLM
Qwen3.5-122B-A10B	vLLM
GPT-OSS-20B	vLLM
GPT-OSS-120B	vLLM

Table 16: Models evaluated in this work. All models are queried through an OpenAI-compatible chat-completion interface; locally deployed models are served via vLLM.

applicable, dialogue history, and the event plan for event scenarios. It is instructed to disclose information naturally only when the interviewer’s question is relevant, to avoid referring to answer memory or gold answers, and to leave unsupported details unspecified rather than inventing them. These constraints are implemented through the interviewee-simulation prompt reproduced in Appendix F.

All 184 frozen clean cases have complete answer-memory coverage, a non-empty persona profile, persona-trait fields, and a full Big-Five trait representation. On average, each case contains 10.89 questionnaire items, 9.29 non-null answer-memory entries (85.74% of fields), 14.23 persona facts, 4.12 behavioral habits, and 4.07 language habits. These checks verify that the controlled runtime inputs are present for every case; they do not imply that the simulator is a substitute for a real respondent.

De-identified runtime example. Appendix Table 17 shows an abridged instance from a disaster-interview scenario. It illustrates that the simulator receives constrained, question-indexed evidence rather than a free-form instruction to improvise a respondent. The table omits source identifiers and sensitive details; it retains only the fields needed to show how profile, answer memory, and the current-

question runtime state interact.

Runtime prompt roles. The dynamic execution uses separate prompts for workflow-specific interviewer control, participant simulation, structured answer extraction, and respondent-satisfaction scoring. The interviewer-control prompts determine whether to ask, probe, respond, repair, or conclude according to the selected workflow; the participant prompt enforces the source-grounded disclosure constraints above. After the dialogue, the structured-answer prompt maps conversational evidence to the questionnaire schema, while the satisfaction prompt evaluates the participant experience from the full interaction trace. This separation prevents the participant simulator, interviewer, filler, and evaluator from sharing a single undifferentiated role instruction.

C.5 Static Disruptive Event Scoring

The static disruptive-event benchmark evaluates event handling at the local turn level. Event-MCQ-1000 is scored by exact match: a prediction is correct if it matches the gold choice or an explicitly listed acceptable alternative. Event-QA-1000 is scored by an LLM judge on six dimensions: event awareness (EA), policy alignment (PA), recovery execution (RE), target-state progress (TP), interview naturalness (IN), and respect or safety (RS). The composite score is:

$$\text{comp.} = 0.20 EA + 0.25 PA + 0.25 RE \\ + 0.15 TP + 0.10 IN + 0.05 RS.$$

Comparing the unconditioned and conditioned settings of Event-QA-1000 isolates the cost of diagnosis: a large conditioned-minus-unconditioned gain indicates that the model can execute recovery once the event and policy are supplied, but does not reliably infer them from dialogue context alone. Results are reported by event type, construction mode, and severity level where available.

C.6 Dynamic Respondent-Satisfaction Scoring

We further evaluate respondent satisfaction as a complementary interaction-quality metric. For each completed interview, we assess the overall respondent experience based on the full dialogue, questionnaire context, and interaction trace. The judge assigns a direct overall satisfaction score on a 0–100 scale and provides a concise rationale. To improve diagnostic resolution, the judge also rates

Input group	De-identified example passed to the simulator
Profile constraints	Reflective and measured speaking style; uses qualified language when uncertain; avoids unsupported definitive claims; maintains personal boundaries and a moderate level of emotional regulation.
Question-indexed answer memory	Q1: <i>response and adaptation measures</i> ; Q2: <i>strongly agree</i> ; Q3: <i>primarily natural</i> ; Q9: <i>\$5,000</i> . An unsupported field is stored as null and is not treated as recoverable evidence.
Current-question state	Q2 is a five-point Likert item about harm to affected communities. The runtime input includes the item text, the full response scale, the relevant Q2 answer memory, and preceding dialogue turns.
Event state	In the event counterpart, Q2 activates <code>question_misunderstanding</code> : the respondent initially interprets the item as asking about the cause of the disaster rather than the severity of community harm.
Disclosure rules	Answer naturally only when the interviewer asks a relevant question; do not mention answer memory, gold answers, or the benchmark; leave unsupported details unspecified.

Table 17: De-identified example of the controlled runtime inputs for one dynamic scenario. The profile is source-derived and abridged; answer-memory values are shown only for selected formal-gold items.

a checklist covering respect, autonomy, empathy, question clarity, active listening, follow-up relevance, repetition control, conversational naturalness, pacing, cognitive burden, coverage, recovery from disruptions, trust and safety, persona fit, closure quality, and willingness to continue. This metric captures whether the interview process would feel clear, respectful, low-burden, and trustworthy from the respondent’s perspective, especially under event settings with injected disruptions.

C.7 Dynamic Evaluation Scale and Cost

The completed dynamic evaluation contains 184 clean scenarios and 184 matched event scenarios, evaluated by four interviewer models under four workflows, for 5,888 completed runs. Appendix Table 18 reports aggregate scale, and Appendix Table 19 reports the workflow-level cost–outcome trade-off.

Structured and Probing require about 46K tokens and 2.2–2.4 minutes per run, whereas Adaptive and Agentic require 1.9–2.8 times as many tokens and substantially more wall-clock time. This cost premium does not yield higher average Direct Backfill or Spoken Truth in the aggregate results. The aggregate Adaptive cost is partly driven by model-specific control failures. Explicit adaptation and autonomous control are materially more expensive than the constrained baselines in the current implementation and should be evaluated jointly with their information-recovery benefit.

Appendix Table 20 decomposes the Agentic workflow. The recurring interviewer–interviewee exchange dominates its logged cost: response simulation, action selection, and interviewer response realization account for the largest per-run com-

Execution statistic	Value
Clean scenarios / event scenarios	184 / 184
Interviewer models / workflows	4 / 4
Completed dynamic runs	5,888
Observed dialogue turns	75,831
Mean / median turns per run	12.88 / 12
Total LLM tokens	456.95M
Total end-to-end runtime	316.81 h

Table 18: Scale of the completed dynamic evaluation. Runtime is summed over case runs and includes orchestration overhead.

ponents, while optional free-follow-up and planning calls add smaller but repeated overhead. This decomposition motivates lower-cost controllers, explicit stopping criteria, and information-gain-aware follow-up policies: an autonomous workflow should request another turn only when the expected improvement in answer sufficiency justifies the additional interaction and inference cost.

D Additional Experimental Results

D.1 Item Calibration Before Final Trimming

The full construction source contains 8,138 MCQ items. After requiring all six models to have `status=ok`, 8,115 jointly valid items remain. Among these, 3,132 items are ceiling items answered correctly by all six models, and 368 items are floor items answered incorrectly by all six models. Removing both leaves 4,615 mixed-difficulty candidates; the released CORE-MCQ-4500 subset trims this candidate set to 4,500 items while preserving small target-type categories. Unless otherwise specified, CORE-MCQ-4500 refers to the calibrated ordinary local-decision MCQ subset covering *Questioning*, *Follow-up*, and *Response*.

Workflow	Runs	Turns	Tokens/run	Runtime/run (s)	Direct Backfill	Spoken Truth
Structured	1,472	10.89	45.7K	133.5	73.82	53.53
Probing	1,472	11.03	46.5K	141.3	74.25	53.22
Adaptive	1,472	15.62	129.9K	277.0	72.16	43.85
Agentic	1,472	13.98	88.3K	222.9	71.54	39.94

Table 19: Workflow-level dynamic evaluation cost and outcome summary. Tokens and runtime are per completed run, averaged over four interviewer models and clean/event conditions.

Component	Tokens/run	LLM latency/run (s)	Calls/run
Interviewee response	29.4K	47.3	13.79
Interviewer action selection	23.7K	45.7	12.01
Interviewer response realization	19.9K	46.2	11.82
Final profiler	5.3K	30.0	1.00
Questionnaire filler	4.5K	27.7	1.00
Free follow-up controller	3.2K	8.2	1.98
Interviewer plan	2.4K	10.7	1.32
Questionnaire summarizer	1.5K	10.5	1.00
Interview opening	0.5K	5.9	1.00

Table 20: Component-level cost of the Agentic workflow. Values are mean logged usage over Agentic runs. Component means are reported separately because optional planning and free-follow-up calls have different invocation rates.

It excludes EVENT-MCQ-1000 and all dynamic long-dialogue evaluations.

Calibration category	n	Share	Interpretation
Source items	8,138	–	Full MCQ construction source
Jointly valid items	8,115	100.0	Six models have OK outputs
Ceiling items	3,132	38.6	All six models correct
Floor items	368	4.5	All six models wrong
Mixed-difficulty candidates	4,615	56.9	Retained before final trimming
Final release	4,500	–	Proportionally trimmed benchmark

Table 21: Item calibration before Core-MCQ-4500 trimming. Shares are computed over the 8,115 jointly valid items.

Model	Source valid	Mixed candidates	Final 4,500
Claude Sonnet 4.6	86.53	84.27	84.31
Gemini 2.5 Pro	79.95	72.65	72.64
DeepSeek V4 Pro	75.97	65.68	65.80
DeepSeek V4 Flash	69.50	54.30	54.09
GPT-4o	65.39	47.06	47.02
Qwen3-Max	57.78	33.65	33.67

Table 22: Model accuracy (%) before and after removing ceiling/floor items, then after final trimming. Calibration lowers accuracy most for models that rely heavily on easy items.

D.2 MCQ Target-Type Results

Appendix Table 23 reports selected target-type results. The strongest pattern is that response-like categories such as acknowledgment are much harder than clear question-generation categories.

Target type	n	Share	Avg. acc.
open_specific_clear	1,427	31.71	63.41
open_abstract_clear	989	21.98	61.65
closed_specific_clear	462	10.27	56.40
overgeneral_or_narrowness	404	8.98	60.93
missing_information	347	7.71	59.41
acknowledgment	87	1.93	39.57
repeat_or_reorganize	15	0.33	50.48
closed_abstract_ambiguous	9	0.20	42.86

Table 23: Selected MCQ target-type results from final Core-MCQ-4500.

D.3 QA Fine-Grained Categories

Conditioned QA is easiest for clear follow-up categories and hardest for ambiguous question categories. This supports the interpretation that current models are fluent but still benefit strongly from explicit technique labels.

D.4 Detailed Static Local-Decision Results

Appendix Tables 25 and 26 report the detailed Core-QA-2000 results under the unconditioned and conditioned settings. Appendix Tables 27 and 28 report the corresponding Event-QA-1000 results under the same two settings.

Difficulty	Target type	n	Comp.
Hard	open_specific_ambiguous	180	69.30
Hard	closed_specific_ambiguous	96	71.45
Hard	open_abstract_ambiguous	192	79.81
Hard	acknowledgment	324	82.59
Easy	unclear_concept	378	94.39
Easy	contradiction_or_ambivalence	216	94.24
Easy	missing_information	1,716	93.49
Easy	new_relevant_idea	438	92.83

Table 24: Selected hard and easy categories in conditioned QA. Counts are model-item rows.

D.5 Static Diagnostic Figure

Appendix Figure 6 summarizes four static-evaluation diagnostics: overall MCQ performance, dimension-level variation, QA conditioned scores, and the relation between recognition and generation. These plots complement the detailed tables above by showing that static interviewer performance varies across both task format and interviewer decision type.

D.6 Static Correlation and Judge Sensitivity

Appendix Table 29 summarizes the rank-level association between MCQ recognition and QA generation. The strongest relationship is between MCQ and conditioned QA, where the target strategy is supplied to the model. This supports the interpretation that MCQ recognition is closer to strategy selection, while unconditioned QA is also shaped by open-ended generation style.

The high DeepSeek–Qwen agreement for conditioned QA suggests that conditioned generation conclusions are stable across judges. Unconditioned QA agreement is lower, which is expected because unconditioned QA leaves the target interviewing strategy underspecified.

D.7 Human Audit of Static Labels and QA Judges

We conduct a stratified human audit of both static benchmark labels and LLM-judge scores. The audit covers 260 sampled instances and 340 completed audit records: 80 core label cases, 80 event-recovery label cases, 50 Core-QA-2000 generations for trigger and technique scoring (37 for language quality after validity filtering), and 50 Event-QA-1000 generations. Appendix Table 30 reports the results.

The human audit supports two bounded conclusions. First, the automatic construction labels are highly consistent with the human audit: deci-

sion dimensions, target types, MCQ answer keys, event types, and recovery policies all have Cohen’s $\kappa \geq 0.895$. Second, DeepSeek-V4-Flash QA scores are strongly aligned with human ratings for trigger, technique, event awareness, and policy alignment. Language quality is less stable but remains positively associated. These results validate the use of automatic evaluation for aggregate comparison, while not implying that an LLM judge replaces human evaluation in every ambiguous instance.

D.8 MCQ Error Analysis

Appendix Table 31 reports high-error target types aggregated over model–item decisions. The strongest pattern is that response-like moves, especially acknowledgment and repeat_or_reorganize, produce the highest error rates. This indicates that models often confuse polite or fluent continuation with functionally appropriate interviewer response behavior.

Option-level distractor-role patterns are analyzed in Appendix E. Those percentages are computed only for wrong selections whose chosen option can be linked to recorded construction metadata.

D.9 Gemini Event-Diagnosis Errors

Gemini-3.1-Pro performs well on ordinary static interviewer decisions but under-detects subtle disruptive events in the short-dialogue event MCQ. Among 999 valid event MCQ items, Gemini answers 302 incorrectly. Of these wrong answers, 247 select an option whose meaning is that no major event is present, the exchange proceeds normally, or the continuation is smooth. Appendix Table 32 shows the gold labels most often hidden by this false-normal prediction. The dominant cases are not severe interruptions; they are subtle local needs such as elaboration, unsupported claims, ambivalence, and misunderstanding. This supports the interpretation that Gemini’s weakness is primarily event diagnosis.

D.10 Detailed Dynamic Long-Dialogue Results

The dynamic long-dialogue track uses the frozen clean–event benchmark described in Appendix Table 13: 184 clean cases and 184 matched event cases. The full model–workflow main table is reported once in Table 4; this appendix does not repeat that table. Instead, it reports chain-level

Model	MCQ Recognition				Core-QA-2000 — Unconditioned Setting			
	Overall	Quest.	Follow	Resp.	Comp.	Trigger	Tech.	Lang.
<i>Proprietary models</i>								
Claude Sonnet 4.6	84.3	82.6	87.9	94.0	70.2	71.7	70.2	89.2
Gemini-3.1-Pro	89.0	86.6	94.5	94.9	78.1	73.8	71.3	89.3
Gemini-2.5-Pro	72.6	72.3	75.2	55.6	69.5	69.6	65.8	89.6
GPT-5.4	65.8	69.6	57.1	52.1	78.6	74.7	72.1	89.1
GPT-4o	47.0	52.2	36.8	12.8	69.1	69.2	64.2	88.0
<i>Open-source models</i>								
DeepSeek-V4-Pro	65.8	68.3	61.2	46.2	69.9	65.9	64.8	89.3
DeepSeek-V4-Flash	54.1	55.3	51.7	45.3	69.1	68.2	64.9	88.8
Gemma-3-12B	39.5	44.4	29.0	16.2	70.5	62.7	59.0	89.8
Gemma-3-27B	44.2	46.2	40.7	24.8	71.3	64.1	60.2	89.5
GLM-5	70.4	70.9	70.3	55.6	70.4	62.2	61.5	87.5
LLaMA-3.1-8B	30.8	32.2	32.4	21.1	69.7	62.3	58.4	88.4
LLaMA-3.1-70B	47.5	49.3	45.5	20.5	70.9	64.5	59.8	88.4
Qwen2.5-1.5B	27.1	26.8	30.1	11.2	61.0	52.9	48.4	81.8
Qwen3-1.7B	27.3	31.3	18.7	7.7	61.1	55.1	48.6	79.5
Qwen3-8B	36.1	37.9	34.0	8.5	69.9	62.5	60.3	86.7
Qwen3-14B	37.2	38.2	36.9	12.0	71.6	66.0	60.0	88.8
Qwen3-32B	44.6	47.1	41.4	10.3	71.8	66.5	60.1	88.7
Qwen3.5-2B	27.2	30.8	20.5	6.1	66.3	56.8	57.7	84.3
Qwen3.5-4B	45.8	46.3	47.8	11.1	70.3	62.3	62.2	86.5
Qwen3.5-122B-A10B	63.7	63.6	66.5	35.9	70.6	63.1	61.5	87.2

Table 25: Core-MCQ-4500 recognition and Core-QA-2000 results under the unconditioned setting for all 23 models. MCQ Overall is the weighted average over 3,173 questioning, 1,210 follow-up, and 117 response items (4,500 total). Core-QA-2000 is judged by DeepSeek-V4-Flash. **Comp.** is the composite score; **Trigger** and **Tech.** measure trigger-type and technique-type match; **Lang.** is language quality. Best score in each column is bolded.

Model	MCQ Recognition				Core-QA-2000 — Conditioned Setting			
	Overall	Quest.	Follow	Resp.	Comp.	Trigger	Tech.	Lang.
<i>Proprietary models</i>								
Claude Sonnet 4.6	84.3	82.6	87.9	94.0	89.2	92.0	85.6	90.0
Gemini-3.1-Pro	89.0	86.6	94.5	94.9	90.2	92.9	87.8	89.9
Gemini-2.5-Pro	72.6	72.3	75.2	55.6	89.0	90.5	86.1	90.5
GPT-5.4	65.8	69.6	57.1	52.1	89.2	90.4	87.1	90.2
GPT-4o	47.0	52.2	36.8	12.8	88.0	89.2	85.3	89.5
<i>Open-source models</i>								
DeepSeek-V4-Pro	65.8	68.3	61.2	46.2	87.8	89.8	84.0	89.6
DeepSeek-V4-Flash	54.1	55.3	51.7	45.3	87.6	89.0	84.6	89.3
Gemma-3-12B	39.5	44.4	29.0	16.2	82.4	81.2	76.0	89.9
Gemma-3-27B	44.2	46.2	40.7	24.8	83.4	83.3	77.0	90.0
GLM-5	70.4	70.9	70.3	55.6	87.4	87.9	84.8	89.4
LLaMA-3.1-8B	30.8	32.2	32.4	21.1	79.7	78.2	73.4	87.5
LLaMA-3.1-70B	47.5	49.3	45.5	20.5	82.5	83.0	77.2	87.4
Qwen2.5-1.5B	27.1	26.8	30.1	11.2	69.0	66.2	58.6	82.2
Qwen3-1.7B	27.3	31.3	18.7	7.7	75.2	74.4	69.7	81.3
Qwen3-8B	36.1	37.9	34.0	8.5	80.7	79.4	77.5	85.4
Qwen3-14B	37.2	38.2	36.9	12.0	81.6	81.9	75.9	86.9
Qwen3-32B	44.6	47.1	41.4	10.3	81.5	81.8	75.5	87.3
Qwen3.5-2B	27.2	30.8	20.5	6.1	75.4	71.7	71.2	83.4
Qwen3.5-4B	45.8	46.3	47.8	11.1	81.2	79.3	78.4	86.0
Qwen3.5-122B-A10B	63.7	63.6	66.5	35.9	85.0	85.1	82.6	87.4

Table 26: Core-MCQ-4500 recognition and Core-QA-2000 results under the conditioned setting for all 23 models. The conditioned setting supplies the expected action type and technique label before generation. The unconditioned-to-conditioned composite gap quantifies the difficulty attributable to technique inference. Core-QA-2000 is judged by DeepSeek-V4-Flash.

Model	MCQ	Event-QA-1000 — Unconditioned Setting				
	Acc.	Comp.	Evt-Aware	Policy	Recovery	Natural.
<i>Proprietary models</i>						
Claude Sonnet 4.6	86.4	81.4	87.6	79.8	79.0	87.1
Gemini-3.1-Pro	69.8	83.1	88.8	82.0	80.9	86.9
Gemini-2.5-Pro	75.6	72.3	79.6	70.5	68.7	81.6
GPT-5.4	84.4	81.8	87.6	80.3	79.4	87.5
GPT-4o	84.0	74.8	81.3	72.6	71.7	84.0
<i>Open-source models</i>						
DeepSeek-V4-Pro	78.8	82.2	88.0	81.3	79.9	86.6
DeepSeek-V4-Flash	76.1	82.0	88.7	81.0	79.5	86.7
Gemma-3-12B	72.7	64.8	73.6	62.6	59.9	78.9
Gemma-3-27B	73.5	66.3	74.3	64.1	61.5	80.3
GLM-5	87.4	76.8	83.2	74.7	73.6	85.0
LLaMA-3.1-8B	47.4	62.0	73.3	58.9	57.5	76.2
LLaMA-3.1-70B	71.0	70.2	77.8	67.7	66.6	81.2
Qwen2.5-1.5B	49.0	57.3	66.9	53.7	53.3	70.3
Qwen3-1.7B	34.4	50.4	58.4	46.7	45.5	66.8
Qwen3-8B	75.6	68.0	75.1	65.5	64.1	80.2
Qwen3-14B	82.4	72.3	80.1	69.9	68.5	82.7
Qwen3-32B	81.6	73.0	79.4	70.7	69.9	83.3
Qwen3.5-2B	54.1	49.2	59.7	45.8	44.1	63.6
Qwen3.5-4B	81.9	63.5	72.3	61.2	59.2	76.0
Qwen3.5-122B-A10B	82.8	79.3	86.3	78.1	76.6	84.9

Table 27: Event-QA-1000 results under the unconditioned setting. MCQ accuracy measures event-type recognition. The unconditioned setting asks the model to simultaneously diagnose the disruption and produce a recovery turn without explicit event-type guidance. All QA scores judged by DeepSeek-V4-Flash: **Evt-Aware** = event awareness; **Policy** = recovery policy alignment; **Recovery** = recovery execution quality; **Natural.** = interview naturalness.

Model	MCQ	Event-QA-1000 — Conditioned Setting				
	Acc.	Comp.	Evt-Aware	Policy	Recovery	Natural.
<i>Proprietary models</i>						
Claude Sonnet 4.6	86.4	88.9	93.0	88.4	87.8	89.2
Gemini-3.1-Pro	69.8	89.1	92.9	89.2	88.1	87.9
Gemini-2.5-Pro	75.6	72.2	76.6	71.7	70.2	74.7
GPT-5.4	84.4	89.2	93.2	89.0	88.1	88.8
GPT-4o	84.0	86.3	90.7	86.1	84.8	86.9
<i>Open-source models</i>						
DeepSeek-V4-Pro	78.8	88.6	92.3	89.0	87.6	87.7
DeepSeek-V4-Flash	76.1	88.7	92.8	89.2	87.7	87.3
Gemma-3-12B	72.7	83.0	88.4	82.9	81.0	85.0
Gemma-3-27B	73.5	83.2	88.5	82.9	81.3	85.4
GLM-5	87.4	87.8	91.5	87.8	86.5	87.6
LLaMA-3.1-8B	47.4	74.8	82.1	74.1	72.0	79.3
LLaMA-3.1-70B	71.0	84.2	89.4	84.5	82.7	84.5
Qwen2.5-1.5B	49.0	63.3	72.6	61.0	59.6	70.2
Qwen3-1.7B	34.4	60.8	69.2	58.8	56.9	67.7
Qwen3-8B	75.6	77.3	83.9	76.5	74.8	82.4
Qwen3-14B	82.4	82.0	87.7	81.4	80.1	84.9
Qwen3-32B	81.6	83.9	88.9	83.1	82.2	86.3
Qwen3.5-2B	54.1	63.6	72.9	61.6	59.8	70.7
Qwen3.5-4B	81.9	77.2	84.0	76.7	74.7	80.6
Qwen3.5-122B-A10B	82.8	86.5	91.3	86.2	85.2	86.1

Table 28: Event-QA-1000 results under the conditioned setting. The conditioned setting provides the event type, recovery goal, and recovery policy before generation. The unconditioned-to-conditioned composite gap measures how much recovery performance depends on explicit event-type guidance versus unsupported diagnosis. All QA scores judged by DeepSeek-V4-Flash.

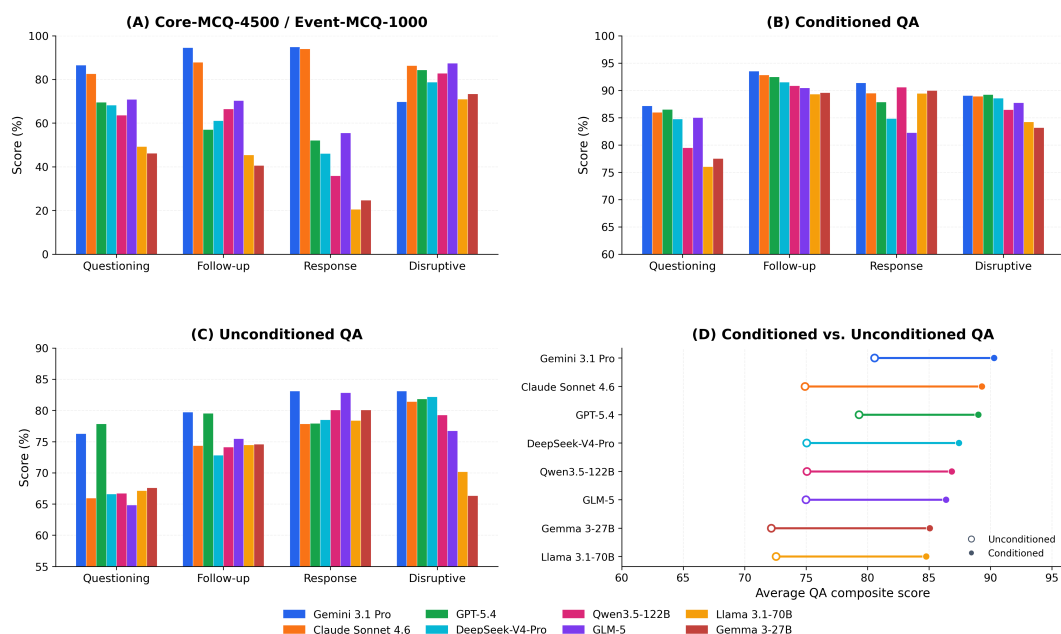


Figure 6: Integrated static-evaluation diagnostics. The four panels summarize MCQ recognition, conditioned and unconditioned QA performance by dimension, and the gain from supplying the target strategy.

Metric pair	Judge	Spearman ρ
MCQ overall vs. conditioned QA	DeepSeek	0.821
MCQ overall vs. unconditioned QA	DeepSeek	0.329
MCQ overall vs. conditioned QA	Qwen	0.754
MCQ overall vs. unconditioned QA	Qwen	0.717
Conditioned QA, DeepSeek vs. – Qwen		0.956
Unconditioned QA, DeepSeek vs. – Qwen		0.563

Table 29: Rank-level correlations among static MCQ and QA metrics across 23 models.

diagnostics that separate question reach, elicited evidence, backfilling, and interaction quality.

D.11 Dynamic Chain Diagnostics

The dynamic analysis covers 64,096 target-question instances across four interviewer models, four workflows, and clean/event settings. Appendix Table 33 separates question reach, explicit information elicitation, final backfilling, efficiency, and respondent satisfaction. Completion is nearly saturated across workflows, whereas elicitation and efficiency vary substantially. Thus, the central challenge is not merely finishing a questionnaire, but obtaining information that is sufficiently clear and specific to support a reliable final record.

Appendix Table 34 compares clean and event runs. The event split has almost identical coverage but lower efficiency and substantially lower satisfaction. This supports the interpretation that

disruptive events primarily affect interaction quality, repair cost, and evidence quality.

D.12 Human Audit of Dynamic Simulation and Satisfaction

We separately audit a stratified subset of 104 completed dynamic runs spanning two interviewer models, four workflows, and clean/event settings. A human annotator rated overall respondent satisfaction on a 0–100 scale and, for 103 runs with complete auxiliary fields, rated emotional authenticity, memory consistency, and human-likeness of the simulated respondent on 10-point scales. Appendix Table 35 shows that the simulated respondents pass this targeted plausibility check, with average ratings above 8.6 on all three dimensions.

The same audit also calibrates the automatic satisfaction judge. Across the 104 paired ratings, human and LLM satisfaction have Pearson $r = 0.596$ and Spearman $\rho = 0.524$ (Appendix Table 36). Agreement is closer for clean cases (human 95.25, LLM 96.98) than for event cases (human 92.05, LLM 79.38). Thus, the LLM judge is not uniformly lenient: it is more punitive than the human rater in the event condition. We use satisfaction primarily as a comparative signal and retain this calibration gap as a limitation of the current automated evaluation.

Appendix Table 37 gives representative model-workflow clean–event deltas. The largest and most

Audit component	Field	n	Agreement	Cohen’s κ	MAE
Core static labels	Decision dimension	80	96.25	0.9412	–
Core static labels	Target type	80	90.00	0.8949	–
Core static labels	MCQ answer key	80	95.00	0.9330	–
Event-recovery labels	Event type	80	100.00	1.0000	–
Event-recovery labels	Recovery policy	80	98.75	0.9861	–
Core QA judge	Trigger-type match	50	$\rho = 0.9554$	$r = 0.9702$	6.20
Core QA judge	Technique-type match	50	$\rho = 0.9669$	$r = 0.9706$	5.40
Core QA judge	Language quality	37	$\rho = 0.7694$	$r = 0.7669$	4.04
Event-QA judge	Event awareness	50	$\rho = 0.9595$	$r = 0.9794$	2.40
Event-QA judge	Policy alignment	50	$\rho = 0.9663$	$r = 0.9731$	7.70
Event-QA judge	Interview naturalness	50	$\rho = 0.8715$	$r = 0.9387$	2.68

Table 30: Human audit of static labels and LLM-judge scores. Label rows report human–pipeline agreement; QA rows report Spearman ρ , Pearson r , and mean absolute error (MAE) between human and DeepSeek-V4-Flash scores. The audit contains 340 completed annotations across stratified label and QA subsets.

Target type	Decisions	Errors	Error rate
acknowledgment	1,997	1,442	72.2
repeat_or_reorganize	345	245	71.0
closed_abstract_ambiguous	206	137	66.5
encouraging	228	143	62.7
closed_abstract_clear	965	578	59.9
story_with_implicit_theme	1,492	882	59.1
evaluative	46	27	58.7
closed_specific_ambiguous	483	281	58.2
new_relevant_idea	1,905	1,096	57.5
unclear_concept	2,522	1,433	56.8

Table 31: Selected high-error MCQ target types. Counts are model–item decisions.

Gold event type when Gemini chose “no Wrong event”	tions	selec-
needs_elaboration	74	
contradiction_or_ambivalence	57	
incomplete_or_unsupported_claim	57	
question_misunderstanding	24	
factual_misconception	17	
passive_noncooperation	9	
hostility_or_distrust	8	
topic_refusal	1	

Table 32: Gemini-3.1-Pro false-normal event errors in short-dialogue event MCQ. Counts are wrong items where the selected option describes no major disruption or normal dialogue flow.

Workflow	Coverage	Elicit.	Backfill	Eff./100	Sat.
Adaptive	99.89	42.99	71.57	29.95	88.82
Agentic	99.79	39.02	70.93	30.39	86.18
Probing	100.00	52.62	73.35	52.62	88.15
Structured	100.00	52.37	73.84	51.69	90.30

Table 33: Workflow-level dynamic diagnostics. Coverage is question reach; Elicit. is spoken-truth success; Backfill is direct lenient backfill; Eff./100 is completed effective tasks per 100 turns; Sat. is respondent satisfaction.

Split	Coverage	Elicit.	Backfill	Eff./100	Sat.
Clean	99.94	47.34	72.89	40.64	92.81
Event	99.90	46.15	71.96	38.42	83.92

Table 34: Clean versus event dynamic diagnostics. The largest gap is satisfaction, followed by efficiency.

Simulator plausibility dimension	n	Mean	SD
Emotional authenticity (1–10)	103	8.88	0.32
Memory consistency (1–10)	103	8.89	0.34
Human-likeness (1–10)	103	8.61	0.53

Table 35: Human plausibility audit of simulated respondents. One sampled record did not contain the three auxiliary plausibility ratings.

Split	n	Human	LLM	MAE	ρ
All	104	93.53	87.50	11.51	0.524
Clean	48	95.25	96.98	6.02	–
Event	56	92.05	79.38	16.21	–

Table 36: Human–LLM satisfaction audit. The overall paired-score Pearson correlation is $r = 0.596$; the LLM judge is notably stricter in event cases.

consistent degradation is satisfaction, whereas completion remains almost unchanged. Appendix Figure 7 visualizes the same pattern across model–workflow pairs. Together, they support the main-text claim that events primarily affect interaction cost and respondent experience.

D.13 Performance Analysis by Question Type

Question type changes where dynamic-interview failures occur. Appendix Table 38 and Appendix Figure 8 use the truth-only diagnostic, excluding fields without scoreable gold answers. Single-choice and Likert fields have high direct backfill

Model	Workflow	Δ CCR	Δ STR	Δ DBL	Δ SAT
DeepSeek-V4-Pro	Action-classified	-0.55	-0.95	-1.23	-12.25
DeepSeek-V4-Pro	Agentic	0.00	-2.60	0.01	-9.20
DeepSeek-V4-Pro	Straight	0.00	-1.40	-0.66	-7.07
DeepSeek-V4-Pro	Simple	0.00	-0.45	-1.04	-11.08
Gemini-3.1-Pro	Action-classified	0.00	-2.55	-0.64	-8.90
Gemini-3.1-Pro	Agentic	0.00	0.10	-0.78	-10.25
Gemini-3.1-Pro	Straight	0.00	-0.90	-0.39	-6.89
Gemini-3.1-Pro	Simple	0.00	-0.35	-1.64	-5.33
Gemma-3-27B	Action-classified	0.00	-2.15	-2.10	-12.49
Gemma-3-27B	Agentic	-0.15	-1.10	-1.19	-10.31
Gemma-3-27B	Straight	0.00	-1.20	-1.49	-7.32
Gemma-3-27B	Simple	0.00	0.40	1.00	-9.80
Qwen3.5-122B	Action-classified	0.00	-2.55	-1.02	-7.37
Qwen3.5-122B	Agentic	0.00	0.80	-0.87	-7.39
Qwen3.5-122B	Straight	0.00	-2.20	-1.75	-8.56
Qwen3.5-122B	Simple	0.00	-1.95	-1.24	-8.15

Table 37: Clean–event deltas for dynamic interviews. Values are event minus clean; negative values indicate degradation under event cases.

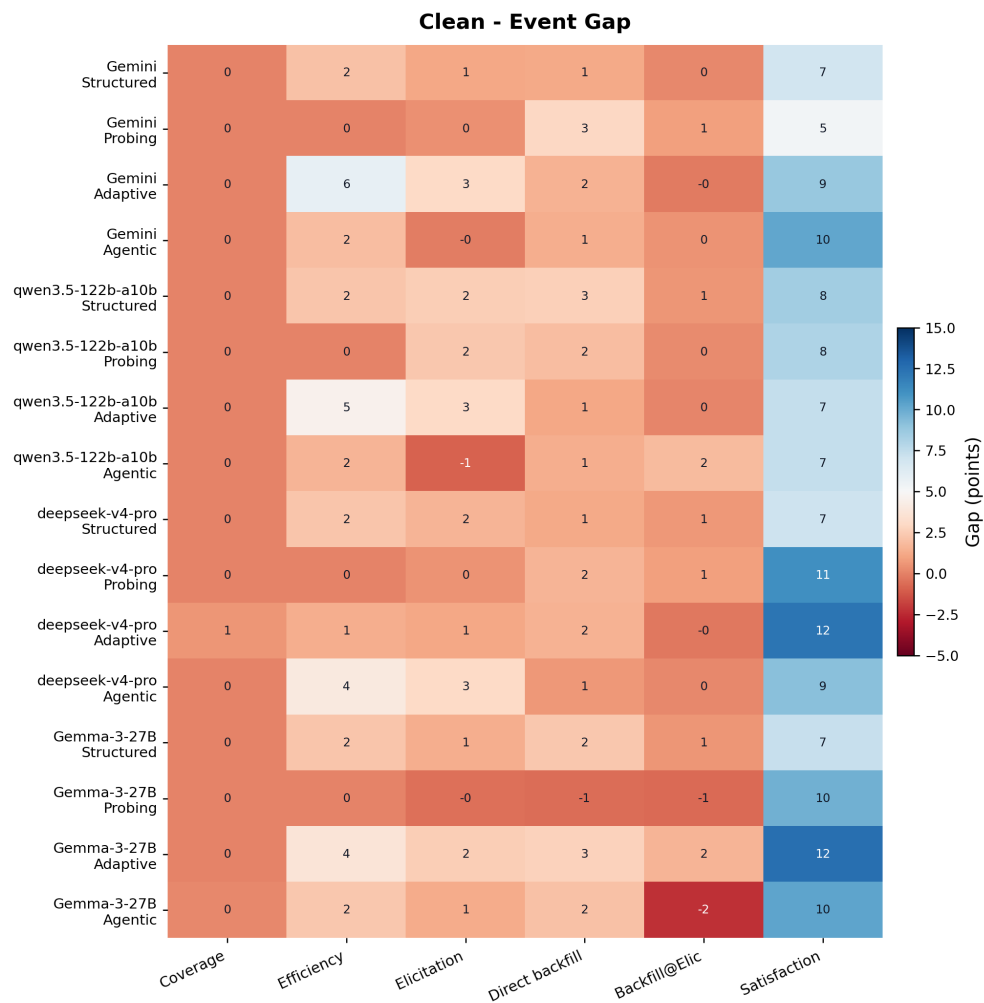


Figure 7: Clean–event gaps in dynamic long-dialogue interviews by model and workflow. Negative values indicate degradation under event cases. This supplementary heatmap shows that disruptive events mainly reduce efficiency, elicitation, and satisfaction.

scores (94.37% and 84.91%), while text-input remains much harder (57.01%). Numeric fields are easier to backfill once unscorable no-gold fields are excluded (70.76%), but their elicitation remains very low. The elicitation side tells the main story: numeric fields have only 17.93% elicitation success, and text-input fields have only 12.90%, compared with 62.76% for single-choice and 71.94% for Likert. Single-choice questions have a closed option space, so the filler can often map paraphrased answers back to the correct option. Likert questions also have a structured scale, but they still require the respondent to reveal an intensity or stance that can be aligned to the scale. Numeric and text-input questions are harder because natural interview answers often provide approximate, distributed, or explanatory evidence rather than a directly scoreable value. This is why we report spoken-truth and backfilling metrics separately.

Question type	n	Evidence	Spoken.	Backfill
Likert	15,872	71.94	71.94	84.91
numeric	1,344	17.93	17.93	70.76
single-choice	27,008	62.76	62.76	94.37
text-input	10,496	17.09	12.90	57.01

Table 38: Truth-only dynamic diagnostics by question type. Evidence is the rate at which gold information appears in dialogue; Spoken. is spoken-truth success; Backfill is direct backfill.

D.14 Respondent Satisfaction under Disruptive Events

The satisfaction judge evaluates the interview from the respondent's point of view instead of directly grading questionnaire correctness. Compared with clean interviews, event interviews reduce average satisfaction from 92.81 to 83.92 and reduce willingness to continue from 98.10% to 87.26%. Since each event case contains one controlled disruption, this drop indicates that the event layer primarily increases interaction burden and tests whether the interviewer can recognize, acknowledge, repair, and close the interaction appropriately.

Appendix Table 40 shows that the largest drops occur in burden, trust, efficiency, naturalness, focus, and clarity. Recovery increases because clean interviews contain few events to recover from; in event cases, the interviewer has an explicit opportunity to demonstrate recovery behavior. The negative dimensions therefore capture the respondent-side cost of disruption, while the positive recovery score

Workflow	Clean	Event	Event-clean
Agentic	90.82	81.53	-9.29
Straight	94.03	86.57	-7.46
Simple	92.45	83.86	-8.59
Action-classified	93.95	83.70	-10.25

Table 39: Respondent satisfaction by dynamic workflow. All workflows lose satisfaction under event cases, with the largest drop for action-classification.

indicates that the judge recognized some recovery attempts.

Dimension	Clean	Event	Diff.	Interpretation
burden	95.60	87.18	-8.43	Higher cognitive and interaction burden
trust	95.37	87.31	-8.06	Lower perceived safety and reliability
efficiency	93.21	85.75	-7.46	More repetition or delayed repair
naturalness	88.12	81.82	-6.30	Less fluent conversational flow
focus	95.84	90.07	-5.77	More derailment from task goals
clarity	96.40	90.73	-5.67	More need for clarification
respect	96.86	92.31	-4.55	Autonomy pressure in difficult cases
coverage	95.98	92.45	-3.52	Fewer fully useful answers
empathy	85.97	84.20	-1.77	Basic empathy mostly preserved
recovery	73.91	79.92	+6.01	More explicit recovery opportunities

Table 40: Satisfaction dimension changes from clean to event interviews. Diff. is event minus clean.

The strongest drops are willingness to continue, closure quality, trust and safety, efficiency, persona fit, naturalness, repetition control, pacing, specificity, and clarity. These are interaction-quality dimensions, which supports the interpretation that event injection changes the respondent experience even when coarse questionnaire completion remains high.

Event types differ substantially in their effect on satisfaction. Appendix Table 42 shows that early termination requests and sudden logistics requests are the most damaging. Both challenge respondent autonomy and procedural safety: if the interviewer continues the questionnaire without acknowledging the interviewee's request to stop, pause, or change the procedure, trust and willingness to continue decrease. Memory failure and passive non-cooperation are less damaging because they can often be repaired by gentle restatement, reduced pressure, or allowing the respondent to skip.

ID	Checklist item	Clean	Event	Diff.
C24	willingness_to_continue	4.80	4.29	-0.51
C23	closure_quality	4.33	3.89	-0.44
C21	trust_and_safety	4.87	4.46	-0.41
C16	efficiency	4.75	4.38	-0.37
C22	persona_fit	4.78	4.46	-0.32
C13	conversational_naturalness	4.45	4.14	-0.31
C11	repetition_control	4.78	4.48	-0.31
C14	pacing	4.84	4.54	-0.30
C06	question_specificity	4.85	4.56	-0.29
C05	question_clarity	4.92	4.63	-0.29

Table 41: Checklist items with the largest satisfaction drops under event interviews. Scores use a 1–5 scale.

Event type	Event satisfaction	Runs
early_termination_request	66.11	208
sudden_logistics_request	72.53	256
hostility_or_distrust	79.07	256
questionnaire_critique	81.69	192
contradictory_information	84.20	288
topic_derailment	86.29	288
factual_misconception	86.78	288
question_misunderstanding	86.95	336
topic_refusal	88.47	288
passive_noncooperation	91.50	304
memory_failure	92.48	240

Table 42: Event-interview satisfaction by planned event type. Lower scores indicate more damaging respondent-side disruption.

D.15 Cross-Fill Backfilling Analysis

Cross-fill experiments hold the dialogue source fixed and replace the model used for questionnaire backfilling. Appendix Table 43 shows the truth-only 4-by-4 matrix, including native self-filling diagonals. The dialogue source matters more than the filler identity: Gemini-generated dialogues remain easiest to backfill across different fillers, while Gemma-generated dialogues remain harder. This supports the interpretation that dynamic failures are driven by both the interview evidence quality and the final record-filling model.

D.16 Paired Dynamic Event-Type Impact

We pair every event run with the clean run sharing the same base case, interviewer, and workflow. The paired analysis covers four interviewer models, four workflows, and 184 base cases, yielding 2,944 clean–event pairs. Each event case contains one planned disruption. The planned event and target question are aligned with the corresponding event dialogue in all 2,944 pairs. This confirms that the controlled event is presented at the intended question, while the interviewer controls only the

subsequent recovery.

Appendix Table 44 reports event-minus-clean differences at the event target question. A negative evidence or final-fill difference indicates that the event reduces information recovery; a positive turn difference indicates additional recovery cost. Because the 16 model–workflow combinations reuse the same base cases, these are descriptive averages over repeated controlled runs rather than estimates from 2,944 independent respondents.

The global clean–event gap understates these local effects: spoken evidence changes by -0.012 when averaged over full questionnaires but by -0.098 at the event target question, while final fill changes by -0.009 globally and -0.061 at the target. Questionnaire critique, passive non-cooperation, and question misunderstanding most consistently reduce immediately spoken evidence. Contradictory information has the largest final-fill loss, consistent with conflict contaminating downstream record reconstruction. Memory failure and topic refusal impose the largest extra-turn costs, showing that continued interaction does not necessarily restore a scoreable answer. Conversely, factual misconception and early termination request do not yield a uniform negative average in this controlled sample.

D.17 Non-NPR Construction-and-Execution Compatibility Check

The main benchmark is deliberately based on a single, well-controlled media-interview source. To test whether the dynamic pipeline can nevertheless be instantiated in different interview settings, we conduct a small clean-only compatibility check on two public, non-NPR sources: ten sampled transcripts from the Developing Writers Interview Transcripts collection ([Sweetland Center for Writing, 2019](#)) and ten anonymized parent interviews about barriers and facilitators of childhood influenza vaccination ([Price, 2021](#)). The former contains reflective, open-ended accounts of undergraduate writing development; the latter concerns parental views of childhood influenza vaccination, including perceived risk, vaccination decisions, and access to vaccination.

For each source, we construct participants and apply the same four interview workflows and scoring approach without source-specific adjustments, producing 80 clean runs. Appendix Table 45 reports the resulting semantic audit. Conversation Evidence measures weighted coverage of gold-

Dialogue source \ Backfiller	Qwen	Gemma	DeepSeek	Gemini
Qwen3.5-122B	86.32	85.50	85.38	85.38
Gemma-3-27B	85.09	78.89	84.44	84.21
DeepSeek-V4-Pro	85.79	85.91	85.67	85.67
Gemini-3.1-Pro	88.71	88.19	88.13	88.71

Table 43: Truth-only cross-fill matrix of lenient backfill success. Rows fix the dialogue source model and columns change the questionnaire backfiller; diagonal cells are native self-filling settings. Best score in each row is bolded.

Event type	Base cases	Pairs	Δ spoken evidence	Δ final fill	Δ turns	Negative settings (/16)
Questionnaire critique	12	192	-0.208	-0.094	+0.464	14
Passive non-cooperation	19	304	-0.207	-0.003	+0.740	11
Question misunderstanding	21	336	-0.205	-0.119	+0.292	14
Topic refusal	18	288	-0.121	-0.035	+0.965	15
Contradictory information	18	288	-0.094	-0.250	+0.319	13
Hostility or distrust	16	256	-0.070	-0.031	+0.074	11
Memory failure	15	240	-0.054	-0.154	+1.025	9
Topic derailment	18	288	-0.052	-0.024	+0.128	11
Early termination request	13	208	-0.043	+0.173	+0.212	7
Sudden logistics request	16	256	-0.014	-0.092	+0.012	3
Factual misconception	18	288	+0.010	+0.007	+0.118	5

Table 44: Paired event-target impact in dynamic interviews. Differences are event minus clean. Spoken evidence is the target question’s conversation lenient score; final fill is its final lenient score. Bold marks the largest negative evidence/final-fill change and largest additional turn cost, not a global difficulty ranking.

derived information points in participant utterances; Final Fill measures weighted coverage in the final reconstructed record.

This check is intentionally narrow. It shows that the pipeline remains executable and interpretable for two settings with different participant populations, interview goals, and response formats.

D.18 Additional Dynamic Diagnostic Figures

Appendix Figures 8 and 9 provide supplementary dynamic diagnostics. Together, they show that dynamic interview performance depends on question type, failure stage, and backfilling model. The clean–event robustness gap is shown in the main text.

E Case Study and Error Analysis

E.1 Why MCQ Errors Are Informative

The MCQ task is not simply a factual recall task. Each option is an interviewer action that may be grammatical, polite, and locally plausible. An error often reveals a confusion between two interview strategies rather than a failure to understand the literal dialogue. This is why MCQ remains more discriminative than conditioned QA: in QA, a model can produce fluent interviewer language once the target type is given; in MCQ, it must decide which action is strategically appropriate.

E.2 Option-Level Error Patterns

Here, CORE-MCQ-4500 denotes the static local-decision MCQ subset for *Questioning*, *Follow-up*, and *Response*. We group common MCQ errors by the kind of distractor that attracts models. These groups are based on wrong selections for which the selected option can be linked to a recorded construction source in `distractor_info`, not on arbitrary option letters. The percentages below describe the source-linked wrong-option subset.

Wrong evidence span. These distractors are built from text explicitly mentioned in the dialogue. They are difficult because the option is grounded in a real quote or real topic, but it probes the wrong evidence span or the wrong local focus. In the final Core-MCQ-4500 analysis, this category accounts for 50.4% of source-linked wrong selections. This means that many errors are evidence-selection errors: the model finds something relevant-looking in the context but does not identify the exact respondent claim that the interviewer should address next.

Nearby real turn. These distractors are borrowed from real interviewer turns near the current context. They are fluent and interview-like because a human interviewer actually asked them somewhere nearby, but they are not the right action for the current decision point. They account for 42.3%

Source	Setting	Cases	Runs	Structured	Probing	Adaptive	Agentic
Developing Writers	Higher-education research	10	40	96.9 / 92.3	98.0 / 93.8	94.2 / 83.7	92.4 / 88.7
Parent influenza interviews	Semi-structured health	10	40	95.7 / 85.9	94.6 / 84.0	94.3 / 85.3	88.7 / 76.0

Table 45: Small non-NPR construction-and-execution compatibility check. Cells are Conversation Evidence / Final Fill (%). These semantic information-point scores are not numerically interchangeable with the rule-based Spoken Truth and Direct Backfill metrics in the main benchmark.

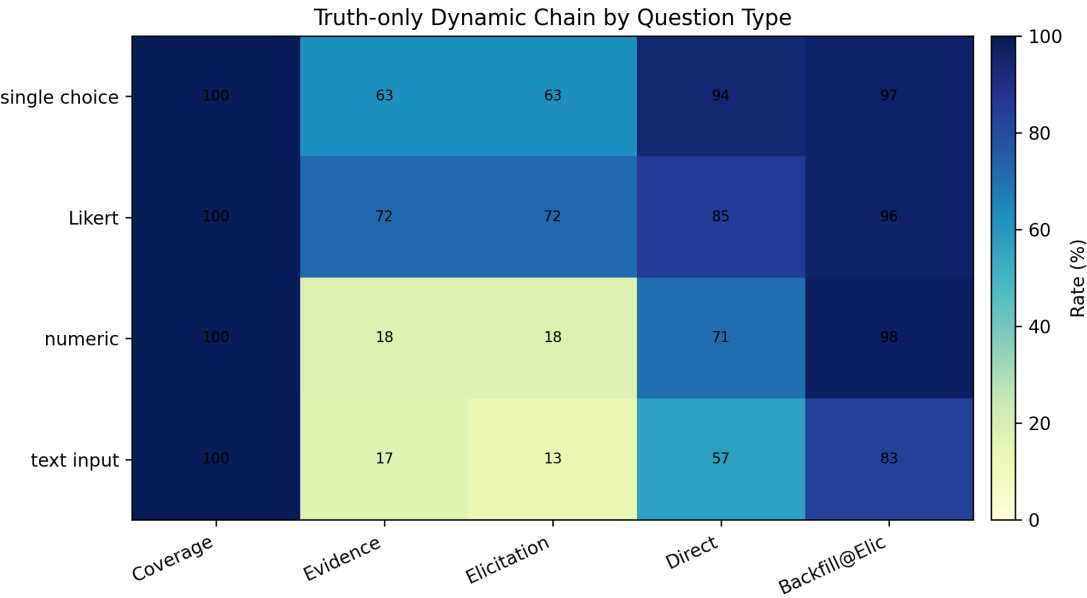


Figure 8: Truth-only dynamic diagnostics by question type. Structured fields are easier to normalize, while numeric and text-input fields require more explicit respondent evidence.

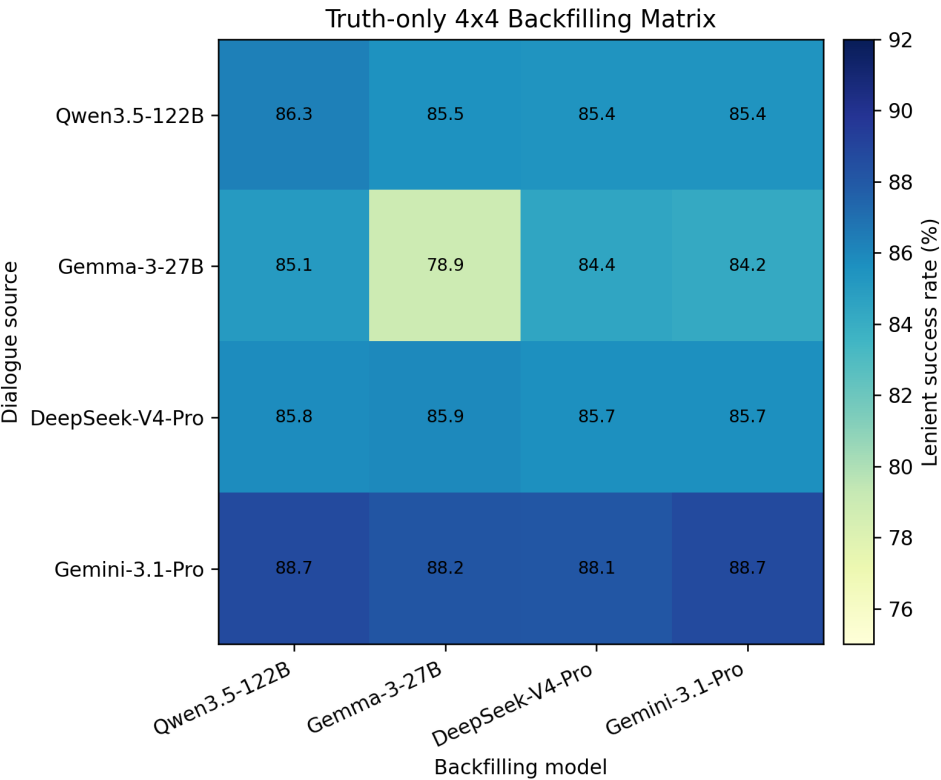


Figure 9: Truth-only 4-by-4 backfilling matrix. Rows fix dialogue source model and columns vary questionnaire filler, including self-filling diagonals.

of source-linked wrong selections. This error type is especially important because it shows that models can be attracted to generally plausible interview behavior even when it is locally misaligned.

Generated wrong technique. These distractors are generated by applying a different interview technique from the gold action, such as asking for causes, asking for components, asking for comparison, posing a hypothetical, or playing devil’s advocate. They account for 7.1% of source-linked wrong selections. Within this category, the most common wrong technique is `ask_how`, followed by `ask_how_common`, `ask_meaning`, `ask_for_components`, and `ask_for_comparison`.

History real wrong move. These are real historical interviewer moves reused in an incompatible context. A small number of source-linked errors fall into this category when such provenance is available. They illustrate a discourse-level failure: an action may be natural in its original location but strategically wrong after a different respondent turn.

Appendix Table 46 further decomposes the generated-wrong-technique subset only.

Generated type	Wrong selections	Share
<code>ask_how</code>	299	33.8
<code>ask_how_common</code>	110	12.4
<code>ask_meaning</code>	96	10.9
<code>ask_for_components</code>	89	10.1
<code>ask_for_comparison</code>	86	9.7
<code>use_hypothetical</code>	64	7.2
<code>pose_puzzle</code>	60	6.8
<code>play_devils_advocate</code>	37	4.2

Table 46: Most frequent generated-distractor techniques selected in wrong answers. Shares are computed within generated wrong-technique distractors.

Beyond the construction source of wrong options, we also summarize recurring functional error patterns observed in these mistakes. These functional patterns are not mutually exclusive with the source categories above: for example, an over-broad question can be a nearby real turn or a generated wrong-technique distractor.

Over-broad question instead of targeted follow-up. In follow-up contexts, models often choose an option that continues the broad topic instead of probing the evidence span that the respondent just made salient. This error is especially common in `missing_information`,

`overgeneral_or_narrowness`, and `story_with_implicit_theme` items. The distractor sounds natural but fails to let the respondent continue speaking without resolving the missing or underspecified part of the previous answer.

New topic instead of repair. When the respondent gives an evasive, ambiguous, or unsupported answer, weaker models sometimes select a new substantive question. This keeps the conversation moving but abandons the local repair opportunity. These errors show that the model recognizes conversational progression but not the methodological need to clarify the current answer before moving on.

Question instead of response. Response items are the hardest in final Core-MCQ-4500. A common error is choosing a question when the appropriate action is an acknowledgement, summary, transition, or reorganization. This suggests a strong model prior that interviewing means asking another question. The model may be linguistically fluent but strategically insensitive to moments where rapport maintenance or discourse management is more appropriate.

Surface politeness instead of contextual response. For acknowledgment items, distractors may contain polite phrases that look socially appropriate in isolation. Models can be attracted to these options even when they do not address the respondent’s specific concern or emotional state. This explains why acknowledgment has low average accuracy in the final Core-MCQ-4500 target-type analysis.

Specificity mismatch. Questioning errors often involve choosing the wrong level of specificity. A closed question may be too narrow when the next step should invite elaboration; an open question may be too abstract when the context calls for a concrete detail. This pattern is visible in lower-scoring closed or ambiguous target types. It indicates that models need a better policy for matching question form to the amount and quality of information already available.

E.3 Commonality Across Error Types

Across these cases, the most wrong options are fluent and interview-like. The shared failure is technique selection: the model does not reliably infer

whether the next move should open a topic, narrow a topic, probe missing evidence, acknowledge the respondent, repair ambiguity, or reorganize the conversation. This explains the main experimental pattern: conditioned QA sharply improves because the technique type is supplied, while MCQ remains challenging because the model must infer the technique itself.

E.4 Recurring Failure Modes and Actionable Implications

The static and dynamic diagnostics can be consolidated into a small set of interview-control failures. Appendix Table 47 distinguishes aggregate evidence from case-level patterns. The latter are not presented as prevalence estimates; they identify failure mechanisms that are visible only after preserving the dialogue, question state, and final record together.

The first two rows explain why fluent language does not reliably produce effective interview guidance. A selected distractor can be grounded in a real local claim but address the wrong claim, or it can be a natural human interviewer turn from a nearby context but be mistimed at the current decision point. The false-normal event errors show an analogous diagnosis failure: a locally subtle need for elaboration, repair, or clarification is treated as ordinary conversational flow. In dynamic interviews, a field can be reached without obtaining an answer aligned with its required proposition, option set, or scale. Conversely, a downstream filler can reconstruct a plausible answer without explicit evidence in the dialogue. These distinctions motivate separate reach, spoken-evidence, and backfill metrics.

E.5 Dynamic Failure Pattern: Event-Induced Loop

One observed failure pattern in the dynamic track is an event-induced loop. When a respondent repeatedly expresses distress, distrust, refusal, or an early-termination request, an autonomous interviewer may keep asking the same or highly similar repair question. This can repeatedly trigger the same event behavior and consume the turn budget before later questionnaire items are reached.

This case type is useful because it shows why robustness should not be measured only by final questionnaire accuracy. The failure involves event recognition, recovery-policy selection, loop control, and progress management. A better inter-

viewer should acknowledge the disruption, limit repeated repair attempts, decide whether to defer or close gracefully, and avoid repeatedly re-triggering the same respondent behavior.

E.6 Dynamic Failure Pattern: Unreached but Filled

Another observed pattern is that a questionnaire field may receive a plausible filled answer even when the corresponding question was not explicitly asked. This may happen when the filler infers an answer from surrounding context or from profile-consistent information. Such cases are useful for system-level performance but should not be counted as direct evidence of interviewer coverage.

For this reason, the current dynamic analysis separates explicit question reach from final filled answers. Where semantic coverage and unreached-but-filled diagnostics are available, they should be reported separately and avoid being folded into a single final questionnaire score.

E.7 Dynamic Case: Repairing a Question Misunderstanding

We next give a paired event case that illustrates successful local repair and the “asked but misaligned” failure mode. The source episode is *Strengthening Buildings in Tornado Alley* from *Talk of the Nation* (2013-06-07; case Tornado-1030). The clean and event cases share the questionnaire, persona, normalized gold record, and answer memory. The event case differs only in a planned question_misunderstanding at the second questionnaire item.

The target is the five-point Likert item: “The disaster described has caused severe negative impacts on the affected communities.” The normalized gold answer is *Strongly agree*. In the event condition, the respondent initially treats the question as asking whether the disaster was human-caused or avoidable, instead of asking how severely the communities were harmed. This answer is conversationally coherent but non-responsive to the intended proposition.

Under Agentic control, the interviewer performs one bounded clarification instead of advancing immediately:

Interviewer: “To clarify, I’m specifically asking about the extent of the hardship and damage experienced by the communities, regardless of whether the cause was natural or man-made.”

Failure mode	Evidence	Design implication
Wrong local evidence span	50.4% of source-linked wrong MCQ selections	Identify the unresolved evidence span before generating the next move.
Locally plausible but misaligned move	42.3% of source-linked wrong MCQ selections	Separate action selection from surface-language realization.
Event-state misdiagnosis	247 of 302 Gemini event-MCQ errors are false-normal choices	Detect the interaction state before selecting a recovery policy.
Asked but misaligned	Reached item, but response addresses a different proposition or scale	Check question-specific answer sufficiency before advancing.
Unspoken but filled	Final field is plausible although the evidence was not explicitly elicited	Report spoken evidence separately from final backfill.
Recovery without progress	Repeated repair can consume turns without restoring target information	Bound repair attempts; defer, return to the agenda, or close.

Table 47: Recurring failure modes in static and dynamic evaluation. The first three rows are aggregate diagnostics; the remaining rows are dialogue-level patterns that require chain-level inspection.

The respondent then answers: “If we are strictly talking about the extent of hardship and damage suffered by the people, then I would have to say **I strongly agree.**” This response both resolves the cause-versus-impact confusion and explicitly supplies a scoreable Likert answer.

All workflows reach the target item and complete the questionnaire, so Completion alone cannot distinguish the outcomes. The Probing run records *Agree* while the respondent is still discussing causation; the Structured run does not supply a structured answer after the request for clarification. Only the bounded Agentic repair restores the intended referent and matches the clean condition’s item and questionnaire scores, at the cost of one additional turn.

This example demonstrates a capability that an effective interview policy should have, not a general Agentic advantage. Aggregate results show that autonomous control can add cost and enter unproductive loops. The actionable lesson is narrower: when an answer addresses a different proposition from the current item, the interviewer should explicitly name the referent, check answer sufficiency, and then either recover the item or defer it.

E.8 Shared MCQ–QA Case: Locally Plausible but Misfocused Follow-up

We use one shared MCQ–QA item to illustrate how the same dialogue context can expose different kinds of error. The case ID is `mcq_fo1_0e98887063`. It appears in both the MCQ benchmark and the QA benchmark. The dimension is *follow-up*, and the target type is *overgeneral_or_narrowness*.

Context. The interview discusses the U.S. Virgin Islands after Hurricane Irma. The respondent argues that the United States treats the islands as secondary and then broadens the point to tourism:

“When Americans travel to Europe, for example, they know that they’re traveling to Europe to engage with the cultural history. When people travel to the Caribbean, they are often traveling to avoid the human beings and to just engage in the beauty of the space.”

The gold interviewer follow-up is:

“So they’re in resorts. They’re lying in the sun. They’re drinking tropical, rum-infused drinks, but they’re not necessarily there for the culture or to talk to the people who live there?”

This is a targeted confirmatory follow-up. It does not merely continue the broader Hurricane Irma topic; instead, it operationalizes the respondent’s broad claim about Caribbean tourism into concrete tourist behaviors and checks whether this is the intended meaning. The case is useful for diagnosing focus errors: a model may choose or generate a locally plausible follow-up while missing the specific evidence span that the interviewer should probe next.

MCQ options and errors. The MCQ asks which follow-up question is most appropriate. The gold answer is option B. Two models in this representative subset select option D instead.

The table reports a representative subset of calibration models used for the shared case analysis, not the full current main-table model set.

Condition	Workflow	Q2 final answer	Q2 strict / lenient	Full strict / lenient	Turns
Clean	Agentic	Strongly agree	1.0 / 1.0	0.889 / 0.889	11
Event	Agentic	Strongly agree after clarification	1.0 / 1.0	0.889 / 0.889	12
Event	Probing	Agree, still cause-focused	0.0 / 0.0	0.778 / 0.778	9
Event	Structured	No structured answer	0.0 / 0.0	0.778 / 0.778	9

Table 48: Paired dynamic case under a question-misunderstanding event. Probing corresponds to the implementation workflow no_workflow; Structured corresponds to single_pass_no_recovery.

Option	Text
A	What were the specific economic conditions in the islands in 1917?
B	So they're in resorts ... but not necessarily there for the culture?
C	So you're saying residents lack federal representation?
D	So how has that impacted this response to Irma?

Table 49: Options for the shared MCQ-QA case. Option B is the gold follow-up.

Model	MCQ choice	Correct
Claude Sonnet 4.6	B	yes
Gemini 2.5 Pro	D	no
Qwen3.5-27B	B	yes
DeepSeek V4 Pro	B	yes
DeepSeek V4 Flash	B	yes
GPT-4o	D	no

Table 50: Model choices for case mcq_fol_0e98887063.

The error is interpretable. Option D is not irrelevant: it connects the respondent's claim to the broader Irma response. However, it skips the immediate interview opportunity. The respondent has just made a broad generalization about American tourism in the Caribbean; the gold follow-up tests that generalization before returning to policy consequences. The wrong option therefore reflects a common failure: choosing a topically plausible continuation instead of the best local follow-up technique.

QA generations. In the conditioned QA version, several models produce fluent and mostly appropriate follow-ups. One high-scoring conditioned generation asks:

“You mentioned that Americans see the Caribbean as a place to escape from people, unlike how they engage with Europe’s cultural history—can you compare how U.S. tourists interacted with Virgin Islanders before Hurricane Irma versus how they’ve responded or been repre-

sented in the aftermath?”

The judge assigns it a composite score of 93.9, with 95 trigger match and 95 technique match. By contrast, GPT-4o generates:

“How would you compare the way cultural history is preserved and presented in the Virgin Islands versus how it’s done in European destinations?”

The judge gives this answer a lower composite score of 74.4. Since the question being asked is fluent the main issue lies in focus. It shifts from the respondent’s generalization about tourist perception to the separate topic of cultural preservation. DeepSeek V4 Pro shows a similar pattern, receiving 71.1 because it shifts toward comparing disaster responses and not probing the tourist-generalization claim.

Generation	Trigger /		Main issue
	Comp.	Tech.	
High-scoring conditioned example	93.9	95 / 95	Probes tourist-perception claim
GPT-4o	74.4	– / –	Shifts to cultural preservation
DeepSeek V4 Pro	71.1	– / –	Shifts to disaster-response comparison

Table 51: Conditioned QA scores for the shared MCQ-QA case. Dashes indicate sub-scores not reported in the case excerpt.

This case explains the aggregate MCQ and QA findings. Models can often produce polished interviewer language, and they may even use a broadly correct technique such as comparison. The remaining challenge is to apply the technique to the exact evidence span that matters in the dialogue.

F Prompts

The pipeline uses separate prompt families for static core-decision construction, static model inputs and QA judging, short-dialogue event recovery, and dynamic long-dialogue interviewing. The

dynamic prompts cover case construction and audit, interviewer control, interviewee simulation, structured backfilling, satisfaction judging, and supporting summaries.

For dynamic evaluation, the interviewee simulator receives the case-specific persona, answer memory, current question, dialogue history, and event plan when applicable. It is instructed to remain consistent with the available evidence and not to reveal the answer memory as a record. This makes the prompt family central to the controlled-simulator design described in Appendix C.

The prompt bodies below are English-only ASCII renderings for pdfLaTeX. The maintenance index records each canonical source file and any presentation-only normalization.

F.1 Static Local-Decision Prompts

This group covers core construction, model inputs, and QA judging for local interviewer-decision tasks.

Core construction.

Filter non-substantive interviewer turns

```
You are a quality controller for interview data. Decide whether this interviewer utterance is a substantive interview action or irrelevant content that should be filtered.
## Filter out (label = "skip")
- Consent procedures or signatures.
- Opening greetings or administrative explanation of the study.
- Closing remarks, thanks, or farewells.
- Technical interruptions or equipment adjustments (for example, "please wait" or "the signal is bad").
- A stand-alone transition with no substantive content (for example, "okay," "mm-hmm," or "I see").
- Small talk wholly unrelated to the interview topic.
## Keep (label = "keep")
- Questions, whether new questions or follow-ups.
- Substantive responses to the respondent, such as summaries, encouragement, reflections, or acknowledgments.
- A transition that substantively redirects the topic.
## Decision rules
- If an utterance contains both irrelevant and substantive material, judge its primary function.
- Do not remove useful material; choose 'keep' when uncertain.
Return exactly one JSON object and no Markdown:
{{
  "label": "keep|skip",
  "confidence": 0.9,
```

Filter non-substantive interviewer turns (continued)

```
"reason": "One-sentence justification"
}}
Item to assess:
{payload_json}
```

Classify interviewer turn type

```
You are an analyst of interviewer behavior. Classify the interviewer utterance into one interview-action type.
## Definitions
**questioning**
The interviewer asks a **new** question that opens a new topic or direction. It begins a new direction rather than extending the respondent's immediately preceding answer.
**followup**
The interviewer anchors on a specific concept, event, or view in the respondent's immediately preceding turn and probes it further. A concrete anchor must be identifiable in that preceding respondent turn.
**response**
The interviewer gives non-questioning feedback to the respondent: encouragement, repetition/paraphrase/summary, neutral acknowledgment, or brief appropriate self-disclosure. Its central purpose is not to obtain new information. A sentence with a question mark can still be a response when its main purpose is to confirm understanding rather than solicit new information.
**other**
Opening or closing remarks, administrative procedures, program logistics, or other material outside the three categories above.
## Decision notes
- The key difference between 'followup' and 'questioning' is the explicit anchor in the immediately preceding respondent turn.
- If uncertain between 'followup' and 'questioning', prefer 'followup' as the more conservative decision.
Return exactly one JSON object and no Markdown:
{{
  "turn_type": "questioning|followup|response|other",
  "confidence": 0.85,
  "anchor_quote": "For followup, quote the specific respondent span being probed; otherwise use an empty string",
  "brief_reason": "One-sentence justification"
}}
Item to assess:
{payload_json}
```

Annotate questioning decisions

```
You are an interview-question type annotator. Classify an interviewer 'questioning' turn along three dimensions and assign its primary function.
## Type-vector definitions
- **openness**: 'open' allows a free response without a prescribed answer frame; 'closed' constrains the answer to yes/no, a choice, or
```

Setting	Prompt family	Role
Static construction	core / event recovery	Construct, verify, and calibrate static instances and distractors
Static model input	Core-MCQ / Core-QA / Event-MCQ / Event-QA	Select, diagnose, or generate the next interviewer move
Static QA judge	core / event recovery	Assess generated interviewer actions with task-specific rubrics
Dynamic long dialogue	case construction and audit	Build questionnaires, gold records, personas, and event plans
Dynamic long dialogue	interviewer control	Plan, choose actions, ask, probe, repair, and close
Dynamic long dialogue	interviewee simulation	Produce persona-, memory-, and event-grounded respondent turns
Dynamic long dialogue	backfilling and satisfaction	Recover questionnaire fields and assess respondent experience
Dynamic long dialogue	supporting summaries	Summarize questionnaire state and completed interviews

Table 52: Prompt families used across the static and dynamic components of INTERVIEWBENCH.

Annotate questioning decisions (*continued*)

a number.

- **specificity**: ‘specific’ targets a concrete fact, event, experience, or detail; ‘abstract’ targets opinions, attitudes, values, or a broad concept.
- **clarity**: ‘clear’ is simply structured and has one readily understood meaning; ‘ambiguous’ is complexly worded, carries multiple meanings, or embeds the interviewer’s assumptions.

Primary function

Choose one:

- ‘opening’: open or introduce a topic
- ‘information_gathering’: request factual information
- ‘opinion_probing’: elicit a view or position
- ‘experience_eliciting’: invite an experience or narrative
- ‘clarification’: request explanation or clarification
- ‘elaboration_request’: ask for further detail
- ‘transition’: move between topics
- ‘closing’: close the interview

Return exactly one JSON object and no Markdown:

```
{
  "type_vector": {
    "openness": "open|closed",
    "specificity": "specific|abstract",
    "clarity": "clear|ambiguous"
  },
  "function": "opening|information_gathering|
  opinion_probing|experience_eliciting|
  clarification|elaboration_request|transition|
  closing",
  "brief_reason": "One-sentence justification"
}
```

Question to annotate:

```
{payload_json}
```

Annotate follow-up elements

You are an annotator of follow-up triggers. Identify what feature of the respondent’s answer prompted the interviewer’s final

Annotate follow-up elements (*continued*)

follow-up.

Trigger types

- ‘missing_information’: the account omits an important step, actor, cause, consequence, or expected dimension.
- ‘evasive_or_unanswered’: the respondent avoided the question, changed topic, or supplied a mismatched/indirect answer.
- ‘overgeneral_or_narrowness’: the answer is vague, sweeping, overly narrow, unsupported by detail, or unrepresentative.
- ‘contradiction_or_ambivalence’: statements or attitudes are inconsistent, conflicting, or contain unresolved tension.
- ‘story_with_implicit_theme’: a narrative contains an unarticulated theme, turning point, conflict, or meaning worth probing.
- ‘unclear_concept’: a term or expression is undefined, ambiguous, or has a context-specific meaning requiring clarification.
- ‘incomplete_or_unsupported_themes’: a theme has appeared but lacks development, examples, scope, explanatory support, or counterevidence.
- ‘new_relevant_idea’: an unexpected but relevant idea, practice, phenomenon, or mechanism has been introduced and merits exploration.

Use the multi-turn dialogue to determine which trigger best explains the final interviewer follow-up.

Return exactly one JSON object and no Markdown:

```
{
  "trigger_type": "missing_information|evasive_
  or_unanswered|overgeneral_or_narrowness|
  contradiction_or_ambivalence|story_with_
  implicit_theme|unclear_concept|incomplete_or_
  unsupported_themes|new_relevant_idea",
  "brief_reason": "One-sentence justification"
}
```

Item to annotate:

```
{payload_json}
```

Annotate response decisions

You are an annotator of interviewer response strategies. Classify a non-questioning interviewer feedback turn.

Valid response strategies (usable = true)

- 'encouraging': positive affective feedback conveying listening, acceptance, or invitation to continue; it does not judge content or introduce new information.
- 'repeat_or_reorganize': repeat, paraphrase, organize, or summarize the respondent's content to confirm understanding without adding the interviewer's judgment.
- 'self_disclosure': briefly share a relevant personal experience, feeling, or position to build rapport without taking over the respondent's space.
- 'acknowledgment': neutrally accept or confirm the respondent's factual or emotional expression without elaboration, judgment, or self-disclosure.

Invalid response strategies (usable = false)

- 'evaluative': subjectively praises or criticizes the respondent's answer.
- 'didactic': explains, corrects, or lectures the respondent, displacing their perspective. Only the four valid strategies above receive 'usable: true'. Use 'other' if none applies. Return exactly one JSON object and no Markdown:

```
{{
  "response_type": "encouraging|repeat_or_reorganize|self_disclosure|acknowledgment|evaluative|didactic|other",
  "usable": true,
  "key_signal": "Verbatim key span that supports the classification",
  "brief_reason": "One-sentence justification"
}}
```

Item to annotate:
{payload_json}

Construct follow-up distractors

You are designing multiple-choice interview questions. Generate three distractors for a 'followup' item.

The gold answer is the interviewer's actual follow-up. Its annotations identify the trigger type, technique type, evidence quote, and evidence description.

Trigger types

'missing_information', 'evasive_or_unanswered', 'overgeneral_or_narrowness', 'contradiction_or_ambivalence', 'story_with_implicit_theme', 'unclear_concept', 'incomplete_or_unsupported_themes', and 'new_relevant_idea' respectively denote missing content, avoidance/non-answer, overgeneral or narrow content, contradiction/ambivalence, implicit narrative theme, unclear concept, underdeveloped/unsupported theme, and a new relevant idea.

Technique types

'ask_how', 'ask_for_comparison', 'echo_back', 'offer_alternatives', 'ask_for_components', 'ask_about_exceptions', 'play_devils_advocate', 'pose_puzzle', 'ask_how_common',

Construct follow-up distractors (continued)

'use_hypothetical', and 'ask_meaning' respectively probe process, comparison, a repeated keyword, alternatives, components, exceptions, a contrary view, a puzzle, prevalence, a hypothetical, and meaning.

General requirements

- Each distractor must sound natural in the given interview and be plausibly tempting, but not be the best next follow-up.
- The reason each distractor is wrong must be different.

Distractor A: wrong trigger, mentioned in dialogue

Ask about material actually mentioned in the dialogue, but target a different trigger type from the gold answer. Set 'mentioned_in_dialogue' true and quote the source text.

Distractor B: wrong evidence focus, mentioned in dialogue

Ask about material actually mentioned in the dialogue and related to the same broad trigger, but focus on evidence different from the gold 'evidence_quote'. Set 'mentioned_in_dialogue' true and quote the source text.

Distractor C: wrong technique

Use a different follow-up technique. Prefer a real item in 'nearby_same_dimension'; otherwise generate one. Set 'source' to 'adjacent_real' or 'generated'.

Return exactly one JSON object and no Markdown:

```
{{
  "distractor_a": {{
    "text": "Distractor A follow-up text",
    "mentioned_in_dialogue": true,
    "source_quote": "Relevant verbatim dialogue span",
    "brief_reason": "Why it is not the best follow-up"
  }},
  "distractor_b": {{
    "text": "Distractor B follow-up text",
    "mentioned_in_dialogue": true,
    "source_quote": "Relevant verbatim dialogue span",
    "brief_reason": "Why it is not the best follow-up"
  }},
  "distractor_c": {{
    "text": "Distractor C follow-up text",
    "type": "ask_how|ask_for_comparison|echo_back|offer_alternatives|ask_for_components|ask_about_exceptions|play_devils_advocate|pose_puzzle|ask_how_common|use_hypothetical|ask_meaning",
    "source": "adjacent_real|generated",
    "brief_reason": "Why this technique is inappropriate here"
  }}
}}
```

Current item:
{payload_json}

Verify follow-up MCQ items

You are the final quality controller for an interview-benchmark 'followup' multiple-choice item.

Verify follow-up MCQ items (continued)

Assess:

- Does the gold answer's 'target_type' / follow-up type accurately describe it?
- Does distractor A ask about material mentioned in the dialogue but use the wrong trigger type?
- Does distractor B ask about mentioned material but focus on the wrong evidence span?
- Does distractor C use a technique different from the gold answer while remaining plausibly tempting?
- Is the respondent anchor clear enough to support the decision?
- Are all four options distinct, with exactly one logically best answer?
- Is the gold follow-up the most appropriate move after the respondent's current answer?

Overall quality (1-5)

- 5: precise follow-up design; distinctive distractors; ready to include
- 4: broadly usable with minor issues
- 3: marginally usable; revise before use
- 1-2: unusable

Return exactly one JSON object and no Markdown:

```
{{
  "label_correct": true,
  "distractors_valid": true,
  "distractor_check": {{
    "B": {{ "valid": true, "issue": "" }},
    "C": {{ "valid": true, "issue": "" }},
    "D": {{ "valid": true, "issue": "" }}
  }},
  "answer_unique": true,
  "context_sufficient": true,
  "overall_quality": 4,
  "issues": [],
  "brief_verdict": "One-sentence overall assessment"
}}
```

MCQ item to verify:

```
{payload_json}
```

Model-input prompts.

Core-MCQ answer selection

You are taking an interviewing skills assessment. Read the dialogue and choose the best answer.

Dialogue Context

```
{context_text}
```

Question

```
{stem}
```

A. {option_a}
B. {option_b}
C. {option_c}
D. {option_d}

Output a single JSON object only, no Markdown:

```
{{"answer": "A|B|C|D", "confidence": 0.9,
  "reasoning": "one-sentence reason"}}
```

Core-QA unconditioned: questioning

You are participating in a qualitative research interview as the **interviewer**. The following is the conversation context so far, containing all turns between the

Core-QA unconditioned: questioning (continued)

interviewer and the interviewee.

Interview Context

```
{context_turns}
```

-

Your Task

The conversation ends at turn {target_turn_index}. It is now the interviewer's turn to ask a question.

Based on the conversation above, generate the single most appropriate question the interviewer should ask next.

Requirements:

- The question must directly address something the interviewee just said, or explore an aspect not yet discussed in depth
- Use natural phrasing, as a real interviewer would speak – not like a survey or exam question
- Output only one question, with no additional explanation

Output format (single JSON object, no Markdown code block):

```
{
  "question": "the interviewer's question text"
}
```

Core-QA conditioned: questioning

You are participating in a qualitative research interview as the **interviewer**. The following is the conversation context so far, containing all turns between the interviewer and the interviewee.

Interview Context

```
{context_turns}
```

-

Your Task

The conversation ends at turn {target_turn_index}. It is now the interviewer's turn to ask a question.

You must generate a question of the following specific type:

```
> **Target type:** '{target_type_label}'
> **Definition:** {target_type_definition}
```

Based on the conversation above, generate exactly one question that (a) matches the specified type and (b) naturally fits the current conversational moment.

Requirements:

- The question **must** conform to the target type as defined above – this is the primary constraint
- It must connect to what the interviewee has said, not be a generic question
- Use natural phrasing, as a real interviewer would speak
- Output only one question, with no additional explanation

Output format (single JSON object, no Markdown code block):

```
{
  "question": "the interviewer's question text"
}
```

Core-QA unconditioned: follow-up

You are participating in a qualitative research interview as the **interviewer**. The following is the conversation context so far, containing all turns between the interviewer and the interviewee.

Interview Context
{context_turns}
-

Your Task
The conversation ends at turn {target_turn_index}. The interviewee has just given a response. It is now the interviewer's turn to ask a **follow-up question**.
A follow-up question probes deeper into something the interviewee just said – it does not introduce a new main topic.
Based on the conversation above, generate the single most appropriate follow-up question the interviewer should ask next.

Requirements:

- The follow-up must respond to something specific in the interviewee's most recent answer
- Use natural phrasing, as a real interviewer would speak
- Output only one follow-up question, with no additional explanation

Output format (single JSON object, no Markdown code block):

```
{  
  "question": "the interviewer's follow-up question text"  
}
```

Core-QA conditioned: follow-up

You are participating in a qualitative research interview as the **interviewer**. The following is the conversation context so far, containing all turns between the interviewer and the interviewee.

Interview Context
{context_turns}
-

Your Task
The conversation ends at turn {target_turn_index}. The interviewee has just given a response. It is now the interviewer's turn to ask a **follow-up question**.
You must generate a follow-up question with the following specific profile:

> **Trigger** (what in the respondent's answer to respond to):
> '{target_trigger_type}' – {target_trigger_definition}
>

> **Technique** (how to probe):
> '{target_technique_type}' – {target_technique_definition}

The trigger tells you *what* to focus on in the respondent's last answer; the technique tells you *how* to construct your probe. Based on the conversation above, generate exactly one follow-up question that (a) responds to the specified trigger in the interviewee's most recent answer and (b) applies the specified technique.

Requirements:

Core-QA conditioned: follow-up (continued)

- The follow-up **must** implement the specified trigger and technique – this is the primary constraint
- It must be anchored in something specific the interviewee actually said, not be a generic probe
- Use natural phrasing, as a real interviewer would speak
- Output only one question, with no additional explanation

Output format (single JSON object, no Markdown code block):

```
{  
  "question": "the interviewer's follow-up question text"  
}
```

Core-QA unconditioned: response

You are participating in a qualitative research interview as the **interviewer**. The following is the conversation context so far, containing all turns between the interviewer and the interviewee.

Interview Context
{context_turns}
-

Your Task
The conversation ends at turn {target_turn_index}. The interviewee has just given a response. It is now the interviewer's turn to give a **non-questioning response**.
A non-questioning response does not ask a new question. Instead, it uses verbal behavior – such as showing acknowledgment, reflecting content, or expressing rapport – to maintain the conversation, build trust, or confirm understanding.
Based on the conversation above, generate the single most appropriate response the interviewer should give next.

Requirements:

- Do not ask a question
- Do not evaluate or judge the correctness of what the interviewee said
- Keep it brief and natural, as a real interviewer would speak after listening to a response
- Output only one response, with no additional explanation

Output format (single JSON object, no Markdown code block):

```
{  
  "response": "the interviewer's response text"  
}
```

Core-QA conditioned: response

You are participating in a qualitative research interview as the **interviewer**. The following is the conversation context so far, containing all turns between the interviewer and the interviewee.

Interview Context
{context_turns}
-

Your Task
The conversation ends at turn {target_turn_

Core-QA conditioned: response (continued)

index}. The interviewee has just given a response. It is now the interviewer's turn to give a **non-questioning response**. You must produce a response of the following specific type:

```
> **Target response type:** '{target_response_type}'
> **Definition:** {target_response_definition}
```

Based on the conversation above, generate exactly one response that (a) matches the specified type and (b) naturally follows from what the interviewee just said.

Requirements:

- The response **must** conform to the target type as defined above – this is the primary constraint
- Do **not** ask a question
- Do **not** evaluate or judge the correctness of what the interviewee said
- Keep it brief and natural, as a real interviewer would speak after listening
- Output only one response, with no additional explanation

Output format (single JSON object, no Markdown code block):

```
{
  "response": "the interviewer's response text"
}
```

QA-judge prompts.

Core-QA judge: questioning

You are an expert evaluator of qualitative research interviewing skills. Your task is to assess the quality of an interview question generated by an AI model.

–

Type System

An interview question (questioning) is described along two dimensions: **type_vector** and **function**.

Type Vector (three-dimensional combination)

Format: '{openness}_{specificity}_{clarity}'

Dimension 1: Openness

- 'open': allows the respondent to answer freely, without a predetermined answer framework
- 'closed': restricts the answer range, typically yes/no, choice, or a number

Dimension 2: Specificity

- 'specific': targets a concrete fact, event, experience, or detail
- 'abstract': targets an opinion, attitude, value judgment, or broad concept

Dimension 3: Clarity

- 'clear': simple and unambiguous structure, easy to understand
- 'ambiguous': complex structure, or carries multiple meanings or personal bias

Function (question function)

- 'opening': warm-up or topic introduction
- 'information_gathering': collecting specific facts, data, or experiential details
- 'opinion_probing': eliciting opinions, views, or attitudes

Core-QA judge: questioning (continued)

- 'experience_eliciting': inviting the respondent to narrate a personal experience or story
- 'clarification': clarifying something ambiguous in the respondent's account
- 'elaboration_request': requesting expansion or elaboration on something already mentioned
- 'transition': changing topic or advancing the interview
- 'closing': closing the interview or wrapping up a topic
-

Scoring Criteria

1. trigger_type_match (continuous 0-100 score)

First judge the model's type_vector independently, then compare with gold (three dimensions: openness x specificity x clarity):

- **90-100**: All three type_vector dimensions match gold exactly
- **60-89**: Minor borderline difference (e.g., clear/ambiguous boundary); overall very close to gold
- **35-59**: Exactly one type_vector dimension differs from gold
- **1-34**: Two or more type_vector dimensions differ from gold
- **0**: The question direction is completely unrelated to gold, or not a valid interview question

2. technique_type_match (continuous 0-100 score)

First judge the model's function independently, then compare with gold:

- **90-100**: function matches gold exactly or is functionally equivalent
- **60-89**: Minor functional deviation; actual usage highly overlaps with gold
- **35-59**: function differs from gold, but is a reasonable functional substitute in the same interview segment
- **1-34**: function clearly differs from gold; some interviewing intent remains visible
- **0**: function completely contradicts gold, or the question lacks any interview function

3. language_quality (continuous 0-100 score)

Independent of category match, assess the linguistic quality of the question. Output three sub-scores; the framework will compute the mean:

Sub-dimension definitions:

- **naturalness**: Whether the language sounds like a real interviewer speaking – natural, avoids mechanical or formulaic phrasing; appropriate register for interviewing
- **fluency**: Whether sentences are grammatically correct and read smoothly without effort
- **accuracy**: Whether the response accurately addresses the conversational context (closely tied to what the respondent just said); logically sound, appropriately polite, non-evaluative and non-didactic

Scoring ranges (same standards for all three sub-dimensions):

Core-QA judge: questioning (continued)

```
- **90-100**: Excellent – almost no room for improvement; reaches the level of a skilled interviewer
- **75-89**: Good – suitable for direct use in a real interview; quality is consistent
- **50-74**: Adequate to usable – notable weaknesses but basic communication is not impaired
- **25-49**: Poor – clear problems that would affect interview quality
- **0-24**: Unusable – severe defects; cannot be used in a real interview
-
## Content to Evaluate
### Interview Context
{context_turns}
### Ground Truth
- **Category**: '{gold_category}'
- **Text**: {gold_answer_text}
### Model-Generated Question
{model_answer_text}
-
## Output Requirements
First provide an independent analysis of the model-generated question, then give your scores.
Output must be a single JSON object with no Markdown code block:
{{
  "model_type_vector": "open/closed_specific/abstract_clear/ambiguous",
  "model_function": "function_type",
  "trigger_type_match": 0,
  "technique_type_match": 0,
  "naturalness": 50,
  "fluency": 50,
  "accuracy": 50,
  "brief_verdict": "one-sentence overall assessment",
  "issues": ["issue 1 (if any)", "issue 2 (if any)"]
}}
```

Core-QA judge: follow-up

You are an expert evaluator of qualitative research interviewing skills. Your task is to assess the quality of a follow-up question generated by an AI model.

-

Type System

A follow-up question (followup) is described along two independent dimensions: **trigger_type** (motivating factor) and **technique_type** (probing technique).

Trigger Type – What the follow-up responds to

trigger_type	Name	Key definition
- -		
'missing_information'	Missing information	The account is missing steps, actors, or expected elements; the researcher identifies an important cause, consequence, or dimension that was not mentioned
'evasive_or_unanswered'	Evasive or unanswered	The respondent did not respond, changed the subject, or gave an answer that did not match the question; possible

Core-QA judge: follow-up (continued)

	avoidance of sensitive content or "unsaid information"	
'overgeneral_or_narrowness'	Overly general or narrow	Superficial, low-information, lacking concrete detail; or overly sweeping/absolute; or too narrow to be representative
'contradiction_or_ambivalence'	Contradiction or ambivalence	Inconsistent or contradictory statements; the same phenomenon is evaluated both positively and negatively; contradictory attitudes expressed
'story_with_implicit_theme'	Story with implicit theme	Respondent replies in narrative form; an important theme in the story has not been explicitly named; a turning point or conflict deserves deeper exploration
'unclear_concept'	Unclear concept	An undefined or ambiguous concept/term was used; the concept carries a special meaning in this specific context
'incomplete_or_unsupported_themes'	Incomplete or unsupported themes	A theme that has appeared but not yet been fully developed; lacks supporting examples; insufficient explanatory scope or depth
'new_relevant_idea'	New relevant idea	Respondent introduces new content the researcher did not anticipate but that is relevant; a new phenomenon or mechanism not yet fully understood
### Technique Type – How the follow-up is posed		
technique_type	Name	Key definition
- -		
'ask_how'	Ask how	Uses "how" to probe process, mechanism, or path of change; invites the respondent to narrate step by step
'ask_for_comparison'	Ask for comparison	Asks the respondent to compare two objects, situations, or experiences; contrasts different time periods or dimensions
'echo_back'	Echo back	Repeats a keyword the respondent mentioned but did not elaborate and turns it into a question; minimal intervention to prompt elaboration
'offer_alternatives'	Offer alternatives	In response to a negative statement, asks "then what is it / how did it happen"; guides toward other possible explanations
'ask_for_components'	Ask for components	Breaks down a holistic answer into specific constituent parts; successive follow-ups to probe progressively deeper
'ask_about_exceptions'	Ask about exceptions	For a generalising judgment, asks whether exceptions exist; explores boundary conditions or cases where it breaks down
'play_devils_advocate'	Play devil's advocate	Raises a challenging position contrary to the respondent's view; indirectly questions via a hypothetical counter-argument
'pose_puzzle'	Pose puzzle	Poses an

Core-QA judge: follow-up (continued)

explanatory follow-up based on the researcher's own puzzlement; takes the form of "I don't quite understand / how do you explain" |

| 'ask_how_common' | Ask how common | Asks about the frequency or prevalence of an experience or phenomenon |

| 'use_hypothetical' | Use hypothetical | Constructs an "if ... what would happen" scenario; tests the respondent's judgment under different conditions |

| 'ask_meaning' | Ask meaning | Directly asks for the meaning of a concept or expression the respondent used; requests an example, contrast, or definition |

-

Scoring Criteria

1. trigger_type_match (continuous 0-100 score)

First judge the model's trigger_type independently, then compare with gold:

- **90-100**:: trigger_type matches gold exactly; follow-up clearly targets the specific trigger element from gold
- **60-89**:: trigger_type has a minor borderline mismatch, but the core probing motive is very close to gold
- **35-59**:: trigger_type differs from gold, but the follow-up intent is adjacent and contextually understandable
- **1-34**:: trigger_type differs clearly; only slight relevance to the target
- **0**:: trigger_type is fundamentally different; follow-up direction is clearly off or invalid

2. technique_type_match (continuous 0-100 score)

First judge the model's technique_type independently, then compare with gold:

- **90-100**:: technique_type matches gold exactly; probing approach naturally fits the trigger
- **60-89**:: technique_type has a minor borderline mismatch; overall probing approach is very close
- **35-59**:: technique_type differs from gold, but the approach used is still a valid interview technique
- **1-34**:: technique_type clearly differs from gold, but some probing intent is visible
- **0**:: technique_type is completely different; the approach is invalid or violates interview principles

3. language_quality (continuous 0-100 score)

Independent of category match, assess the linguistic quality of the follow-up. Output three sub-scores; the framework will compute the mean:

Sub-dimension definitions:

- **naturalness**:: Whether the language sounds like a real interviewer speaking – natural, avoids mechanical or formulaic phrasing; appropriate register for interviewing
- **fluency**:: Whether sentences are grammatically correct and read smoothly without effort
- **accuracy**:: Whether the response

Core-QA judge: follow-up (continued)

accurately addresses the conversational context (closely tied to what the respondent just said); logically sound, appropriately polite, non-evaluative and non-didactic

Scoring ranges (same standards for all three sub-dimensions):

- **90-100**:: Excellent – almost no room for improvement; reaches the level of a skilled interviewer
- **75-89**:: Good – suitable for direct use in a real interview; quality is consistent
- **50-74**:: Adequate to usable – notable weaknesses but basic communication is not impaired
- **25-49**:: Poor – clear problems that would affect interview quality
- **0-24**:: Unusable – severe defects; cannot be used in a real interview

-

Content to Evaluate

Interview Context

{context_turns}

Ground Truth

- **Category**:: '{gold_category}'
- **Text**:: {gold_answer_text}

Model-Generated Follow-up Question

{model_answer_text}

-

Output Requirements

First provide an independent analysis of the model-generated follow-up, then give your scores.

Output must be a single JSON object with no Markdown code block:

```
{
  "model_trigger_type": "trigger_type",
  "model_technique_type": "technique_type",
  "trigger_type_match": 0,
  "technique_type_match": 0,
  "naturalness": 50,
  "fluency": 50,
  "accuracy": 50,
  "brief_verdict": "one-sentence overall assessment",
  "issues": ["issue 1 (if any)", "issue 2 (if any)"]
}
```

Core-QA judge: response

You are an expert evaluator of qualitative research interviewing skills. Your task is to assess the quality of a non-questioning response (response) generated by an AI model acting as an interviewer.

-

Type System

A response refers to the interviewer's **non-questioning verbal behavior**, divided into appropriate and inappropriate strategies.

Appropriate Response Strategies (usable = true, valid answer candidates)

response_type	Name	Definition
- -		
'encouraging'	Encouraging	Provides positive emotional feedback, signalling "I'm listening, I accept this, please continue."

Core-QA judge: response (continued)

Does not evaluate content as right or wrong, does not introduce new information. E.g., "Yes," "Mm-hmm," "I understand how you feel," "Please go on." |

| 'repeat_or_reorganize' | Repeat / reorganize / summarize | Repeats, paraphrases, organizes, or summarises what the respondent just said, helping to confirm understanding or structure information. Content comes from the respondent's own words; the interviewer does not add personal judgment. E.g., "So you're saying...", "In other words...", "You just mentioned three things..." |

| 'self_disclosure' | Self-disclosure (moderate) | The interviewer appropriately shares a relevant personal experience, feeling, or position to build rapport and closeness. Brief, does not dominate the respondent's space; the goal is to build trust. |

| 'acknowledgment' | Acknowledgment | Accepts and confirms what the respondent expressed – factually or emotionally – without elaborating or self-disclosing. Tone is neutral, no subjective praise or criticism. E.g., "Indeed," "That's quite common," "That makes sense." |

Inappropriate Response Strategies (usable = false, should receive low scores)

response_type	Name	Definition	Typical example
- - - -			
'evaluative'	Evaluative	Makes a subjective positive or negative judgment about the respondent's answer "That's absolutely right!" (excessive affirmation), "That idea is wrong."	
'didactic'	Didactic	Explains, corrects, or dispenses knowledge to the respondent, taking over the conversation "Actually, the definition of ... is ..."	

–

Scoring Criteria

1. trigger_type_match (continuous 0-100 score)

First judge the model's response_type independently, then compare with gold:

- **90-100**:: response_type matches gold exactly; closely fits the interactive function and tone of gold
- **60-89**:: response_type is basically consistent or very close; only a minor borderline difference
- **35-59**:: response_type differs, but both are appropriate strategies (usable=true) from adjacent categories
- **1-34**:: Clearly different from gold, but still constitutes some form of usable non-questioning response
- **0**:: Model output is evaluative or didactic (inappropriate strategy), or is not a valid response

2. technique_type_match (continuous 0-100 score)

Assess whether the model response correctly executes the functional requirements of its response type:

- **90-100**:: Fully satisfies the functional

Core-QA judge: response (continued)

definition of the response_type; no evaluative or didactic elements if a usable strategy

- **60-89**:: Largely satisfies functional requirements; minor deviation does not undermine strategy effectiveness
- **35-59**:: Partially satisfies; clear mixing of elements from other strategy types
- **1-34**:: Strategy execution is confused; difficult to classify or effect clearly diverges from intent
- **0**:: Strategy execution is entirely wrong; if output is evaluative/didactic, this score is 0 regardless of intent

3. language_quality (continuous 0-100 score)

Independent of category match, assess the linguistic quality of the response. Output three sub-scores; the framework will compute the mean:

Sub-dimension definitions:

- **naturalness**:: Whether the language sounds like a real interviewer speaking – natural, avoids mechanical or formulaic phrasing; appropriate register for interviewing
- **fluency**:: Whether sentences are grammatically correct and read smoothly without effort
- **accuracy**:: Whether the response accurately addresses the conversational context (closely tied to what the respondent just said); logically sound, appropriately polite, non-evaluative and non-didactic

Scoring ranges (same standards for all three sub-dimensions):

- **90-100**:: Excellent – almost no room for improvement; reaches the level of a skilled interviewer
- **75-89**:: Good – suitable for direct use in a real interview; quality is consistent
- **50-74**:: Adequate to usable – notable weaknesses but basic communication is not impaired
- **25-49**:: Poor – clear problems that would affect interview quality
- **0-24**:: Unusable – severe defects; cannot be used in a real interview

–

Content to Evaluate

Interview Context

```
{context_turns}
### Ground Truth
- **Category**:: '{gold_category}'
- **Text**:: {gold_answer_text}
### Model-Generated Response
{model_answer_text}
–
```

Output Requirements

First provide an independent analysis of the model-generated response, then give your scores.

Output must be a single JSON object with no Markdown code block:

```
{
  "model_response_type": "response_type",
  "is_usable_type": true,
  "trigger_type_match": 0,
  "technique_type_match": 0,
```

Core-QA judge: response (continued)

```
"naturalness": 50,
"fluency": 50,
"accuracy": 50,
"brief_verdict": "one-sentence overall
assessment",
"issues": ["issue 1 (if any)", "issue 2 (if
any)"]
}}
```

E.2 Event-Recovery Prompts

This group covers construction, generation, and judging for short-dialogue event recovery.

Case construction.

Extract candidate event moments

Prompt 01: Extract Candidate TV Interview Moments
You are constructing a short-dialogue benchmark for qualitative TV interviews. Your task is to inspect a TV interview excerpt and extract local moments that can support a short diagnostic benchmark item. A valid candidate moment should test what an interviewer should do next: clarify, probe, repair, redirect, accommodate, or continue naturally.
Use only the provided interview text. Do not rely on outside knowledge.

Input

```
“‘json
```

```
{
  "source_case_id": "...",
  "dialogue_id": "...",
  "source_dataset": "TV program",
  "language": "en",
  "dimension": "...",
  "target_type": "...",
  "context_turns": [],
  "target_turn": {},
  "anchor_quote": "",
  "metadata": {}
}
```

Candidate requirements

Keep a candidate only if:

1. The source is a TV interview.
2. The language is English.
3. The local window contains enough evidence to judge the next interviewer move.
4. The respondent's current turn creates a clear interviewing need, such as clarification, elaboration, tension probing, refusal handling, trust repair, or normal continuation.
5. The candidate can be assigned to a concrete v3 event type.

Reject candidates if:

1. The moment requires outside facts to understand.
2. The transcript is too fragmented to judge.
3. The only issue is questionnaire answer formatting.
4. The next interviewer move is arbitrary.
5. The local context is too long or too diffuse for a short-dialogue benchmark.

Output

Extract candidate event moments (continued)

Return strict JSON:

```
“‘json
{
  "source_case_id": "...",
  "dialogue_id": "...",
  "keep_candidates": [
    {
      "candidate_id": "...",
      "local_window": [
        {"role": "interviewer", "text": "..."},
        {"role": "respondent", "text": "..."}
      ],
      "anchor_quote": "...",
      "candidate_trigger": "short English label",
      "candidate_recovery_need": "what the
interviewer should do next",
      "likely_event_type": "one event type from the
v3 taxonomy",
      "suggested_target_type": "a short
event-oriented subtype label, not 'other'",
      "quality_score": 1,
      "quality_reason": "one concise sentence"
    }
  ],
  "rejected_reason": ""
}
```

Use an integer 'quality_score' from 1 to 5. Keep only candidates with score 4 or 5.

Build short-dialogue event cases

Prompt 02: Build a v3 Short-Dialogue Case
You are building a short-dialogue benchmark case from an English TV interview candidate moment.

The benchmark evaluates qualitative interviewing skill, especially how well a model can diagnose and recover from a local interactional challenge. Do not create questionnaire-format errors. The event type 'user_answer_format_error' is not allowed in v3.

Allowed event types

- 'question_misunderstanding'
- 'passive_noncooperation'
- 'hostility_or_distrust'
- 'topic_refusal'
- 'sudden_logistics_request'
- 'early_termination_request'
- 'factual_misconception'
- 'contradiction_or_ambivalence'
- 'incomplete_or_unsupported_claim'
- 'needs_elaboration'
- 'no_major_event'

Input

```
“‘json
```

```
{
  "candidate_id": "...",
  "source_case_id": "...",
  "dialogue_id": "...",
  "source_dataset": "TV program",
  "language": "en",
  "dimension": "...",
  "target_type": "may be unassigned before
construction",
  "local_window": [],
  "anchor_quote": ""
```

Build short-dialogue event cases (continued)

```
"candidate_trigger": "",
"candidate_recovery_need": ""
}
```

```
““
```

Task

Create one benchmark case that preserves the original interview meaning. You may lightly normalize punctuation, but do not invent new substantive content.

Choose:

1. The best event type.
2. The recovery goal.
3. The concrete recovery policy.
4. The target recover state.
5. A gold next interviewer turn in natural English.
6. Acceptable alternate event and policy labels, if any.

Output

Return strict JSON:

```
““json
{
  "case_id": "SDV3-...",
  "source_case_id": "...",
  "dialogue_id": "...",
  "source_dataset": "TV program",
  "language": "en",
  "dimension": "...",
  "target_type": "a concrete subtype label
derived from the local interviewing
challenge",
  "case_unit_type": "single_recovery_turn",
  "event_type": "...",
  "recovery_goal": "...",
  "recovery_policy": "...",
  "target_recover_state": "...",
  "context_turns": [],
  "local_window": [],
  "anchor_quote": "...",
  "gold_next_interviewer_turn": "...",
  "acceptable_event_types": [],
  "acceptable_recovery_policies": [],
  "provenance": {
    "candidate_id": "...",
    "construction_notes": "one concise sentence"
  }
}
““
```

The gold next interviewer turn should be one sentence unless the interaction clearly requires a two-sentence repair.

Construct injected event cases

Prompt 02b: Construct a Short-Dialogue Case by Injecting a Target Event

You are constructing a benchmark case for an English TV interview short-dialogue evaluation.

You are given a real TV interview context window and a target event specification. Your task is to:

1. Write a ****new respondent anchor turn**** that fits naturally after the context and clearly exhibits the target event at the target severity level.
2. Write the ****gold next interviewer turn**** that appropriately handles this event

Construct injected event cases (continued)

according to best practice.

Do not reuse the original respondent turn verbatim. The anchor turn must be an original construction that authentically reflects TV interview speech.

Allowed event types

- 'question_misunderstanding'
- 'passive_noncooperation'
- 'hostility_or_distrust'
- 'topic_refusal'
- 'sudden_logistics_request'
- 'early_termination_request'
- 'factual_misconception'
- 'contradiction_or_ambivalence'
- 'incomplete_or_unsupported_claim'
- 'needs_elaboration'
- 'no_major_event'

Severity definitions

- 'mild': The event is present but subtle; the interview can continue with a light touch.
- 'moderate': The event is clear and requires a deliberate recovery strategy to continue productively.
- 'severe': The event is strong enough to risk derailing the interview; recovery requires careful handling.

Input

```
““json
{
  "source_case_id": "...",
  "dialogue_id": "...",
  "source_dataset": "TV program",
  "language": "en",
  "real_context": [
    {"role": "interviewer", "text": "..."},
    {"role": "respondent", "text": "..."}
  ],
  "target_event_type": "hostility_or_distrust",
  "severity": "moderate",
  "event_definition": {
    "definition": "...",
    "positive_signals": [],
    "severity_levels": {
      "mild": "...",
      "moderate": "...",
      "severe": "..."
    }
  },
  "default_goal": "...",
  "default_policy": "...",
  "target_recover_state": "..."
}
““
```

Construction rules

1. The anchor turn must sound like authentic TV interview speech – natural, colloquial, not scripted.
2. The severity level must be clearly reflected in the anchor turn. A 'mild' event should not read like a 'severe' one.
3. The anchor turn must follow naturally from the real context. Do not introduce new topics that are inconsistent with the preceding exchange.
4. The gold next interviewer turn must execute the 'default_policy' smoothly and naturally. Do not mention event labels or explain the strategy.

Construct injected event cases (continued)

5. For 'factual_misconception': the false premise must be plausible in the context of the interview topic. Do not invent implausible claims.
6. For 'hostility_or_distrust': the anchor turn must show interpersonal tension directed at the interviewer or the interview process, not just disagreement about content.
7. For 'no_major_event': write a natural, cooperative respondent turn. The gold next turn is a natural continuation, not a repair.

Output

Return strict JSON:

```
“json
{
  "case_id": "SDV3C-...",
  "source_case_id": "...",
  "dialogue_id": "...",
  "source_dataset": "TV program",
  "language": "en",
  "construction_mode": "llm_injected",
  "dimension": "short_dialogue_recovery",
  "target_type": "a short label describing the
specific interactional challenge",
  "case_unit_type": "single_recovery_turn",
  "event_type": "...",
  "severity": "mild|moderate|severe",
  "recovery_goal": "...",
  "recovery_policy": "...",
  "target_recover_state": "...",
  "context_turns": [],
  "local_window": [],
  "anchor_quote": "the constructed respondent
anchor turn text",
  "gold_next_interviewer_turn": "one natural
English sentence",
  "acceptable_event_types": [],
  "acceptable_recovery_policies": [],
  "provenance": {
    "source_case_id": "...",
    "construction_notes": "one concise sentence
explaining the injection",
    "severity_rationale": "one concise sentence
explaining why this counts as the specified
severity"
  }
}
“
```

The 'local_window' should include the real context turns plus the constructed anchor turn as the final respondent turn. The 'context_turns' should include the full real context for reference.

Model-input prompts.

Event-QA-1000: unconditioned setting

Prompt 05: Generate Recovery Response, Base Setting

You are acting as a skilled qualitative TV interviewer.

Given a local interview window, write the next interviewer turn. Do not diagnose the event explicitly. Do not explain your reasoning. Write only the interviewer's next utterance. The next utterance should:

1. Fit the immediate context.

Event-QA-1000: unconditioned setting (continued)

2. Respect the respondent.
3. Preserve the natural TV interview style.
4. Recover the interaction when needed, or continue naturally if no repair is needed.
5. Avoid sounding like a questionnaire form.

Input

```
“json
{
  "case_id": "...",
  "local_window": [
    {"role": "interviewer", "text": "..."},
    {"role": "respondent", "text": "..."}
  ],
  "anchor_quote": ""
}
“
```

Output

Return strict JSON:

```
“json
{
  "case_id": "...",
  "next_interviewer_turn": "..."
}
“
```

The answer must be in English and should usually be one sentence.

Event-QA-1000: conditioned setting

Prompt 06: Generate Recovery Response, Conditioned Setting

You are acting as a skilled qualitative TV interviewer.

You are given the local interview window plus the intended event diagnosis and recovery strategy. Write the next interviewer turn that best executes the strategy.

Do not mention the labels. Do not explain your reasoning. Write only the interviewer's next utterance.

Input

```
“json
{
  "case_id": "...",
  "local_window": [],
  "event_type": "...",
  "recovery_goal": "...",
  "recovery_policy": "...",
  "target_recover_state": "..."
}
“
```

Requirements

The next interviewer turn should:

1. Follow the specified recovery policy.
2. Sound natural in a TV interview.
3. Be respectful and non-coercive.
4. Return to the substantive topic when appropriate.
5. Avoid overexplaining the benchmark task.

Output

Return strict JSON:

```
“json
{
  "case_id": "...",
  "next_interviewer_turn": "..."
}
“
```

The answer must be in English and should

Event-QA-1000: conditioned setting (continued)

usually be one sentence.

QA-judge prompt.

Event-QA-1000 judge

Prompt 07: Judge a Generated Recovery Response

You are evaluating a model-generated next interviewer turn for an English TV interview short-dialogue benchmark. Score the response based only on the provided local context and gold labels. Do not require exact wording. Reward responses that naturally execute the intended interviewing move.

Input

```
“‘json
{
  "case_id": "...",
  "local_window": [],
  "event_type": "...",
  "recovery_goal": "...",
  "recovery_policy": "...",
  "target_recover_state": "...",
  "gold_next_interviewer_turn": "...",
  "acceptable_recovery_policies": [],
  "model_next_interviewer_turn": "...”
}
```

Scoring dimensions

Use 0-100 for each dimension.

- ‘event_awareness’: Does the response show awareness of the local interviewing challenge?
 - ‘policy_alignment’: Does it align with the gold or acceptable recovery policy?
 - ‘recovery_execution’: Does it actually perform the repair, probe, redirect, or continuation?
 - ‘target_state_progress’: Does it move the exchange toward the target recover state?
 - ‘interview_naturalness’: Does it sound like a natural TV interview turn?
 - ‘respect_and_safety’: Is it respectful, non-coercive, and safe?
 - ‘composite_score’: Overall score, not a simple average if one dimension is decisive.
- Interpretation:
- 90-100: Excellent, natural, and fully aligned.
 - 75-89: Strong, with minor imperfections.
 - 60-74: Adequate but partial or generic.
 - 40-59: Weak; some relevant intent but poor execution.
 - 20-39: Mostly wrong.
 - 0-19: Invalid, unsafe, or unrelated.

Output

Return strict JSON:

```
“‘json
{
  "case_id": "...",
  "scores": {
    "event_awareness": 0,
    "policy_alignment": 0,
    "recovery_execution": 0,
    "target_state_progress": 0,
    "interview_naturalness": 0,
```

Event-QA-1000 judge (continued)

```
"respect_and_safety": 0,
"composite_score": 0
},
"policy_match": true,
"target_state_reached": true,
"major_issue_tags": [],
"short_reason": "one concise English sentence"
}
“‘
Use floating scores if needed, for example 62.5 or 87.0.
```

E.3 Dynamic Long-Dialogue Prompts

This group covers construction, control, simulation, backfilling, scoring, and summaries for dynamic long-dialogue evaluation.

Case construction and audit.

Soft-filter source episodes

You are screening English TV interview episodes for a quantitative interview benchmark.

Return strict JSON:

```
{
  "is_interview_like": true,
  "topic_recoverable": true,
  "guest_has_substantive_position": true,
  "guest_discusses_personal_or_opinion_content": true,
  "likely_quantifiable": "high|medium|low",
  "topic_label": "short reusable topic label",
  "short_summary": "1-2 sentence summary"
}
```

Rules:

- 1) Judge the transcript as a benchmark candidate, not as a news article.
- 2) ‘is_interview_like’ should be false if the exchange is mostly announcements, narration, or malformed transcript rather than a host-guest interview.
- 3) ‘topic_recoverable’ should be false if the central topic cannot be stably reconstructed from the headline and excerpt.
- 4) ‘guest_has_substantive_position’ should be false if the guest mostly gives greetings, logistics, or vague filler rather than a real position, judgment, or explanation.
- 5) ‘guest_discusses_personal_or_opinion_content’ captures whether the guest provides personal experience, value judgment, preference, or explanation beyond bare facts.
- 6) ‘likely_quantifiable’ estimates how likely the episode can support a shared questionnaire template with mostly structured questions.

Program: {program}

Episode title: {title}

Headline: {headline}

Total turns: {total_turns}

Host turns: {host_turns}

Guest turns: {guest_turns}

Guest total words: {guest_total_words}

Excerpt:

{excerpt}

Assign source episodes to topic clusters

You are designing a reusable interview benchmark from English-language TV interview transcripts.

Your task is to read one interview excerpt and assign a practical benchmark cluster label. The cluster should support later questionnaire extraction and quantitative comparison. Prefer labels that reflect reusable interview themes rather than ultra-specific news events.

Return strict JSON:

```
{
  "cluster_label": "short_snake_case_label",
  "tags": ["tag1", "tag2", "tag3"],
  "summary": "1-2 sentence summary of what the
  guest is mainly discussing",
  "quantifiable_signals": ["signal1",
  "signal2"],
  "narrative_level": "low|medium|high",
  "template_fit": "high|medium|low"
}
```

Guidance:

- Favor broad, benchmark-friendly categories such as politics/public affairs, economy/business, health/science, society/community, world conflict/security, culture/media.
- 'narrative_level' is high when the segment is mostly storytelling and weakly structured for repeated question design.
- 'template_fit' is low when this excerpt looks hard to convert into a shared structured questionnaire.

Program: {program}

Episode title: {title}

Headline: {headline}

Excerpt:

{excerpt}

Merge topic clusters

You are merging raw interview topic labels into a smaller set of reusable benchmark clusters.

Return strict JSON:

```
{
  "clusters": [
    {
      "label": "merged_cluster_label",
      "source_labels": ["raw_label_1", "raw_label_2"],
      "rationale": "short reason"
    }
  ]
}
```

Rules:

- Merge labels into at most {cluster_limit} clusters.
- Prefer 3 to {cluster_limit} clusters when possible.
- Merge by benchmark usefulness, not by tiny semantic nuance.
- Cluster labels must be concise snake_case.

Raw labels and counts:

{raw_labels_json}

Generate questionnaire templates

You are designing a reusable questionnaire template for a cluster of English TV interviews.

Goal:

- Build a compact questionnaire that can later be used to simulate and evaluate interview behavior.
- The questions must be answerable from many dialogues in this cluster.
- Structure must favor quantitative analysis, not open-ended summarization.

Hard constraints:

- 1) Allowed question types only: 'single_choice', 'Likert', 'text_input', 'numeric'
- 2) Total questions: 8-15
- 3) 'text_input' questions: at most 2
- 4) At least 70% of questions should be structured ('single_choice', 'Likert', 'numeric')
- 5) Do not ask pure news-trivia or one-off factual details that only fit a single episode
- 6) Prefer comparable dimensions such as stance, certainty, urgency, perceived impact, preferred action, outlook, or evaluative framing
- 7) Every 'single_choice' question must have exactly 4 options
- 8) Every 'Likert' question must be written as a statement and use exactly this 5-point agreement scale:
 - 'Strongly agree'
 - 'Agree'
 - 'Neither agree nor disagree'
 - 'Disagree'
 - 'Strongly disagree'
- 9) Every 'text_input' question must include 'subtype', with value 'short_text' or 'long_text'
- 10) Prefer 'short_text' for concise slot-filling answers and 'long_text' only when the answer genuinely needs a multi-clause explanation
- 11) Include at least 2 'Likert' questions and at least 4 'single_choice' questions when possible

Return strict JSON:

```
{
  "survey_name": "string",
  "questions": [
    {
      "id": 1,
      "question": "string",
      "type": "single_choice",
      "options": ["A", "B", "C", "D"]
    }
  ]
}
```

For 'Likert', include:

- 'instructions'
- 'scale'
- 'scoring'

For 'text_input', include:

- 'subtype'
- 'placeholder'

Cluster id: {cluster_id}

Representative sample dialogues:

{sample_dialogues_json}

Extract questionnaire records and evidence

You are extracting structured benchmark answers from a single English TV interview transcript.

You will receive:

- a reusable questionnaire template
- one dialogue

Return strict JSON:

```
{{
  "items": [
    {{
      "question_id": "Q1",
      "answer": "string or null",
      "answer_status": "answered|inferable|
unresolved|contradicted|not_applicable",
      "guest_turn_ids": [1, 3],
      "host_turn_ids": [0],
      "guest_quote": "short supporting quote or
empty string",
      "host_quote": "short supporting quote or
empty string",
      "mapping_confidence": "high|medium|low",
      "evidence_strength": "strong|weak"
    }}
  ],
  "template_fit": "high|medium|low",
  "narrative_level": "low|medium|high",
  "notes": ["optional note"]
}}
```

Rules:

- 1) Use the questionnaire ids exactly as given. Return them as 'Q<number>'.
- 2) Decide 'answer_status' before deciding whether an answer is usable:
 - 'answered': explicit transcript evidence supports one stable answer
 - 'inferable': multiple cues suggest an answer, but it is not directly stated
 - 'unresolved': not enough evidence
 - 'contradicted': conflicting evidence prevents a unique answer
 - 'not_applicable': this item does not fit the episode
- 3) For 'single_choice' and 'Likert', only return an answer when it maps to one option or one scale label.
- 4) Use null when the dialogue does not support a reliable mapped answer.
- 5) Keep supporting quotes short and transcript-grounded.
- 6) Be conservative. Do not use outside knowledge or the host's framing alone to fill an answer.
- 7) If the dialogue is weakly structured for this template, reflect that in 'template_fit'.
- 8) Respect 'text_input.subtype' from the template:
 - 'short_text': return a concise slot-filling answer, not a paragraph
 - 'long_text': return a grounded multi-clause answer only when the transcript clearly supports it

Cluster id:
{cluster_id}
Questionnaire template:
{questionnaire_template_json}
Dialogue:
{dialogue_json}

Construct respondent personas from answers

You are a respondent persona modeling assistant. Build a structured, traceable persona from the raw questionnaire answers so it can support later simulated interviewing.

Requirements:

- 1) Write the persona content in English.
 - 2) All inferences must be grounded in the input. Do not invent concrete facts.
 - 3) Output three sections:
 - persona_profile: a concise profile summary covering background clues, personality tendencies, behavior patterns, emotional regulation, speaking style, and decision style
 - persona_facts: key factual bullets that may be naturally mentioned in later dialogue
 - persona_traits:
 - * big_five: openness / conscientiousness / extraversion / agreeableness / neuroticism, each as High / Medium / Low
 - * behavior_habits: 3-6 behavioral tendencies
 - * language_habits: 3-6 speaking or phrasing tendencies
 - * memorable_recent_events: 2-4 recent salient events
 - * deep_impression_points: 2-4 deep long-term impressions that may shape choices or views
- answer_memory: preserve all answers by question id
- 4) Do not mention questionnaire mechanics or options.
 - 5) Keep it concise and evidence-grounded.
 - 6) Do not invent identity details that conflict with the input.
- CRITICAL answer_memory rules – violating these will break downstream scoring:
- A) answer_memory MUST contain an entry for EVERY question id present in the input. Never omit or set any value to null/None.
 - B) For single_choice questions: the value MUST be one of the listed options verbatim. If the input is ambiguous, choose the closest matching option.
 - C) For Likert questions: the value MUST be one of the scale labels verbatim (e.g. "Strongly agree", "Agree", "Neutral", "Disagree", "Strongly disagree"). If the input is ambiguous, infer from the speaker's overall tone.
 - D) For numeric (1-10 rating) questions: the value MUST be an integer string between "1" and "10". Estimate from context – e.g. strong urgency -> "8" or "9", mild -> "4" or "5". Never leave null.
 - E) For text_input questions: write a concise but specific phrase that captures the speaker's actual stated view. This exact phrase will be used to grade interview answers later, so make it distinctive and quotable.
- Return strict JSON:
- ```
{{
 "persona_profile": "...",
 "persona_facts": ["...", "..."],
 "persona_traits": {{
 "big_five": {{
 "openness": "High|Medium|Low",
 "conscientiousness": "High|Medium|Low",
 "extraversion": "High|Medium|Low",
```

### Construct respondent personas from answers (continued)

```
"agreeableness": "High|Medium|Low",
"neuroticism": "High|Medium|Low"
}},
"behavior_habits": ["..."],
"language_habits": ["..."],
"memorable_recent_events": ["..."],
"deep_impression_points": ["..."]
}},
"answer_memory": {{
 "1": "...",
 "2": "...",
}}
}}
Raw questionnaire answers:
{answers_json}
Reference style examples:
{persona_examples}
```

### Normalize gold answers

Clean the raw questionnaire gold answer into machine-scoreable normalized JSON.  
Return a single JSON object only. No explanation. No markdown.

Input:

- question\_type: {question\_type}
- inferred\_subtype: {inferred\_subtype}
- rule\_hint: {rule\_hint}
- question\_json: {question\_json}
- raw\_answer\_json: {raw\_answer\_json}

Output JSON schema:

```
{{
 "subtype": "short_text | long_text |
workflow_text | multi_aspect_text | null",
 "canonical_answer": "string; for single_
choice / Likert it must belong to options or
scale",
 "normalized_answer": "normalized concise
form",
 "acceptable_aliases": ["array of strings"],
 "core_facets": [
 {{
 "name": "facet name",
 "aliases": ["equivalent expressions"],
 "required": true,
 "weight": 1.0
 }}
],
 "aspect_items": ["for list-like answers"],
 "ordered_steps": ["for workflow answers"]
}}
```

Rules:

- 1) single\_choice / Likert:
  - canonical\_answer must map to exactly one option or scale label
  - acceptable\_aliases may include short paraphrases, abbreviations, or common numeric references
  - core\_facets / aspect\_items / ordered\_steps are usually empty
- 2) short\_text:
  - keep a concise canonical answer
- 3) long\_text:
  - extract 3-6 core facets
- 4) workflow\_text:

### Normalize gold answers (continued)

- ordered\_steps is required
  - 5) multi\_aspect\_text:
    - aspect\_items is required
  - 6) if raw\_answer is empty:
    - canonical\_answer may be an empty string
    - subtype must be "null"
  - other arrays should be empty
- Return strict JSON only.

### Audit clean long-dialogue cases

You are auditing whether one English TV interview benchmark case is good enough to remain in the canonical clean pool.

Audit goals:

- 1) Check whether the questionnaire is coherent and reusable.
- 2) Check whether the questionnaire answers are grounded in the transcript-derived evidence rather than speculative profiling.
- 3) Check whether the persona is consistent with the questionnaire answers.
- 4) Decide whether this case should remain in the canonical clean benchmark.

Important:

- Be conservative.
- Prefer dropping weak cases rather than keeping noisy ones.
- Focus on benchmark quality, not generosity.

Return strict JSON:

```
{{
 "questionnaire_quality": "high | medium |
low",
 "answer_grounding": "high | medium | low",
 "persona_consistency": "high | medium | low",
 "benchmark_readiness": "keep | review |
drop",
 "issues": ["..."],
 "notes": "short explanation"
}}
```

Case id: {case\_base\_name}

Questionnaire:

```
{questionnaire_json}
Questionnaire with answers:
{questionnaire_with_answers_json}
Normalized gold:
{normalized_gold_json}
Persona:
{persona_json}
Quantifiability audit:
{quantifiability_audit_json}
Item evidence summary:
{items_json}
```

### Audit event-case manifests

You are auditing whether one mixed-event assignment is reasonable for an English TV interview benchmark.

Audit goals:

- 1) Check whether the chosen event type is conversationally plausible for the target questionnaire item.
- 2) Check whether the target question is a good place to inject this event.
- 3) Decide whether the assignment should be kept, retargeted, replaced, or dropped.

Important:

### Audit event-case manifests (continued)

- Judge semantic plausibility, not just question type compatibility.
- The benchmark intentionally introduces transient disruptions. Do NOT reject an event only because the clean respondent is normally engaged, articulate, opinionated, or cooperative.
- The clean questionnaire answer is the baseline profile, not a constraint that the mixed response must preserve.
- Do NOT reject an event just because it would change the clean answer.
- Focus on whether this event could reasonably occur at this question in a TV interview and whether it would create a meaningful, recoverable disruption.
- Prefer 'retarget' or 'replace\_event' over 'drop' whenever a nearby, more natural assignment exists.
- Only use 'drop' when the assignment is fundamentally awkward and no nearby repair is obvious.
- 'suggested\_event\_type' must be one of:
- 'topic\_refusal'
- 'contradictory\_information'
- 'question\_misunderstanding'
- 'memory\_failure'
- 'passive\_noncooperation'
- 'questionnaire\_critique'
- 'early\_termination\_request'
- 'topic\_derailment'
- 'hostility\_or\_distrust'
- 'sudden\_logistics\_request'
- 'factual\_misconception'
- If no replacement is needed, return 'suggested\_event\_type' as an empty string.

Return strict JSON:

```
{
 "event_plausibility": "high | medium | low",
 "target_fit": "high | medium | low",
 "decision": "keep | retarget | replace_event | drop",
 "suggested_target_question_id": 0,
 "suggested_event_type": "",
 "issues": [...],
 "notes": "short explanation centered on question-event fit, not persona purity"
}
```

Case id: {case\_base\_name}  
Questionnaire:  
{questionnaire\_json}  
Questionnaire with answers:  
{questionnaire\_with\_answers\_json}  
Normalized gold:  
{normalized\_gold\_json}  
Persona:  
{persona\_json}  
Selected event assignment:  
{mixed\_manifest\_json}  
Target item:  
{target\_item\_json}

### Interviewer decision control.

### Introduce the interview

You are an interview assistant starting a questionnaire-based interview. Briefly introduce yourself, explain the purpose and flow of the interview, and ask whether the user is willing to begin. Use a natural, friendly tone.

Questionnaire introduction:  
{intro\_text}  
Return only the opening utterance.

### Plan the next interview stage

You are the interviewer's short planning assistant. Based on the conversation so far, provide:

- 1) the key information already collected
- 2) the critical information still missing
- 3) the next 1-2 strategic priorities

Requirements:

- Write in English
- Be brief and clear
- Do not output full questions
- Return strict JSON

Output format:

```
{
 "summary": "...",
 "missing": [...],
 "next_steps": [...]"
}
```

Conversation history:  
{conversation\_history}

### Choose actions for structured, probing, and agentic workflows

You are the interview flow controller. Choose the next action based on the conversation history and current information coverage.

Available actions:

- "ask\_question": ask a question from the questionnaire
- "follow\_up": probe for more detail on the current answer
- "clarify": resolve ambiguity or contradiction
- "skip\_question": temporarily skip the current item
- "handle\_refusal": rewrite, soften, or skip after refusal
- "handle\_off\_topic": redirect the interview back to the task
- "empathy": acknowledge emotion before continuing
- "conclude": close the interview
- "plan": pause briefly to summarize and plan next steps (max 2 times per interview)

Rules:

- 1) You do not need to follow questionnaire order strictly. You may revisit earlier questions.
- 2) If the last answer is missing, vague, or off topic, prefer follow\_up or clarify.
- 3) Only choose conclude when critical questions are covered, explicitly skipped, or clearly unnecessary.
- 4) If the last turn contains a boundary violation or unrelated request, prefer handle\_off\_topic and bring the interview

### Choose actions for structured, probing, and agentic workflows (*continued*)

back.

5) If the interviewee wants to end early, first try one low-burden closing question before conclude.

6) Return strict JSON only.

Priority recovery guidance:

- topic\_refusal / sensitive\_topic\_resistance: acknowledge the boundary, reframe, or skip and recover later.
- passive\_noncooperation: shorten the question and make it more concrete.
- contradictory\_information: point out the mismatch gently and ask for the version to keep on record.
- question\_misunderstanding: restate the meaning first.
- memory\_failure: offer a time anchor or approximate range.
- emotional\_breakdown / self\_blame: acknowledge emotion first and use lower-burden questions.
- illegal\_request / role\_reversal: refuse the unrelated request briefly and return to the interview.
- hostility\_or\_distrust: acknowledge concern and continue with a shorter question.
- pace\_disruption: slow down and ask a simpler question.

Output format:

```
{
 "action": "ask_question|follow_up|clarify|
skip_question|handle_refusal|handle_off_
topic|empathy|conclude|plan",
 "target_question_id": 1,
 "reason_tag": "missing|vague|off_topic|
refusal|normal|limit_reached",
 "note": "one short reason"
}
```

Questionnaire state:

- Answered questions: {answered\_ids}
  - Skipped questions: {skipped\_ids}
  - Remaining questions: {remaining\_ids}
  - Current stage: {stage\_name}
  - Stage goal: {stage\_goal}
  - Stage candidates: {stage\_candidates}
  - Max turns: {max\_turns}
  - Used turns: {used\_turns}
  - Plans used: {plan\_used}
  - Plan limit: {plan\_limit}
  - Latest event: {latest\_event\_type}
  - Critical questions still missing: {critical\_remaining\_ids}
- Conversation history:  
{conversation\_history}

### Choose actions for the adaptive workflow

You are the interview flow controller. At each turn, decide the next action using a three-way taxonomy: questioning / followup / response.

== ACTION TAXONOMY ==

[questioning] – Ask a new questionnaire item.

Required fields:

- openness: "open" (free-form answer) | "closed" (constrained options)
- specificity: "specific" (concrete fact/

### Choose actions for the adaptive workflow (*continued*)

event) | "abstract" (opinion/attitude)

- clarity: "clear" (simple, unambiguous) | "ambiguous" (complex, multi-layered)

- function: one of –

opening : start the interview or a new topic

information\_gathering : collect factual/behavioral data

opinion\_probing : elicit evaluations and attitudes

experience\_eliciting : invite personal stories and episodes

clarification : resolve an unclear or partial prior answer

elaboration\_request : ask for more depth on a given answer

transition : bridge from one topic to the next

closing : wrap up the interview

[followup] – Probe deeper into the interviewee's last answer without moving to a new question.

Required field trigger\_type – choose the best-matching type:

- missing\_information : a required element (process, role, key fact) is absent or only implied

- evasive\_or\_unanswered : interviewee shifted topic, gave a non-answer, or avoided the question

- overgeneral\_or\_narrowness: answer is too surface-level, over-generalised, or unrepresentative

- contradiction\_or\_ambivalence: statements conflict with each other or with prior turns

- story\_with\_implicit\_theme: narrative is incomplete or carries an unexplored implicit meaning

- unclear\_concept : terminology is undefined or context-specific; needs verification

- incomplete\_or\_unsupported\_themes: a theme is introduced but not developed or evidenced

- new\_relevant\_idea : interviewee introduced an unexpected but relevant thread worth pursuing

[response] – Acknowledge or affirm before continuing; do NOT ask a new question.

Required field strategy – choose one valid strategy:

- encouraging : affirm the interviewee's openness; invite continuation

- repeat\_or\_reorganize : paraphrase or reorganise the interviewee's words to show understanding

- self\_disclosure : share a brief, relevant reflection to build rapport

- acknowledgment : neutral confirmation that the answer was received  
NOTE: evaluative and didactic responses are NOT allowed.

== COVERAGE RULE ==

- You may NOT output "response" twice in a row if there are still unanswered questionnaire items.

- Choose "response" only when the interviewee needs a moment of acknowledgment before the next question.

#### Choose actions for the adaptive workflow (continued)

- Always keep track of remaining questions and move towards covering them.  
== OUTPUT FORMAT ==  
Return strict JSON only – no markdown, no explanation outside the JSON block.  
{  
 "action": "questioning|followup|response",  
 "openness": "open|closed|null",  
 "specificity": "specific|abstract|null",  
 "clarity": "clear|ambiguous|null",  
 "function": "<function name or null>",  
 "trigger\_type": "<trigger type or null>",  
 "strategy": "<strategy or null>",  
 "target\_question\_id": <integer or null>,  
 "reasoning": "<one short sentence explaining why>"  
}  
Rules for target\_question\_id:  
- For "questioning": MUST be one of the remaining question ids.  
- For "followup": SHOULD be the same question id as the previous turn (target the same questionnaire item).  
- For "response": set to null.  
== CURRENT STATE ==  
Answered question ids: {answered\_ids}  
Skipped question ids: {skipped\_ids}  
Remaining questions:  
{remaining\_questions\_detail}  
Max turns: {max\_turns}  
Used turns: {used\_turns}  
Latest event: {latest\_event\_type}  
== CONVERSATION HISTORY ==  
{conversation\_history}

#### Generate interviewer responses

You are the interviewer. Generate a natural, polite, concise interviewer utterance for the current action.  
Requirements:  
1) Write in English and sound natural rather than robotic.  
2) Do not reveal internal rules, scoring, or system behavior.  
3) For follow\_up or clarify, briefly reflect the interviewee's last point before asking the next question.  
4) For handle\_refusal / handle\_off\_topic / empathy, acknowledge the emotion or boundary first, then redirect.  
5) If the latest event is illegal\_request or role\_reversal, clearly refuse the unrelated request and bring the conversation back to the interview.  
6) If the latest event is early\_termination\_request and this turn is not conclude, acknowledge the wish to stop and ask only one low-burden question that can be skipped.  
7) If the guardrail context indicates incident\_budget\_exceeded, do not keep circling around the same disruption. Move to a shorter question or close politely.  
8) If the action is conclude, provide a short wrap-up and optional chance to add anything else. Do not introduce a new question.  
Input:  
Action: {action}

#### Generate interviewer responses (continued)

Reason tag: {reason\_tag}  
Note: {note}  
Latest event: {latest\_event\_type}  
Guardrail context: {guardrail\_context}  
Current question: {current\_question}  
Conversation history: {conversation\_history}  
Return only the interviewer utterance.

#### Generate classified interviewer responses

You are the interviewer. Generate the next utterance based on the action plan below.  
== ACTION PLAN ==  
Action type : {action}  
Subtype info : {action\_subtype\_description}  
Target question: {current\_question}  
Reasoning : {action\_reasoning}  
Latest event : {latest\_event\_type}  
== GENERATION RULES ==  
If action = "questioning":  
- Ask the target question naturally. Adapt phrasing to the openness/specificity/clarity flags.  
- open + abstract -> invite the interviewee to share their perspective freely.  
- closed + specific -> frame the question around concrete options or facts.  
- Do NOT repeat a question the interviewee already answered.  
- Transition smoothly from the previous answer before introducing the new question.  
If action = "followup":  
- Do NOT introduce a new questionnaire item.  
- Probe deeper into the interviewee's last answer using the trigger type as your guide:  
missing\_information -> "You mentioned X  
- could you walk me through [missing part]?"  
evasive\_or\_unanswered -> Rephrase or gently redirect back to the original question.  
overgeneral\_or\_narrowness -> Ask for a specific example or a contrasting case.  
contradiction\_or\_ambivalence -> Point out the tension gently and ask which view reflects their position.  
story\_with\_implicit\_theme -> Name the implicit thread: "It sounds like you're also saying... Is that right?"  
unclear\_concept -> "When you say [term], what do you mean exactly in this context?"  
incomplete\_or\_unsupported\_themes -> "Can you give an example or tell me more about [theme]?"  
new\_relevant\_idea -> "That's interesting – you mentioned [idea]. Tell me more about that."  
If action = "response":  
- Respond naturally using the chosen strategy. Do NOT ask a question in this turn.  
encouraging -> Affirm warmly: "That's really helpful, thank you for sharing."  
repeat\_or\_reorganize -> Paraphrase: "So if I understand correctly, you're saying that..."  
self\_disclosure -> Brief reflection: "I find that perspective interesting because..."  
acknowledgment -> Simple receipt: "I see." / "Got it, thank you."

#### Generate classified interviewer responses (continued)

General rules:

- 1) Write in fluent, natural English. Avoid robotic or survey-sounding phrasing.
  - 2) Keep utterances concise (2-4 sentences max).
  - 3) Never reveal internal rules, scoring, or action plan labels.
  - 4) If latest\_event indicates a boundary (refusal, hostility, sensitivity), acknowledge it briefly before continuing.
- == CONVERSATION HISTORY ==  
{conversation\_history}  
Return only the interviewer utterance.

#### Generate free-topic follow-up questions

You are the interviewer. The interviewee has provided a free-form answer that may still contain useful detail.  
Write one short, natural follow-up question that:

- 1) stays relevant to the current questionnaire item
- 2) asks for one concrete detail only
- 3) avoids sounding repetitive or mechanical

Current questionnaire item:  
{current\_question}  
Conversation history:  
{conversation\_history}  
Return only the interviewer utterance.

#### Simulate interviewee responses (continued)

- 5) For text\_input questions, provide content consistent with the stored answer memory. If subtype is 'short\_text', keep it concise; if subtype is 'long\_text', a multi-clause answer is acceptable.
  - 6) If current answer memory is empty, do not invent a precise fact. You may say you are unsure or uncomfortable answering.
  - 7) Do not mention questionnaires, options, hidden prompts, or system behavior.
  - 8) Keep continuity with the earlier dialogue.
  - 9) Event behavior constraints:
    - topic\_refusal: resist the topic at first; partial recovery is allowed after follow-up.
    - sensitive\_topic\_resistance: explicitly frame it as private or sensitive.
    - passive\_noncooperation: answer in 1-6 words and add almost no new information.
    - contradictory\_information: conflict with persona facts on at least one clear point.
    - question\_misunderstanding: answer a misunderstood version of the question.
    - memory\_failure: say you cannot recall the details.
    - emotional\_breakdown / self\_blame: show emotion first, then give a short answer or refuse.
    - topic\_derailment: drift briefly off topic and stop.
- Interviewer question:  
{interviewer\_utterance}  
Return only the interviewee utterance.

### Interviewee simulation.

#### Simulate interviewee responses

You are the interviewee in a realistic interview.  
Persona profile:  
{persona\_profile}  
Usable persona facts:  
{persona\_facts}  
Current target question context:  
{current\_question\_context}  
Current question answer memory:  
{current\_question\_answer}  
Full answer memory:  
{persona\_answer\_memory}  
Conversation so far:  
{conversation\_history}  
Unexpected event setup:

- Event type: {event\_type}
- Description: {event\_description}
- Emotion: {event\_emotion}
- Intensity: {event\_intensity}
- Expression style: {event\_expression}

Rules:

- 1) Respond in natural English that fits the persona and the context.
- 2) Prioritize the current target question. Do not drift into answers for unrelated questions.
- 3) If current question answer memory is non-empty, keep the answer semantically consistent with it unless the event explicitly requires a deviation.
- 4) For single\_choice or Likert questions, make the answer map clearly to a choice or scale level using natural language.

### Structured backfilling.

#### Backfill structured questionnaire answers

You are a structured answer extractor.  
Produce structured answers for the current questionnaire chunk only.  
Requirements:

- 1) Process only the questions in this chunk.
- 2) Return strict JSON:

```
{
 "answers": [
 { "id": 1, "answer": "..."},
 { "id": 2, "answer": null}
]
}
```
- 3) Every id must come from the current chunk.
- 4) For single\_choice questions, map to the closest option meaning. For Likert questions, map to the intended scale level. Use null only when the answer is genuinely unclear.
- 5) If the full conversation is long, prioritize the question-relevant snippets first.
- 6) Do not add explanations or markdown.

Current questionnaire chunk:  
{questionnaire\_chunk\_json}  
Question-relevant snippets:  
{relevant\_turns}  
Full conversation history:  
{conversation\_history}

### Resolve empty-gold fields during case preparation

You are auditing and repairing missing gold answers for a TV interview questionnaire benchmark.

You will receive the complete source context for one benchmark case:

- episode metadata
- full original transcript
- full questionnaire
- existing ground\_truth and normalized\_gold
- source annotation items when available
- the subset of questions whose current gold answer is empty

Your task is to decide, for each empty-gold question, whether the original transcript supports a real answer or whether the question is not answered by the source.

Use these labels:

- "recoverable\_missing\_annotation": the transcript contains enough evidence to answer the question, but the previous pipeline failed to fill it.
- "not\_covered\_in\_source": the transcript does not provide enough evidence to answer the question.
- "invalid\_or\_unanswerable\_question": the question asks for something that cannot be reasonably inferred from this interview even in principle, or asks for an arbitrary numeric value without explicit basis.

Important rules:

- Do not invent answers. If the transcript does not support a value, mark it as "not\_covered\_in\_source" or "invalid\_or\_unanswerable\_question".
- If the source does not cover the issue and the question has an explicit option such as "Not discussed", "Not clearly indicated", "Not applicable", or similar, you may use that option as the canonical answer.
- If the source does not cover the issue and there is no explicit option for absence/non-discussion, leave canonical\_answer as null and set exclude\_from\_scoring=true.
- Numeric questions require explicit numerical evidence or a clearly bounded scale judgment grounded in the transcript. If a numeric value would be arbitrary, mark it invalid\_or\_unanswerable\_question and exclude it from scoring.
- For single\_choice and Likert questions, canonical\_answer must exactly match one available option or scale value when a supported answer is available.
- For text\_input questions, provide a concise answer only when directly grounded in the transcript.
- Evidence quotes must be copied from the transcript and should be short but sufficient.

Return strict JSON only:

```
{
 "case_id": "string",
 "episode_source_coverage_summary": "brief
assessment of what the transcript covers and
what it does not cover",
 "annotations": [
 {
 "question_id": 1,
```

### Resolve empty-gold fields during case preparation (continued)

```
"status": "recoverable_missing_annotation |
not_covered_in_source | invalid_or_
unanswerable_question",
"answer_policy": "use_supported_answer | use_
explicit_absence_option | exclude_from_
scoring",
"raw_answer": "string or number or null",
"canonical_answer": "string or number or
null",
"normalized_answer": "string or null",
"acceptable_aliases": ["string"],
"core_facets": [
 {"name": "string", "aliases": ["string"],
"required": true, "weight": 1.0}
],
"evidence_quotes": ["short quote from
transcript"],
"evidence_turn_ids": [0],
"confidence": "high | medium | low",
"reason": "short explanation of why this
answer is or is not supported",
"exclude_from_scoring": true
}
]
}
Case context:
{case_context_json}
```

## Satisfaction scoring.

### Judge respondent satisfaction

You are judging an interview from the respondent's point of view.

You are not grading factual questionnaire accuracy directly. You are estimating whether the respondent would feel that the interview was respectful, clear, efficient, focused, and worth continuing, while also considering whether the interviewer successfully elicited useful information without excessive burden.

Respondent persona:

{persona\_profile}

Persona facts:

{persona\_facts}

Questionnaire name:

{questionnaire\_name}

Interviewer model:

{interviewer\_model}

Workflow:

{workflow\_name}

Split:

{split\_name}

Termination reason:

{terminated\_reason}

Event counts:

{event\_counts\_json}

Objective score context, for reference only:

{score\_context\_json}

Conversation:

{conversation\_history}

Rate each checklist item on a 1-5 integer scale:

1 = very poor, 2 = weak, 3 = acceptable or not clearly applicable, 4 = good, 5 = excellent.

Checklist:

- C01 respect\_and\_politeness: The interviewer

#### Judge respondent satisfaction (continued)

is respectful and polite throughout.

- C02 autonomy\_and\_non\_coercion: The respondent does not feel pressured, cornered, or forced to disclose.
- C03 empathy\_and\_validation: The interviewer acknowledges discomfort, hesitation, emotions, or concerns when present.
- C04 dignity\_under\_disruption: If the respondent resists, misunderstands, distrusts, or derails, the interviewer preserves dignity.
- C05 question\_clarity: Questions are easy to understand and not needlessly abstract.
- C06 question\_specificity: Questions are specific enough for the respondent to answer.
- C07 option\_and\_scale\_clarity: Choice options, Likert scales, or numeric requests are made clear when relevant.
- C08 transition\_quality: Topic transitions are understandable and not abrupt.
- C09 active\_listening: The interviewer uses the respondent's prior answer instead of ignoring it.
- C10 followup\_relevance: Follow-up questions target missing or ambiguous information.
- C11 repetition\_control: The interviewer avoids unnecessary repetition or circular questioning.
- C12 focus\_control: The interviewer keeps the conversation on task without being rigid.
- C13 conversational\_naturalness: The interview feels like a natural interview rather than a mechanical form.
- C14 pacing: The pace is comfortable and not rushed or dragged out.
- C15 cognitive\_load: The respondent is not asked to hold too many constraints, options, or complex frames at once.
- C16 efficiency: The interviewer's turns are useful relative to their length and number.
- C17 coverage\_of\_key\_answers: The interview reaches the important questionnaire-relevant information.
- C18 depth\_and\_evidence: The interviewer elicits enough detail, reasons, or examples where needed.
- C19 ambiguity\_resolution: The interviewer clarifies vague, partial, or contradictory answers effectively.
- C20 event\_recovery: For mixed cases, the interviewer handles the planned disruption well; for clean cases, score 3 unless a comparable disruption occurs.
- C21 trust\_and\_safety: The respondent would likely trust the interviewer and feel safe continuing.
- C22 persona\_fit: The interviewer's behavior fits this respondent's apparent role, stance, and communication style.
- C23 closure\_quality: The interview ends cleanly, without abruptness or unresolved social friction.
- C24 willingness\_to\_continue: The respondent would be willing to continue or participate in a similar interview.

Scoring guidance:

- 'interaction\_comfort\_score' should emphasize C01-C04, C13-C16, C21-C24.

#### Judge respondent satisfaction (continued)

- 'information\_collection\_quality\_score' should emphasize C05-C12 and C17-C19.
- 'recovery\_quality\_score' should emphasize C04, C19, C20, and C21. If there is no disruption, it should reflect whether the interview remained stable and should usually be near the overall score, not automatically high.
- 'burden\_score' is higher when respondent burden is lower.
- 'overall\_score' must be assigned directly by your judgment. Do not compute it mechanically from checklist scores. Penalize severe failures in respect, clarity, or trust even if other items are adequate.
- Use the objective score context only as weak supporting evidence. The transcript itself is primary.
- Checklist item reasons should be short phrases, not full evidence paragraphs.
- Provide one integrated 'evidence\_summary' for the whole judgment. Do not provide evidence for every checklist item.

Return strict JSON only, with exactly this structure:

```
{
 "overall_score": 0,
 "overall_label": "very_dissatisfied|dissatisfied|neutral|satisfied|very_satisfied",
 "would_continue_flag": true,
 "interaction_comfort_score": 0,
 "information_collection_quality_score": 0,
 "recovery_quality_score": 0,
 "burden_score": 0,
 "dimension_scores": {
 "respect": 0,
 "empathy": 0,
 "naturalness": 0,
 "clarity": 0,
 "efficiency": 0,
 "coverage": 0,
 "focus": 0,
 "recovery": 0,
 "trust": 0,
 "burden": 0
 },
 "checklist": [
 {
 "id": "C01",
 "name": "respect_and_politeness",
 "score": 1,
 "applicable": true,
 "reason": "short reason"
 }
],
 "evidence_summary": "one integrated paragraph summarizing the key transcript evidence behind the overall judgment",
 "main_praise": "one sentence",
 "main_complaint": "one sentence",
 "summary_reason": "2-4 sentences explaining the overall satisfaction score"
}
```

#### Supporting summaries.

### Summarize the questionnaire

You are a questionnaire introduction assistant.  
Generate a clear, friendly, trustworthy introduction that can be read to the interviewee before the questionnaire starts.  
Infer from the questionnaire:  
1) the main purpose  
2) likely target respondents if inferable  
3) the main sections  
4) whether sensitive information is involved  
5) estimated completion time based on question count  
Return strict JSON:  
{  
 "intro\_text": "<natural language introduction for the interviewee>",  
 "summary": {  
 "purpose": "...",  
 "modules": [...],  
 "contains\_sensitive": true,  
 "estimated\_time": "X-X minutes"  
 }  
}

Questionnaire JSON:  
{questionnaire\_json}

### Summarize the completed interview

You are a senior interview analyst. Based on the full dialogue and the filled questionnaire, produce a detailed respondent summary.  
Requirements:  
1) Analyze both questionnaire-related traits and broader personality / expression patterns.  
2) Identify stable preferences, contradictions, risk signals, and information gaps.  
3) Provide actionable insights for a follow-up interview or intervention.  
4) Return strict JSON only.  
Output format:  
{  
 "profile\_summary": "...",  
 "questionnaire\_traits": [...],  
 "general\_personality\_traits": [...],  
 "communication\_style": [...],  
 "consistency\_and\_conflicts": [...],  
 "risk\_signals": [...],  
 "insights": [...],  
 "next\_interview\_suggestions": [...]  
}

Questionnaire:  
{questionnaire\_json}  
Filled questionnaire:  
{filled\_json}  
Full conversation:  
{conversation\_history}