RESQ: BENCHMARKING QUESTIONING, ANSWERING, AND RECOVERY IN EMBODIED AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Embodied agents inevitably encounter uncertainty during long-horizon task execution, requiring them not only to seek missing information but also to use it effectively to recover. We introduce Recovery through Selective Questioning (RESQ), a benchmark built from actual failures of autonomous LLM agents for studying the complete interaction loop of questioning, answering, and recovery. RESQ-Q contains 2,160 failure-grounded question demonstrations from 678 failed ALFRED trajectories, specifying when, what, and why to ask. RESQ-A provides 6,480 independently collected human responses and supports a fixed response oracle for scalable and reproducible interaction. RESQ-R provides answer-conditioned recovery trajectories demonstrating how agents revise their reasoning and actions after receiving information. Across diverse LLM families and scales, question supervision generally improves task completion, with recovery supervision providing further gains. Our selective-questioning analysis reveals distinct asking regimes across models, showing how beneficial and harmful interactions jointly shape task success. We further find that interaction effectiveness depends on question timing and response source. RESQ provides a unified testbed for learning agents that seek help selectively and recover effectively.

1 INTRODUCTION

Recent advances in large language models (LLMs) have enabled increasingly capable instruction-following embodied agents, which must interpret natural-language instructions, plan over long horizons, and continually decide what to do next under partial observability (Ichter et al., 2023; Song et al., 2023; Huang et al., 2023). Despite this progress, agents inevitably encounter uncertainty. They may fail to locate a target object, face ambiguity in the instruction, or become unsure how to proceed. When such uncertainty cannot be resolved from their own observations or interaction history, seeking information from the user provides a natural way to continue execution.

A systematic study of agents that seek information from users requires addressing several tightly connected challenges. Agents should operate independently by default and ask questions selectively, since unnecessary interaction burdens the user and undermines autonomy. At the same time, open-ended questions require a scalable respondent for reproducible evaluation, and agents must use the acquired information to guide subsequent decisions and actions. This calls for a benchmark with linked supervision and evaluation across questioning, answering, and subsequent task execution.

The first challenge is obtaining supervision for selective questioning. Existing approaches typically derive questions from predefined clarification or ambiguity structures (Gao et al., 2022; Ivanova et al., 2025), or determine when to seek help using uncertainty (Ren et al., 2023; Liang et al., 2024; Park et al., 2024). However, these predefined interaction structures and uncertainty criteria do not necessarily reflect what information an autonomous agent needs when its execution is at risk of failure. Grounding supervision in actual failure trajectories provides the context needed to identify both when to ask and what information to request to redirect execution.

The second challenge is providing scalable responses to open-ended agent questions. Existing approaches rely on prerecorded or templated responses, human users, or prompted LLMs, with the latter offering a scalable alternative for online interaction (Dai et al., 2024; Ramrakhya et al., 2025). However, simply prompting a general-purpose LLM to act as a user does not ensure that its responses reflect how humans select and scope the information they provide, consistent with recent

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

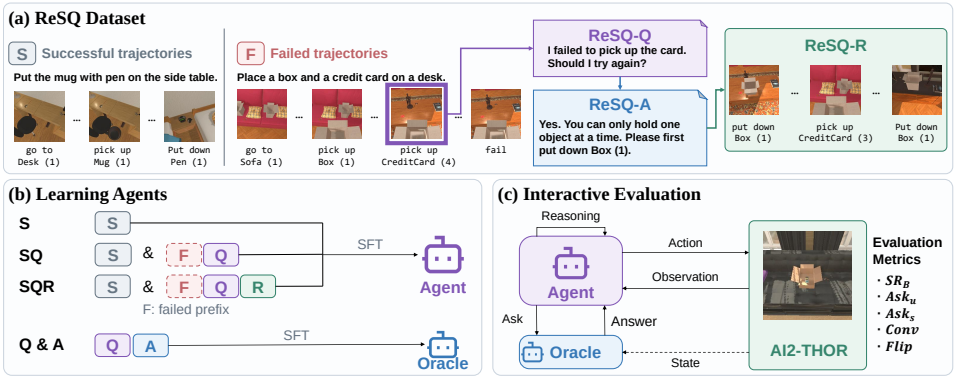

Figure 1: Overview of RESQ: (a) failure-grounded collection of questions, human responses, and recovery trajectories; (b) agent and response-oracle training; and (c) interactive evaluation of task success and question use.

findings on LLM-based user simulation (Naous et al., 2026). Because such differences can affect downstream agent behavior and evaluation outcomes, interactive evaluation requires a scalable and reproducible respondent grounded in human response data.

The third challenge is using the acquired information to redirect task execution. Obtaining a useful answer does not by itself determine how an agent should revise its plan or act from its current state. An agent may therefore ask an appropriate question and receive the information it needs, yet still fail to complete the task. Learning answer-conditioned recovery requires supervision that connects question-answer interactions to the subsequent reasoning and actions needed to resume execution.

To address these challenges, we introduce Recovery through Selective Questioning (RESQ), a benchmark for studying questioning, answering, and recovery in embodied instruction following. Figure 1 summarizes the overall RESQ framework. Building on ALFRED (Shridhar et al., 2020), we first collect successful and failed trajectories from LLM agents to capture failures that naturally arise during task execution. From these failures, 34 human annotators provide 2,160 question demonstrations specifying when to ask, what to ask, and why the information is needed, forming RESQ-Q. Three independent annotators then answer all 2,160 questions, yielding 6,480 human responses that constitute RESQ-A and provide multi-reference supervision for training a fixed response oracle for scalable and reproducible interaction. Finally, RESQ-R provides answer-conditioned recovery trajectories demonstrating how acquired information can guide subsequent reasoning and actions. Together, these components provide linked supervision for failure-grounded selective questioning, human-grounded answering, and answer-conditioned recovery within a unified benchmark.

Our experiments reveal three key findings. Failure-grounded question supervision generally improves task success across model families and scales; a small question budget captures most of the benefit, while learned and uncertainty-based timing generally outperforms random timing. Answer-conditioned recovery supervision provides further gains, showing that learning when and what to ask is distinct from learning how to use the resulting answer. Finally, interactive evaluation is sensitive to the respondent: our human-response-trained oracle more closely reproduces measured human response characteristics, while prompted frontier LLMs select and scope information differently, affecting both task success and comparisons between candidate agents.

2 Related Work

Question Asking and Help Seeking. Prior work has learned clarification behavior from human interaction data. DIALFRED (Gao et al., 2022) and TEACH (Padmakumar et al., 2022) collect clarification questions or dialogue alongside task execution, while other datasets focus on predefined or intentionally unclear instructions (Ivanova et al., 2025; Wang et al., 2025). Building on such interactive settings, ELBA learns when and what questions to ask during embodied task execution (Shen et al., 2025), while ASK-TO-ACT learns a policy to ask for clarification and use the answers to complete intentionally underspecified tasks (Ramrakhya et al., 2025). These approaches

demonstrate the value of learning from interaction data, but their supervision originates from human dialogue, predefined interaction structures, or ambiguities introduced during task construction. In contrast, RESQ derives question supervision from actual failures of autonomous LLM agents, using human annotations to identify when to ask and what information to request for recovery.

A complementary line of work uses uncertainty to determine when external assistance is needed, including conformal prediction over candidate actions, ambiguity or infeasibility detection, and uncertainty over candidate plans (Ren et al., 2023; Park et al., 2024; Liang et al., 2024). Han et al. (2024) estimate uncertainty from token-level probabilities of free-form LLM outputs, while Taioli et al. (2025) use uncertainty-aware interaction triggers in embodied navigation. Rather than adopting a particular uncertainty formulation, RESQ-Q provides failure-grounded demonstrations for learning when to ask directly, enabling controlled comparisons with uncertainty-based and random triggers while holding question generation and the respondent fixed.

Response Oracles and Recovery. Compared with question asking, response modeling and post-answer recovery have received less attention as joint components of embodied-agent benchmarks. Existing studies answer agent questions using prerecorded or templated responses, human users, or prompted LLMs (Gao et al., 2022; Padmakumar et al., 2022; Dai et al., 2024). Prompted LLMs enable scalable interaction but can differ substantially from human users, motivating simulators grounded in human data (Naous et al., 2026; Zhu et al., 2026). Separately, interactive agents use clarification answers during task execution (Ramrakhya et al., 2025), while recovery-oriented methods learn recovery behaviors from failure regions or perturbed trajectories (Xiao et al., 2026; Zhao et al., 2026). These directions are typically studied separately rather than as a unified post-question process linking human-grounded responses to the subsequent reasoning and actions needed for recovery. RESQ-A and RESQ-R connect these stages within a unified benchmark.

3 RESQ BENCHMARK SUITE

3.1 BENCHMARK SETTING

We consider a partially observable embodied instruction-following setting in which an LLM agent receives a natural-language instruction l and must complete the task through sequential decision making. Following the ReAct paradigm (Yao et al., 2023), our interaction interface interleaves reasoning with environment interaction and adds an explicit asking decision for information seeking. At each step t, the agent produces a high-level decision $a_t \in \mathcal{A}$, where $\mathcal{A} = \mathcal{A}^{\text{reason}} \cup \mathcal{A}^{\text{act}} \cup \mathcal{A}^{\text{ask}}$, corresponding to reasoning, acting in the environment, or asking a question. The agent maintains an interaction history $h_t = (o_0, a_0, o_1, a_1, \ldots, a_{t-1}, o_t)$, where o_0 is the initial observation and o_{t+1} is the observation, if any, following a_t. Conditioned on the task instruction and interaction history, the agent generates $a_t \sim \pi_\theta(\cdot \mid l, h_t)$, where π_θ denotes the LLM policy.

A reasoning decision $a_t \in \mathcal{A}^{\text{reason}}$ produces a natural-language intermediate reasoning step but yields no new observation, denoted by $o_{t+1} = \phi$. An environment action $a_t \in \mathcal{A}^{\text{act}}$ is generated as a free-form natural-language command that invokes one of a predefined set of executable skills, with the resulting environment feedback returned as o_{t+1}. For an asking decision $a_t \in \mathcal{A}^{\text{ask}}$, the agent asks the user a free-form natural-language question, and the response is returned as o_{t+1}. Multiple questions may be issued within an episode, subject to an interaction budget specified by the evaluation protocol. For scalable and reproducible evaluation, RESQ provides a learned response oracle trained on RESQ-A to simulate user responses.

3.2 DATASET CONSTRUCTION

Autonomous agent trajectories. We construct RESQ on ALFRED (Shridhar et al., 2020), a benchmark for long-horizon household instruction following in which agents navigate and manipulate objects in the AI2-THOR simulator (Kolve et al., 2017). We follow the embodied interaction setup of REACTREE (Choi et al., 2026), in which the agent interacts through executable skills such as go to, pick up, put down, open, close, turn on/off, and slice, and receives action-dependent textual feedback indicating success or failure and the objects and receptacles observable after each action.

To collect autonomous task-execution trajectories, we first manually collect three successful demonstrations for each ALFRED task type. A LLaMA-3.1 70B (Grattafiori et al., 2024) ReAct agent re-

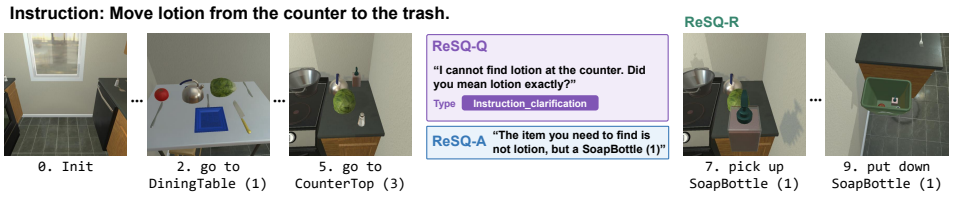

Figure 2: A representative RESQ instance linking a failed execution prefix to a failure-grounded question (RESQ-Q), a human response (RESQ-A), and answer-conditioned recovery (RESQ-R).

trieves seed demonstrations by instruction similarity using Sentence-BERT (Reimers & Gurevych, 2019) within a 5K-token budget and uses them as in-context demonstrations. Running the agent over the ALFRED training set yields 1,530 successful and 8,288 failed trajectories. Successful trajectories provide supervision for autonomous task completion, while failed trajectories serve as execution contexts for collecting failure-grounded questions in RESQ-Q. Figure 2 illustrates a representative failed trajectory extended with RESQ-Q, RESQ-A, and RESQ-R.

RESQ-Q: Failure-grounded questions. RESQ-Q provides failure-grounded supervision for selective questioning. From the 8,288 failed trajectories, we uniformly sample 700 across the seven ALFRED task types, with 100 trajectories per type, for human annotation. For each trajectory, annotators propose at least three questions that could help redirect execution toward successful task completion and specify when to ask (the trajectory step after which the question should be issued), what to ask (the free-form question), and why to ask (the rationale for seeking the information). Annotators inspect each trajectory step by step through visualizations of the agent's actions, with the complete textual trajectory available for reference. We recruit 35 participants, one of whom is used for piloting, leaving 34 annotators for the main collection. The resulting RESQ-Q contains 2,160 questions grounded in 678 failed trajectories. Data collection is conducted under institutional IRB approval. Most annotations are written in Korean and translated into English using GPT-4o (Hurst et al., 2024), with the translation prompt provided in Appendix A.1.

To characterize the information needs underlying the collected questions, we inductively derive six uncertainty types: object location, object identification, planning and decision, instruction clarification, state and memory, and failure cause. Rather than categorizing questions by surface form, the taxonomy captures the source of information sought by the agent. We obtain initial labels with GPT-4o and manually review all assignments; the initial labels exactly match the final human-reviewed labels for 1,950 of 2,160 questions (90.3%). Object location (32.1%), object identification (28.4%), and planning and decision (22.0%) are the three most common types, together accounting for 82.6% of the resulting labels. Instruction clarification (9.6%), state and memory (4.0%), and failure cause (3.9%) occur less frequently; category definitions and labeling details are provided in Appendix A.2.

RESQ-A: Human responses and response oracle. To collect human responses to the failure-grounded questions in RESQ-Q, we recruit three annotators who are independent of the question annotators. Each annotator independently answers all 2,160 questions, yielding three responses per question and 6,480 responses in total. For each question, we reset the corresponding ALFRED task and replay the original failed trajectory up to the annotated question point. Annotators are provided with the same task instruction and interaction history available to the agent, together with privileged simulator information needed to answer the question. This privileged context includes the structured task goal, ground-truth progress toward that goal, the locations of goal-relevant objects including those hidden from the agent, and the contents of all receptacles. When this information is insufficient, annotators may further inspect the simulator using the same natural-language environment-action interface available to the agent.

Rather than selecting or consolidating the three responses into a single canonical answer, we retain all responses to preserve variation in wording, specificity, and information scope. The resulting 6,480 responses constitute RESQ-A and provide multi-reference supervision for learning a response oracle. We train a fixed oracle to generate a natural-language response conditioned on an agent's free-form question and the same task, interaction, and privileged simulator context provided to the annotators. We fine-tune the oracle with LoRA (Hu et al., 2022) using the standard autoregressive language-modeling objective, treating each human response as an independent training target. The

resulting oracle provides a scalable and reproducible respondent and is held fixed across candidate agents during evaluation.

RESQ-R: Answer-conditioned recovery. Receiving a useful answer does not by itself guarantee successful recovery: the agent must still determine how to revise its plan and use the acquired information to guide subsequent actions. RESQ-R therefore provides answer-conditioned recovery trajectories demonstrating how execution can resume after a question-answer interaction.

For recovery collection, we randomly select one of the three RESQ-A annotators and use that annotator's complete response set as a fixed human response source. For each instance, we replay the original failed trajectory up to the annotated question point, present the corresponding question and human response, and ask a recovery annotator to continue task execution from that state. Recovery annotators explicitly record natural-language reasoning steps: they first interpret the response and formulate an overall plan and next subgoal, and after completing each subgoal, summarize progress and identify the next one. They interact with the environment through the same natural-language action interface as the autonomous agent. We verify each trajectory by replaying it in the simulator and checking the final ALFRED goal conditions. For the experiments in this paper, we use 514 replay-verified successful recovery trajectories.

Data splits. All RESQ data are constructed from trajectories in the ALFRED training split. We divide RESQ-Q and the associated RESQ-A responses into training and test sets that are disjoint with respect to both source trajectories and question annotators, yielding 1,728 questions from 538 trajectories for training and 432 questions from 140 trajectories for testing. All 514 RESQ-R trajectories used in our experiments are associated with questions in the training partition. Online evaluation is conducted separately on the original ALFRED Valid-Seen and Valid-Unseen splits.

4 LEARNING AND INTERACTIVE EVALUATION WITH RESQ

4.1 LEARNING QUESTION-ASKING AND RECOVERY AGENTS

Following prior work on trajectory-based imitation learning for LLM agents (Chen et al., 2023; Zeng et al., 2024; Chen et al., 2024), we use supervised fine-tuning (SFT) on interaction trajectories to learn task execution, selective question asking, and answer-conditioned recovery. We consider three training settings that progressively incorporate the benchmark components. **S** uses successful autonomous trajectories to supervise task execution. **SQ** augments **S** with failure-grounded question demonstrations from RESQ-Q to learn when and what to ask. **SQR** further incorporates answer-conditioned recovery trajectories from RESQ-R to learn how to use the received response to resume task execution. All three settings use the same agent architecture and action interface and differ only in their supervision data.

For SFT, we organize the training data into successful-trajectory and question-based examples. $\mathcal{D}_S$ contains successful autonomous trajectories, for which all agent decisions are supervised. $\mathcal{D}_Q$ contains question examples constructed from RESQ-Q: each example consists of a failed-trajectory prefix through the observation at the annotated question point, followed by the annotated question as the next asking action. The failed prefix is provided as context but excluded from the training loss, so only the asking action is supervised. For the subset of questions in $\mathcal{D}_Q$ with a replay-verified recovery trajectory, we construct corresponding joint question-recovery examples $\mathcal{D}_{QR}$. Each $\mathcal{D}_{QR}$ example extends its $\mathcal{D}_Q$ counterpart with the corresponding human response from RESQ-A, followed by the recovery trajectory from RESQ-R. The human response is provided as context, while the annotated question and all post-answer reasoning and environment actions are supervised.

For each trajectory τ, $m_t^\tau \in \{0, 1\}$ denotes the loss mask at step t. We optimize all training examples with the same loss-masked autoregressive objective,

$$\mathcal{L}(\theta; \mathcal{D}) = -\sum_{\tau \in \mathcal{D}} \sum_t m_t^\tau \log \pi_\theta(a_t^\tau \mid l^\tau, h_t^\tau), \tag{1}$$

where $m_t^\tau = 1$ for all decisions in $\mathcal{D}_S$, only for the annotated ask in $\mathcal{D}_Q$, and for the annotated ask and all post-answer decisions in $\mathcal{D}_{QR}$. We train **S** on $\mathcal{D}_S$, **SQ** on $\mathcal{D}_S \cup \mathcal{D}_Q$, and **SQR** on $\mathcal{D}_S \cup \mathcal{D}_Q^{\text{only}} \cup \mathcal{D}_{QR}$, where $\mathcal{D}_Q^{\text{only}} \subset \mathcal{D}_Q$ contains only questions without recovery trajectories; thus, each $\mathcal{D}_{QR}$ example replaces rather than supplements $\mathcal{D}_Q$ example.

4.2 Interactive Evaluation Protocol

At inference time, a candidate agent follows the interaction interface defined in Section 3.1. Environment actions are executed in the simulator and return textual observations, while asking actions are answered by the fixed response oracle trained on ReSQ-A. The oracle receives the agent's free-form question, task instruction, and interaction history, together with privileged simulator context, and generates a natural-language response. The response is returned as the next textual observation and incorporated into the agent's interaction history to inform subsequent decisions.

Each episode allows at most B questions. Once this question budget is exhausted, further asking actions are disabled, and the agent continues with reasoning or environment actions. We additionally limit each episode to 50 agent decisions, counting each reasoning, acting, or asking decision once; environment observations and oracle responses do not count toward this limit. An episode terminates when the agent outputs `done` or reaches the decision limit.

Task success. Our primary metric is the task success rate SR_B, the percentage of episodes that satisfy all ALFRED goal conditions under question budget B, as checked automatically in the simulator. We write SR when the budget is clear from context.

Selective-questioning metrics. Task success alone does not reveal whether an agent asks selectively. We therefore evaluate each agent both with question budget B and with asking disabled (B=0). Under greedy decoding, the two runs coincide until the agent first asks under budget B, so the B=0 outcome serves as a per-episode reference for autonomous execution. Let $\mathcal{S}_0$ and $\mathcal{U}_0$ denote the episodes solved and unsolved at B=0, respectively, $\mathcal{H}$ the episodes in which the agent asks at least one question under budget B, and $\mathcal{S}_B$ those it solves. We report

$$\mathrm{Ask}_s = P(\mathcal{H} \mid \mathcal{S}_0),\ \mathrm{Ask}_u = P(\mathcal{H} \mid \mathcal{U}_0),\ \mathrm{Conv} = P(\mathcal{S}_B \mid \mathcal{H}, \mathcal{U}_0),\ \mathrm{Flip} = P(\mathcal{S}_B^c \mid \mathcal{H}, \mathcal{S}_0). \quad (2)$$

Ask_u measures how often the agent asks on episodes it would otherwise fail, while Conv measures how often these asked episodes succeed. Conversely, Ask_s measures how often the agent asks on episodes it would otherwise solve, while Flip measures how often these asked episodes fail.

Because episodes in which the agent does not ask are identical under budgets B and 0, the change in task success due to questioning decomposes into

$$\mathrm{SR}_B - \mathrm{SR}_0 = \underbrace{(1 - \mathrm{SR}_0)\,\mathrm{Ask}_u\,\mathrm{Conv}}_{\text{Gain}} - \underbrace{\mathrm{SR}_0\,\mathrm{Ask}_s\,\mathrm{Flip}}_{\text{Harm}}. \quad (3)$$

5 Experiments

5.1 Experimental Setup

Environment and evaluation. We evaluate agents on the full ALFRED validation sets, comprising 820 episodes in Valid-Seen and 821 episodes in Valid-Unseen, following the interactive evaluation protocol in Section 4.2.

Models. We evaluate ten model configurations across four families: Qwen3 (Yang et al., 2025) at 4B, 8B, 14B, and 32B; Llama 3.1 (Grattafiori et al., 2024) at 8B; Ministral 3 (Liu et al., 2026) at 8B and 14B; and Gemma 3 (Team et al., 2025) at 4B, 12B, and 27B. For each model and training setting, we train with three independent random seeds and report the mean and standard deviation across seeds.

Training. We fine-tune all candidate agents and the Qwen3-4B response oracle with LoRA (rank 16, learning rate 2×10^{-4}, effective batch size 16). Candidate agents are trained for one epoch under the **S**, **SQ**, and **SQR** settings in Section 4.1; within each model, the variants share the same system prompt and training configuration and differ only in supervision data. The oracle is trained for three epochs on the ReSQ-A training split with a separate user-role system prompt. Full training and hardware details are provided in Appendix B.

Inference. Both candidate agents and the response oracle use greedy decoding. Unless otherwise specified, we set the question budget to $B = 3$ and use the same fixed response oracle across candidate agents. For selective-questioning analyses, we additionally run each agent with asking disabled ($B = 0$).

Table 1: ALFRED task success rate (%) for agents trained with **S**, **SQ**, and **SQR** supervision. Mean±std over three seeds; Δ is the percentage-point change from **S**.

	Valid-Seen					Valid-Unseen				
Model	S	SQ	Δ	SQR	Δ	S	SQ	Δ	SQR	Δ
Qwen3-4B	41.7±1.0	48.5±0.5	+6.8	**52.4**±0.9	+10.7	43.3±1.1	50.6±0.7	+7.3	**54.2**±1.3	+10.9
Qwen3-8B	43.4±1.4	47.5±1.6	+4.1	**51.3**±0.9	+7.9	45.4±1.7	48.5±0.5	+3.1	**54.2**±1.3	+8.8
Qwen3-14B	42.8±0.8	50.9±0.5	+8.1	**53.0**±0.3	+10.2	45.4±1.2	51.0±0.4	+5.6	**54.3**±1.6	+8.9
Qwen3-32B	43.6±1.5	50.1±1.8	+6.5	**53.8**±2.1	+10.2	45.6±0.2	50.5±1.2	+4.9	**55.4**±1.7	+9.8
Llama3.1-8B	43.0±0.7	48.2±1.0	+5.2	**52.7**±1.4	+9.7	44.1±1.2	50.4±1.4	+6.3	**53.9**±2.3	+9.8
Ministral3-8B	41.6±2.6	47.7±2.1	+6.1	**49.1**±0.7	+7.5	42.8±2.8	48.0±4.4	+5.2	**51.4**±3.4	+8.6
Ministral3-14B	41.4±0.6	40.5±11.0	−0.9	**50.2**±2.4	+8.8	42.7±1.3	39.6±12.2	−3.1	**53.0**±5.3	+10.3
Gemma3-4B	39.6±0.9	34.9±6.1	−4.7	**46.7**±1.4	+7.1	39.5±1.7	36.9±7.4	−2.6	**45.7**±4.3	+6.2
Gemma3-12B	41.8±1.2	44.7±2.2	+2.9	**48.4**±1.8	+6.6	44.6±1.9	41.0±3.3	−3.6	**50.5**±2.7	+5.9
Gemma3-27B	40.9±1.0	49.5±2.5	+8.6	**51.9**±1.3	+11.0	42.0±0.6	47.9±2.2	+5.9	**52.9**±1.4	+10.9
Average	42.0	46.3	+4.3	**51.0**	+9.0	43.5	46.4	+2.9	**52.6**	+9.0

5.2 Effect of Question and Recovery Supervision

Main results. Table 1 compares the **S**, **SQ**, and **SQR** settings defined in Section 4.1. Question supervision (**SQ**) improves task success over **S** for 8 of 10 models on Valid-Seen and 7 of 10 on Valid-Unseen, by up to 8.6 and 7.3 percentage points, and raises the average over all models by 4.3 and 2.9 points. The effect of model scale is not consistent across families: gains show no clear trend across Qwen3 scales, increase with scale for Gemma3, and decrease for Ministral3. Several Ministral3 and Gemma3 configurations also vary substantially across seeds under **SQ**, with standard deviations of up to 12.2 points, compared with at most 1.8 for Qwen3 and Llama3.1. Adding recovery supervision (**SQR**) improves over **SQ** for all 10 models on both splits, including those that degrade under **SQ**, and raises the average gain over **S** to 9.0 points on both Valid-Seen (6.6–11.0) and Valid-Unseen (5.9–10.9), although RESQ is collected only in training scenes.

Selective questioning analysis. We first examine autonomous execution by disabling asking (B=0). As shown in Table 2, **SQ** changes task success little relative to **S**: 9 of 10 models remain within 1.3 points, with only a +0.1-point improvement on average. In contrast, **SQR** improves autonomous success by 1.9 points on average, indicating that recovery supervision also benefits execution without oracle access. A full budget sweep for **SQ** (Appendix C) shows that a small number of questions captures much of the interaction benefit; notably, **SQR** uses essentially one question per asking episode (1.00–1.01) despite the main budget of B=3.

Table 2: Autonomous success (%) on Valid-Seen at B=0. Δ: change from **S**.

Model	$\mathbf{SQ}_0$	Δ	$\mathbf{SQR}_0$	Δ
Qwen3-4B	42.6	+1.0	44.5	+2.8
Qwen3-8B	42.7	-0.7	44.1	+0.7
Qwen3-14B	44.1	+1.3	46.5	+3.7
Qwen3-32B	43.2	-0.4	46.9	+3.3
Llama3.1-8B	43.6	+0.6	45.9	+2.9
Ministral3-8B	42.6	+1.0	42.6	+1.0
Ministral3-14B	41.7	+0.3	43.3	+1.9
Gemma3-4B	37.5	-2.1	39.1	-0.5
Gemma3-12B	41.4	-0.4	43.0	+1.1
Gemma3-27B	41.6	+0.7	42.8	+1.9
Average	42.1	+0.1	43.9	+1.9

We next examine the effect of question-answer interaction by comparing each agent under B=0 and B=3. Figure 3 reveals two distinct asking regimes on Valid-Seen. Under **SQ**, Qwen3 and Llama3.1 ask on only 2–7% of episodes they would otherwise solve (Ask_s), but also on only 50–55% of those they would otherwise fail (Ask_u), yielding 5.3–7.5 points of Gain with less than 1 point of Harm. In contrast, Ministral3 and Gemma3 ask on 83–92% of otherwise-failed episodes and 53–68% of otherwise-solved ones. Because Flip rates are comparable across the two regimes (15–43% overall), their larger Harm (3.8–11.3 points) stems primarily from asking on many more otherwise-solved episodes. Harm exceeds Gain for Ministral3-14B and Gemma3-4B, yielding net interaction effects of −1.2 and −2.6 points, respectively.

Recovery supervision primarily changes what happens after a question rather than where questions are asked. **SQR** increases Conv for every model by 1.8–12.5 points, narrowing its range from 11.9–25.5% under **SQ** to 24.4–29.7%, while largely eliminating repeated questioning (1.00–1.01 vs. 2.3–

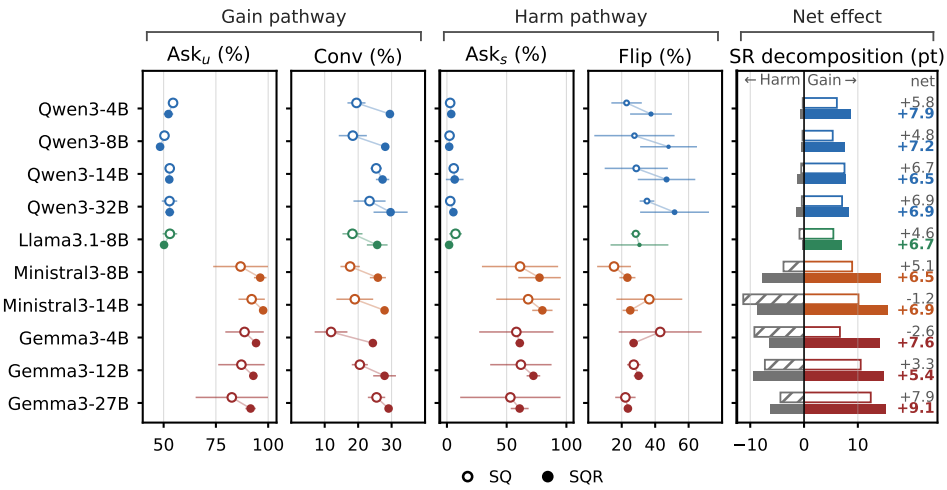

Figure 3: Selective-questioning analysis on ALFRED Valid-Seen. The panels show the Gain pathway (Ask_u, Conv), Harm pathway (Ask_s, Flip), and resulting SR decomposition. Hollow and filled markers denote **SQ** and **SQR**, respectively; error bars show standard deviation over three seeds.

Table 3: Effect of question timing on ALFRED task success. Main values are success rates (%); smaller values show average questions per episode. Rows are grouped by trigger type.

	Valid-Seen				Valid-Unseen			
Trigger	4B	8B	14B	32B	4B	8B	14B	32B
Never	42.7 0.00	42.9 0.00	44.1 0.00	43.7 0.00	44.2 0.00	46.4 0.00	46.0 0.00	47.0 0.00
Random	43.2 0.66	43.8 0.59	46.0 0.59	45.2 0.55	44.7 0.70	48.6 0.75	47.5 0.75	48.4 0.72
KnowNo	43.2 1.38	45.7 1.60	49.3 1.35	47.8 1.33	49.5 1.37	**51.2** 1.44	**53.3** 1.42	**56.2** 1.04
Act-MeanNLL	46.7 0.74	47.0 0.61	50.4 0.81	49.6 0.66	49.0 0.97	49.9 1.01	50.7 0.84	50.1 0.83
Act-MinProb	46.6 0.77	46.2 0.63	50.6 0.82	49.3 0.66	49.5 0.92	49.7 1.01	51.4 0.84	50.5 0.83
Learned gate	**49.0** 0.75	**49.3** 0.71	**51.0** 0.74	**51.0** 0.72	**50.5** 0.80	49.0 0.69	50.5 0.70	51.6 0.67

2.9 questions per asking episode). Qwen3 and Llama3.1 remain selective, with net interaction gains of 6.5–7.9 points, whereas Ministral3 and Gemma3 continue to ask broadly but higher conversion makes the net effect positive for all five models (5.4–9.1 points). Thus, similar benefits arise through two regimes: selective askers incur little Harm but leave many failures unqueried, while broad askers cover most failures at the cost of questioning many otherwise-solved episodes. This suggests complementary headroom in improving coverage for the former and selectivity for the latter.

5.3 When Should Agents Ask?

ReSQ-Q directly supervises when to ask, while prior work uses uncertainty-based criteria to trigger external assistance. We therefore compare question-triggering criteria on the same trained agent. Keeping the trained **SQ** agent, response oracle, and question budget ($B = 3$) fixed, we compare its learned gate with random timing, a KnowNo-style trigger (Ren et al., 2023), and two token-level triggers based on mean action-token NLL (Act-MeanNLL) and minimum action-token probability (Act-MinProb) (Han et al., 2024; Jiang et al., 2023). External triggers disable native asking and invoke the same agent's question generator. Random and token-level triggers are calibrated on the opposite validation split to match the learned gate's interaction rate, whereas KnowNo uses its conformal-coverage criterion (Appendix D); all triggers use the same single seed per model.

Table 3 shows that the choice of question-triggering criterion substantially affects task success. On Valid-Seen, random timing improves over never asking by only 0.5–1.9 points, whereas Act-MeanNLL and Act-MinProb improve success by 3.3–6.5 points at broadly similar interaction rates. The learned gate performs best at all four Qwen3 scales, with gains of 6.3–7.3 points.

On Valid-Unseen, both token-level triggers consistently outperform random timing, while the KnowNo-style trigger performs best at three of four scales, albeit with a higher interaction rate than the learned gate (1.04–1.44 vs. 0.67–0.80 questions per episode). Since the question generator is fixed, the gains of the rate-matched token-level triggers indicate that selecting more useful intervention points matters beyond interaction frequency alone. Overall, the learned gate is strongest on Valid-Seen, whereas uncertainty-based triggers are more competitive on Valid-Unseen.

5.4 Who Answers Matters

Table 4: Offline and online comparison of response sources.

	Offline			SQ		SQR	
Respondent	Pol.	AnsFact	Extra	SR	Conv	SR	Conv
Human	89.0	84.3	57.2	-	-	-	-
ReSQ oracle	**85.6**	87.0	31.0	49.3	21.7	52.6	28.7
GPT-5.5 zs	78.8	95.8	51.2	50.5	**26.2**	53.4	**31.7**
GPT-5.5 fs	80.0	97.7	58.1	49.8	23.7	52.6	29.3

Offline comparison. We first compare response sources on 432 held-out ReSQ-Q questions, each paired with three human responses from ReSQ-A (details in Appendix E). Pol. measures pairwise agreement with human response polarity. AnsFact and Extra are evaluated by Claude Opus 5 (Anthropic, 2026), chosen so that GPT-5.5 responses are not judged by the same model family: AnsFact measures whether the direct answer is supported by the privileged context and task goal, and Extra measures whether a response provides information beyond what is needed to resolve the question; Extra is descriptive, not a quality metric. The ReSQ oracle achieves the highest Pol. among model respondents, only 3.4 points below human inter-annotator agreement, and its AnsFact is close to the human reference (87.0 vs. 84.3). The human AnsFact may be understated: during collection, annotators could additionally inspect the simulator when the recorded context was insufficient, so answers grounded in such inspection can be scored as unsupported by a judge that sees only the recorded context. GPT-5.5 receives substantially higher AnsFact but lower Pol.; deictic confirmation questions such as "Is *this* the bowl you meant?" form the largest category of these disagreements. Respondents also differ markedly in scope: the oracle is far more conservative than humans (31.0 vs. 57.2 Extra), whereas four-shot GPT-5.5 is similar (58.1).

Online comparison. We next substitute the respondent while keeping the agent and interaction protocol fixed; Table 4 reports SR and Conv averaged across Qwen3 4B–32B, with per-scale results in Appendix F. For SQ agents, zero-shot GPT-5.5 raises SR from 49.3 to 50.5 and Conv from 21.7 to 26.2 over the ReSQ oracle. Stronger offline metrics, however, do not imply greater interactive utility: four-shot GPT-5.5 achieves higher AnsFact and Extra, yet yields lower SR and Conv in all eight model-scale–training combinations. Respondent sensitivity decreases under SQR: across sources, the spread drops from 1.2 to 0.8 points in SR and from 4.5 to 3.0 in Conv, while SQR improves SR over SQ in all 12 respondent–scale combinations. Thus, absolute scores require a fixed respondent, whereas the benefit of recovery supervision is robust across the response sources tested.

6 Conclusions

We introduced ReSQ, a benchmark for studying the complete interaction loop of questioning, answering, and recovery in embodied agents. Built from failures that arise during autonomous task execution, ReSQ links failure-grounded question demonstrations (ReSQ-Q), independently collected human responses (ReSQ-A), and answer-conditioned recovery trajectories (ReSQ-R). Across diverse LLM families and scales, question supervision generally improves task completion, while adding recovery supervision consistently provides further gains, yielding an average 9.0-point improvement over autonomous-task supervision on both ALFRED validation splits. Our analyses further show that successful interaction depends on more than simply allowing agents to ask: questions must be issued selectively, received information must be effectively incorporated into subsequent execution, and the choice of respondent can itself affect measured performance. Together, these results motivate evaluating information-seeking agents as a complete questioning–answering–recovery process rather than treating question asking as an isolated capability. We hope ReSQ provides a foundation for studying agents that recognize when external information is needed and use it effectively to recover from failures.

AI USE STATEMENT

We used generative AI tools to assist with literature retrieval, language editing, and manuscript polishing. During dataset construction, GPT-4o was used to translate Korean RESQ-Q annotations into English and to produce initial uncertainty-type labels. All initial labels were subsequently reviewed by the authors, and the prompts used for translation and labeling are provided in Appendix A. Generative AI models were also used to generate autonomous task-execution trajectories and to support intermediate data processing during benchmark construction. All AI-assisted outputs incorporated into the manuscript or benchmark were reviewed by the authors.

ETHICS STATEMENT

The human-participant study used to construct RESQ-Q was conducted under institutional IRB approval. RESQ-A and RESQ-R were collected separately by paid annotators engaged through an external organization and were not part of the IRB-approved participant study. All annotations concern household tasks instantiated in the simulated ALFRED/AI2-THOR environment; the study does not involve physical robot deployment or real-world household data.

REPRODUCIBILITY STATEMENT

We document the procedures and settings used for data construction, training, and evaluation throughout the main paper and appendices. Section 3 describes the RESQ construction procedure and data splits, Section 4 defines the training objectives and interactive evaluation protocol, and Section 5 reports the experimental setup and evaluation results. Appendix A provides the translation and uncertainty-labeling prompts and annotation details; Appendix B reports training hyperparameters, random seeds, input processing, and hardware configurations; Appendix C presents the question-budget analysis; Appendix D describes calibration of the external questioning triggers; and Appendices E and F provide details of the offline and online respondent evaluations, respectively. Main agent results are reported over three independently trained random seeds. We additionally provide selected RESQ data and implementation code in the supplementary materials to facilitate inspection and reproduction of the experimental pipeline. We will publicly release the full RESQ benchmark and code upon publication.

REFERENCES

Anthropic. System card: Claude opus 5, July 2026. URL https://www.anthropic.com/claude-opus-5-system-card.

Baian Chen, Chang Shu, Ehsan Shareghi, Nigel Collier, Karthik Narasimhan, and Shunyu Yao. Fireact: Toward language agent fine-tuning, 2023. URL https://arxiv.org/abs/2310.05915.

Zehui Chen, Kuikun Liu, Qiuchen Wang, Wenwei Zhang, Jiangning Liu, Dahua Lin, Kai Chen, and Feng Zhao. Agent-FLAN: Designing data and methods of effective agent tuning for large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 9354–9366, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.557. URL https://aclanthology.org/2024.findings-acl.557/.

Jae-Woo Choi, Hyungmin Kim, Hyobin Ong, Youngwoo Yoon, Minsu Jang, Dohyung Kim, and Jaehong Kim. Reactree: Hierarchical llm agent trees with control flow for long-horizon task planning. In *Proceedings of the 25th International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '26, pp. 319–328, Richland, SC, 2026. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9798400723179. doi: 10.65109/UCGT7089. URL https://doi.org/10.65109/UCGT7089.

Yinpei Dai, Run Peng, Sikai Li, and Joyce Chai. Think, act, and ask: Open-world interactive personalized robot navigation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3296–3303, 2024. doi: 10.1109/ICRA57147.2024.10610178.

Xiaofeng Gao, Qiaozi Gao, Ran Gong, Kaixiang Lin, Govind Thattai, and Gaurav S. Sukhatme. Dialfred: Dialogue-enabled agents for embodied instruction following. *IEEE Robotics and Automation Letters*, 7(4):10049–10056, 2022. doi: 10.1109/LRA.2022.3193254.

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, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni,

Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Jiuzhou Han, Wray Buntine, and Ehsan Shareghi. Towards uncertainty-aware language agent. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 6662–6685, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.398. URL https://aclanthology.org/2024.findings-acl.398/.

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

Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Tomas Jackson, Noah Brown, Linda Luu, Sergey Levine, Karol Hausman, and brian ichter. Inner monologue: Embodied reasoning through planning with language models. In Karen Liu, Dana Kulic, and Jeff Ichnowski (eds.), *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 1769–1782. PMLR, 14–18 Dec 2023. URL https://proceedings.mlr.press/v205/huang23c.html.

Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Madry, Alex Baker-Whitcomb,

Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoochian, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, Dane Sherburn, Daniel Kappler, Daniel Levin, Daniel Levy, David Carr, David Farhi, David Mely, David Robinson, David Sasaki, Denny Jin, Dev Valladares, Dimitris Tsipras, Doug Li, Duc Phong Nguyen, Duncan Findlay, Edede Oiwoh, Edmund Wong, Ehsan Asdar, Elizabeth Proehl, Elizabeth Yang, Eric Antonow, Eric Kramer, Eric Peterson, Eric Sigler, Eric Wallace, Eugene Brevdo, Evan Mays, Farzad Khorasani, Felipe Petroski Such, Filippo Raso, Francis Zhang, Fred von Lohmann, Freddie Sulit, Gabriel Goh, Gene Oden, Geoff Salmon, Giulio Starace, Greg Brockman, Hadi Salman, Haiming Bao, Haitang Hu, Hannah Wong, Haoyu Wang, Heather Schmidt, Heather Whitney, Heewoo Jun, Hendrik Kirchner, Henrique Ponde de Oliveira Pinto, Hongyu Ren, Huiwen Chang, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian O'Connell, Ian Osband, Ian Silber, Ian Sohl, Ibrahim Okuyucu, Ikai Lan, Ilya Kostrikov, Ilya Sutskever, Ingmar Kanitscheider, Ishaan Gulrajani, Jacob Coxon, Jacob Menick, Jakub Pachocki, James Aung, James Betker, James Crooks, James Lennon, Jamie Kiros, Jan Leike, Jane Park, Jason Kwon, Jason Phang, Jason Teplitz, Jason Wei, Jason Wolfe, Jay Chen, Jeff Harris, Jenia Varavva, Jessica Gan Lee, Jessica Shieh, Ji Lin, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joanne Jang, Joaquin Quinonero Candela, Joe Beutler, Joe Landers, Joel Parish, Johannes Heidecke, John Schulman, Jonathan Lachman, Jonathan McKay, Jonathan Uesato, Jonathan Ward, Jong Wook Kim, Joost Huizinga, Jordan Sitkin, Jos Kraaijeveld, Josh Gross, Josh Kaplan, Josh Snyder, Joshua Achiam, Joy Jiao, Joyce Lee, Juntang Zhuang, Justyn Harriman, Kai Fricke, Kai Hayashi, Karan Singhal, Katy Shi, Kavin Karthik, Kayla Wood, Kendra Rimbach, Kenny Hsu, Kenny Nguyen, Keren Gu-Lemberg, Kevin Button, Kevin Liu, Kiel Howe, Krithika Muthukumar, Kyle Luther, Lama Ahmad, Larry Kai, Lauren Itow, Lauren Workman, Leher Pathak, Leo Chen, Li Jing, Lia Guy, Liam Fedus, Liang Zhou, Lien Mamitsuka, Lilian Weng, Lindsay McCallum, Lindsey Held, Long Ouyang, Louis Feuvrier, Lu Zhang, Lukas Kondraciuk, Lukasz Kaiser, Luke Hewitt, Luke Metz, Lyric Doshi, Mada Aflak, Maddie Simens, Madelaine Boyd, Madeleine Thompson, Marat Dukhan, Mark Chen, Mark Gray, Mark Hudnall, Marvin Zhang, Marwan Aljubeh, Mateusz Litwin, Matthew Zeng, Max Johnson, Maya Shetty, Mayank Gupta, Meghan Shah, Mehmet Yatbaz, Meng Jia Yang, Mengchao Zhong, Mia Glaese, Mianna Chen, Michael Janner, Michael Lampe, Michael Petrov, Michael Wu, Michele Wang, Michelle Fradin, Michelle Pokrass, Miguel Castro, Miguel Oom Temudo de Castro, Mikhail Pavlov, Miles Brundage, Miles Wang, Minal Khan, Mira Murati, Mo Bavarian, Molly Lin, Murat Yesildal, Nacho Soto, Natalia Gimelshein, Natalie Cone, Natalie Staudacher, Natalie Summers, Natan LaFontaine, Neil Chowdhury, Nick Ryder, Nick Stathas, Nick Turley, Nik Tezak, Niko Felix, Nithanth Kudige, Nitish Keskar, Noah Deutsch, Noel Bundick, Nora Puckett, Ofir Nachum, Ola Okelola, Oleg Boiko, Oleg Murk, Oliver Jaffe, Olivia Watkins, Olivier Godement, Owen Campbell-Moore, Patrick Chao, Paul McMillan, Pavel Belov, Peng Su, Peter Bak, Peter Bakkum, Peter Deng, Peter Dolan, Peter Hoeschele, Peter Welinder, Phil Tillet, Philip Pronin, Philippe Tillet, Prafulla Dhariwal, Qiming Yuan, Rachel Dias, Rachel Lim, Rahul Arora, Rajan Troll, Randall Lin, Rapha Gontijo Lopes, Raul Puri, Reah Miyara, Reimar Leike, Renaud Gaubert, Reza Zamani, Ricky Wang, Rob Donnelly, Rob Honsby, Rocky Smith, Rohan Sahai, Rohit Ramchandani, Romain Huet, Rory Carmichael, Rowan Zellers, Roy Chen, Ruby Chen, Ruslan Nigmatullin, Ryan Cheu, Saachi Jain, Sam Altman, Sam Schoenholz, Sam Toizer, Samuel Miserendino, Sandhini Agarwal, Sara Culver, Scott Ethersmith, Scott Gray, Sean Grove, Sean Metzger, Shamez Hermani, Shantanu Jain, Shengjia Zhao, Sherwin Wu, Shino Jomoto, Shirong Wu, Shuaiqi, Xia, Sonia Phene, Spencer Papay, Srinivas Narayanan, Steve Coffey, Steve Lee, Stewart Hall, Suchir Balaji, Tal Broda, Tal Stramer, Tao Xu, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal

Patwardhan, Thomas Cunninghman, Thomas Degry, Thomas Dimson, Thomas Raoux, Thomas Shadwell, Tianhao Zheng, Todd Underwood, Todor Markov, Toki Sherbakov, Tom Rubin, Tom Stasi, Tomer Kaftan, Tristan Heywood, Troy Peterson, Tyce Walters, Tyna Eloundou, Valerie Qi, Veit Moeller, Vinnie Monaco, Vishal Kuo, Vlad Fomenko, Wayne Chang, Weiyi Zheng, Wenda Zhou, Wesam Manassra, Will Sheu, Wojciech Zaremba, Yash Patil, Yilei Qian, Yongjik Kim, Youlong Cheng, Yu Zhang, Yuchen He, Yuchen Zhang, Yujia Jin, Yunxing Dai, and Yury Malkov. Gpt-4o system card, 2024. URL https://arxiv.org/abs/2410.21276.

Brian Ichter, Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, Dmitry Kalashnikov, Sergey Levine, Yao Lu, Carolina Parada, Kanishka Rao, Pierre Sermanet, Alexander T Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Mengyuan Yan, Noah Brown, Michael Ahn, Omar Cortes, Nicolas Sievers, Clayton Tan, Sichun Xu, Diego Reyes, Jarek Rettinghouse, Jornell Quiambao, Peter Pastor, Linda Luu, Kuang-Huei Lee, Yuheng Kuang, Sally Jesmonth, Nikhil J. Joshi, Kyle Jeffrey, Rosario Jauregui Ruano, Jasmine Hsu, Keerthana Gopalakrishnan, Byron David, Andy Zeng, and Chuyuan Kelly Fu. Do as i can, not as i say: Grounding language in robotic affordances. In Karen Liu, Dana Kulic, and Jeff Ichnowski (eds.), *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 287–318. PMLR, 14–18 Dec 2023. URL https://proceedings.mlr.press/v205/ichter23a.html.

Anastasia Ivanova, Bakaeva Eva, Zoya Volovikova, Alexey Kovalev, and Aleksandr Panov. AmbiK: Dataset of ambiguous tasks in kitchen environment. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 33216–33241, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1593. URL https://aclanthology.org/2025.acl-long.1593/.

Zhengbao Jiang, Frank Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7969–7992, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.495. URL https://aclanthology.org/2023.emnlp-main.495/.

Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, Aniruddha Kembhavi, Abhinav Gupta, and Ali Farhadi. Ai2-thor: An interactive 3d environment for visual ai, 2017. URL https://arxiv.org/abs/1712.05474.

Kaiqu Liang, Zixu Zhang, and Jaime Fernández Fisac. Introspective planning: Aligning robots' uncertainty with inherent task ambiguity. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 71998–72031. Curran Associates, Inc., 2024. doi: 10.52202/079017-2299. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/8451a20c5a7e0ee5671dda28f7daf7f3-Paper-Conference.pdf.

Alexander H. Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, Alexandre Sablayrolles, Amélie Héliou, Amos You, Andy Ehrenberg, Andy Lo, Anton Eliseev, Antonia Calvi, Avinash Sooriyarachchi, Baptiste Bout, Baptiste Rozière, Baudouin De Monicault, Clémence Lanfranchi, Corentin Barreau, Cyprien Courtot, Daniele Grattarola, Darius Dabert, Diego de las Casas, Elliot Chane-Sane, Faruk Ahmed, Gabrielle Berrada, Gaëtan Ecrepont, Gauthier Guinet, Georgii Novikov, Guillaume Kunsch, Guillaume Lample, Guillaume Martin, Gunshi Gupta, Jan Ludziejewski, Jason Rute, Joachim Studnia, Jonas Amar, Joséphine Delas, Josselin Somerville Roberts, Karmesh Yadav, Khyathi Chandu, Kush Jain, Laurence Aitchison, Laurent Fainsin, Léonard Blier, Lingxiao Zhao, Louis Martin, Lucile Saulnier, Luyu Gao, Maarten Buyl, Margaret Jennings, Marie Pellat, Mark Prins, Mathieu Poirée, Mathilde Guillaumin, Matthieu Dinot, Matthieu Futeral, Maxime Darrin, Maximilian Augustin, Mia Chiquier, Michel Schimpf, Nathan Grinsztajn, Neha Gupta, Nikhil Raghuraman, Olivier Bousquet, Olivier Duchenne, Patricia Wang, Patrick von Platen, Paul Jacob, Paul Wambergue, Paula Kurylowicz,

Pavankumar Reddy Muddireddy, Philomène Chagniot, Pierre Stock, Pravesh Agrawal, Quentin Torroba, Romain Sauvestre, Roman Soletskyi, Rupert Menneer, Sagar Vaze, Samuel Barry, Sanchit Gandhi, Siddhant Waghjale, Siddharth Gandhi, Soham Ghosh, Srijan Mishra, Sumukh Aithal, Szymon Antoniak, Teven Le Scao, Théo Cachet, Theo Simon Sorg, Thibaut Lavril, Thiziri Nait Saada, Thomas Chabal, Thomas Foubert, Thomas Robert, Thomas Wang, Tim Lawson, Tom Bewley, Tom Bewley, Tom Edwards, Umar Jamil, Umberto Tomasini, Valeriia Nemychnikova, Van Phung, Vincent Maladière, Virgile Richard, Wassim Bouaziz, Wen-Ding Li, William Marshall, Xinghui Li, Xinyu Yang, Yassine El Ouahidi, Yihan Wang, Yunhao Tang, and Zaccharie Ramzi. Ministral 3, 2026. URL https://arxiv.org/abs/2601.08584.

Tarek Naous, Philippe Laban, Wei Xu, and Jennifer Neville. Flipping the dialogue: Training and evaluating user language models. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 82079–82103, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/84d10f909b96ac12f264c9d801000a6a-Paper-Conference.pdf.

Aishwarya Padmakumar, Jesse Thomason, Ayush Shrivastava, Patrick Lange, Anjali Narayan-Chen, Spandana Gella, Robinson Piramuthu, Gokhan Tur, and Dilek Hakkani-Tur. Teach: Task-driven embodied agents that chat. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 2017–2025, 2022.

Jeongeun Park, Seungwon Lim, Joonhyung Lee, Sangbeom Park, Minsuk Chang, Youngjae Yu, and Sungjoon Choi. Clara: Classifying and disambiguating user commands for reliable interactive robotic agents. *IEEE Robotics and Automation Letters*, 9(2):1059–1066, 2024. doi: 10.1109/LRA.2023.3338514.

Ram Ramrakhya, Matthew Chang, Xavier Puig, Ruta Desai, Zsolt Kira, and Roozbeh Mottaghi. Grounding multimodal llms to embodied agents that ask for help with reinforcement learning, 2025. URL https://arxiv.org/abs/2504.00907.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410/.

Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, Zhenjia Xu, Dorsa Sadigh, Andy Zeng, and Anirudha Majumdar. Robots that ask for help: Uncertainty alignment for large language model planners. In Jie Tan, Marc Toussaint, and Kourosh Darvish (eds.), *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 661–682. PMLR, 06–09 Nov 2023. URL https://proceedings.mlr.press/v229/ren23a.html.

Ying Shen, Daniel Bis, Cynthia Lu, and Ismini Lourentzou. Elba: Learning by asking for embodied visual navigation and task completion. In *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, pp. 5177–5186, February 2025.

Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10737–10746, 2020. doi: 10.1109/CVPR42600.2020.01075.

Chan Hee Song, Jiaman Wu, Clayton Washington, Brian M Sadler, Wei-Lun Chao, and Yu Su. Llm-planner: Few-shot grounded planning for embodied agents with large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2998–3009, October 2023.

Francesco Taioli, Edoardo Zorzi, Gianni Franchi, Alberto Castellini, Alessandro Farinelli, Marco Cristani, and Yiming Wang. Collaborative instance object navigation: Leveraging uncertainty-awareness to minimize human-agent dialogues. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18781–18792, 2025. doi: 10.1109/ICCV51701.2025.01745.

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, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xiaohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Coleman, Yi Gao, Basil Mustafa, Iain Barr, Emilio Parisotto, David Tian, Matan Eyal, Colin Cherry, Jan-Thorsten Peter, Danila Sinopalnikov, Surya Bhupatiraju, Rishabh Agarwal, Mehran Kazemi, Dan Malkin, Ravin Kumar, David Vilar, Idan Brusilovsky, Jiaming Luo, Andreas Steiner, Abe Friesen, Abhanshu Sharma, Abheesht Sharma, Adi Mayrav Gilady, Adrian Goedeckemeyer, Alaa Saade, Alex Feng, Alexander Kolesnikov, Alexei Bendebury, Alvin Abdagic, Amit Vadi, András György, André Susano Pinto, Anil Das, Ankur Bapna, Antoine Miech, Antoine Yang, Antonia Paterson, Ashish Shenoy, Ayan Chakrabarti, Bilal Piot, Bo Wu, Bobak Shahriari, Bryce Petrini, Charlie Chen, Charline Le Lan, Christopher A. Choquette-Choo, CJ Carey, Cormac Brick, Daniel Deutsch, Danielle Eisenbud, Dee Cattle, Derek Cheng, Dimitris Paparas, Divyashree Shivakumar Sreepathihalli, Doug Reid, Dustin Tran, Dustin Zelle, Eric Noland, Erwin Huizenga, Eugene Kharitonov, Frederick Liu, Gagik Amirkhanyan, Glenn Cameron, Hadi Hashemi, Hanna Klimczak-Plucińska, Harman Singh, Harsh Mehta, Harshal Tushar Lehri, Hussein Hazimeh, Ian Ballantyne, Idan Szpektor, Ivan Nardini, Jean Pouget-Abadie, Jetha Chan, Joe Stanton, John Wieting, Jonathan Lai, Jordi Orbay, Joseph Fernandez, Josh Newlan, Ju yeong Ji, Jyotinder Singh, Kat Black, Kathy Yu, Kevin Hui, Kiran Vodrahalli, Klaus Greff, Linhai Qiu, Marcella Valentine, Marina Coelho, Marvin Ritter, Matt Hoffman, Matthew Watson, Mayank Chaturvedi, Michael Moynihan, Min Ma, Nabila Babar, Natasha Noy, Nathan Byrd, Nick Roy, Nikola Momchev, Nilay Chauhan, Noveen Sachdeva, Oskar Bunyan, Pankil Botarda, Paul Caron, Paul Kishan Rubenstein, Phil Culliton, Philipp Schmid, Pier Giuseppe Sessa, Pingmei Xu, Piotr Stanczyk, Pouya Tafti, Rakesh Shivanna, Renjie Wu, Renke Pan, Reza Rokni, Rob Willoughby, Rohith Vallu, Ryan Mullins, Sammy Jerome, Sara Smoot, Sertan Girgin, Shariq Iqbal, Shashir Reddy, Shruti Sheth, Siim Põder, Sijal Bhatnagar, Sindhu Raghuram Panyam, Sivan Eiger, Susan Zhang, Tianqi Liu, Trevor Yacovone, Tyler Liechty, Uday Kalra, Utku Evci, Vedant Misra, Vincent Roseberry, Vlad Feinberg, Vlad Kolesnikov, Woohyun Han, Woosuk Kwon, Xi Chen, Yinlam Chow, Yuvein Zhu, Zichuan Wei, Zoltan Egyed, Victor Cotruta, Minh Giang, Phoebe Kirk, Anand Rao, Kat Black, Nabila Babar, Jessica Lo, Erica Moreira, Luiz Gustavo Martins, Omar Sanseviero, Lucas Gonzalez, Zach Gleicher, Tris Warkentin, Vahab Mirrokni, Evan Senter, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, Yossi Matias, D. Sculley, Slav Petrov, Noah Fiedel, Noam Shazeer, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Jean-Baptiste Alayrac, Rohan Anil, Dmitry, Lepikhin, Sebastian Borgeaud, Olivier Bachem, Armand Joulin, Alek Andreev, Cassidy Hardin, Robert Dadashi, and Léonard Hussenot. Gemma 3 technical report, 2025. URL https://arxiv.org/abs/2503.19786.

Wenxuan Wang, Shi Juluan, Zixuan Ling, Yuk-Kit Chan, Chaozheng Wang, Cheryl Lee, Youliang Yuan, Jen-tse Huang, Wenxiang Jiao, and Michael R. Lyu. Learning to ask: When LLM agents meet unclear instruction. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 21773–21784, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1104. URL https://aclanthology.org/2025.emnlp-main.1104/.

Wenli Xiao, Haotian Lin, Andy Peng, Haoru Xue, Tairan He, Zhengyi Luo, Yuqi Xie, Fengyuan Hu, Jim Fan, Guanya Shi, and Yuke Zhu. Self-improving vision-language-action models with data generation via residual rl. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 13209–13236, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/163a875ab0c2631ed30b7435c4c9b8f1-Paper-Conference.pdf.

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, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger

Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Aohan Zeng, Mingdao Liu, Rui Lu, Bowen Wang, Xiao Liu, Yuxiao Dong, and Jie Tang. Agent-Tuning: Enabling generalized agent abilities for LLMs. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 3053–3077, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.181. URL https://aclanthology.org/2024.findings-acl.181/.

Ganlong Zhao, Zijia Tang, Xingping Chen, Zhanghui Kuang, Ye Tian, and Guanbin Li. Flare: A failure-aware framework for autonomous correction and recovery in visual-language robotic manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22391–22401, June 2026.

Ming Zhu, Juntao Tan, Rithesh Murthy, Jielin Qiu, Liangwei Yang, Wenting Zhao, Silvio Savarese, Shelby Heinecke, and Huan Wang. Realusersim: Bridging the reality gap in agent benchmarking via grounded user simulation. In *Third Conference on Language Modeling*, 2026. URL https://openreview.net/forum?id=1izVEEZAxT.

A ReSQ-Q Details

A.1 Translation Prompt

Most ReSQ-Q annotations were collected in Korean and translated into English using GPT-4o with the prompt below. The prompt also handles English and mixed-language inputs and explicitly preserves object names and identifiers appearing in the trajectories (e.g., `SoapBottle (1)`).

Listing 1: Prompt for Korean-to-English translation.

```
# Role
You are a professional linguist and translator specializing in Embodied
    AI and Robotics.
Your task is to refine or translate the user's input text into natural,
    fluent English.

# Instructions based on Input Language:
1. If the input is in Korean: Translate it into natural spoken English.
2. If the input is in English: Correct any grammatical errors, fix
    typos, and polish it to sound like a native speaker.
3. If the input is Mixed (Korean + English): Translate the whole
    sentence into a coherent English sentence.

# Guidelines:
- Term Preservation: Strictly preserve specific object names and their
    identifiers (e.g., 'soapbottle (1)', 'Desk(1)', 'tissue box'). Do
    not translate, paraphrase, or remove the ID numbers associated with
    these objects.

# Output Format
Output ONLY the final English text string. Do not include quotation
    marks, explanations, or labels.
```

A.2 Uncertainty Taxonomy and Labeling Protocol

We characterize the information needs expressed in ReSQ-Q using six uncertainty types: object location, object identification, planning and decision, instruction clarification, state and memory, and failure cause. The taxonomy is defined by the underlying source of information sought by the agent rather than the surface form of the question.

For initial labeling, we use GPT-4o to assign a single uncertainty type to each question with the prompt below. The prompt defines the six categories and provides disambiguation rules for common boundary cases. Two authors then review all initial assignments in two successive passes using the same category definitions and disambiguation rules, resolving ambiguous cases through discussion. During review, questions judged to simultaneously target two distinct sources of uncertainty are assigned both labels. This results in 15 questions receiving two labels and 2,175 label assignments for the 2,160 questions.

Listing 2: Prompt for uncertainty type labeling.

```
# Role
You are an expert linguist and question classifier.
Your task is to analyze the uncertainty expressed in an embodied agent's
    question during a household task.
You must follow the criteria below.
You must respond only with a perfect JSON object and no other
    explanatory text.

# Criteria 1: Type of Uncertainty (6 Types)
Classify the type of the uncertainty:
1. object_location: Uncertainty about where an object is located. (e.g.,
    "Where is the apple?", "Is there any mug in the room?")
```

```
2. object_identification: Uncertainty about the identity or state of an
    object. (e.g., "Is this the tomato you mentioned?", "Is the apple
    chilled already?")
3. planning_decision: Uncertainty about the next action or how to use a
    tool. (e.g., "What should I do next?", "Should I cut it first?",
    "How do I turn this on?")
4. instruction_clarification: Uncertainty about the given instruction
    itself. (e.g., "Which apple did you mean?", "Did you say pick up
    fork instead of a ladle?")
5. state_memory: Uncertainty about the agent's past actions or current
    status. (e.g., "Did I already close the fridge?", "Am I holding the
    knife?")
6. failure_cause: Uncertainty about why an action or task failed. (e.g.,
    "Why can't I pick this up?", "What is the reason for the failure?")

# Output Format
Provide the result in the following JSON format.
{"uncertainty_type": "[One of the 6 types]"}

# Edge Case Guidelines (Disambiguation Rules)
If a question seems ambiguous, strictly apply the following rules:

- [object_location vs. object_identification]
  If searching for an object's whereabouts or its spatial relation
    (e.g., "Is the apple inside the microwave?"), choose
    `object_location`.
  If the agent is looking at a specific object and verifying if it is
    the correct target or checking its state (e.g., "Is this the counter
    you want?", "Is the apple sliced?"), choose `object_identification`.

- [state_memory vs. object_identification]
  If the question is about the agent's own action log or current
    localization (e.g., "Did I close it?", "Am I in front of the
    microwave?"), choose `state_memory`.
  If the question is about the visual state of the environment (e.g.,
    "Is the fridge closed?"), choose `object_identification`.

- [planning_decision vs. object_location]
  If the agent is asking to find a specific target object, choose
    `object_location`.
  If the agent doesn't know what to look for and is asking for a
    strategy or a tool to solve a problem (e.g., "Is there anything
    useful on the shelf?"), choose `planning_decision`.

- [instruction_clarification vs. planning_decision]
  If the agent understands the goal but doesn't know the causal logic or
    how to proceed (e.g., "Why do I need to pick up the watch?"), choose
    `planning_decision`.
  If the agent doesn't understand the user's constraints or words (e.g.,
    "Can I use object B instead of A?"), choose
    `instruction_clarification`.
```

Table 5 reports the resulting label distribution together with representative questions and corresponding human responses from RESQ-A. Because 15 questions receive two uncertainty labels, the category counts sum to 2,175 rather than the 2,160 unique questions.

B TRAINING DETAILS

We train the **S**, **SQ**, and **SQR** agents using LLaMA-Factory. For each model, the three settings share the same system prompt and training hyperparameters and differ only in the supervision data defined in Section 4.1. Table 6 summarizes the common configuration. Each setting is trained with three random seeds: 42, 43, and 44.

Table 5: Representative question-response pairs for the six uncertainty types in RESQ.

Uncertainty type	# Q	Example question	Example human response
Object location	699	"Where is the lamp?"	"The FloorLamp (1) is near the Box (1)."
Object identification	618	"Is this the table you were talking about?"	"Yes, that's correct. You should slice the Apple (1), heat it in the Microwave (1), and then place it on this DiningTable (1)."
Planning and decision	479	"What should I do if the lamp is already on?"	"The one that is already on is the DeskLamp (1). The one you need to find is the FloorLamp (1). The FloorLamp (1) is near the Box (2)."
Instruction clarification	208	"Do you mean a bowl or a ball?"	"The item you need to put the Pen (1) into is the Bowl (1), not a ball."
State and memory	87	"Did I place the book correctly on the desk?"	"No. You placed the CD (2) on the Desk (1). You need to place the Book (1) instead. The Book (1) is on the Bed (1)."
Failure cause	84	"The refrigerator isn't opening. Could it be locked?"	"No. Find the Fridge (1) door handle, grab it, and pull it."

Table 6: Common hyperparameters for agent fine-tuning.

Hyperparameter	Value
Training epochs	1
Effective batch size	16
Optimizer	AdamW (`adamw_torch`)
Learning rate	2×10^{-4}
Learning-rate schedule	Cosine
Warmup ratio	0.1
Weight decay	0.0
LoRA rank	16
LoRA alpha	32
LoRA dropout	0.0
LoRA target modules	All linear modules
Maximum sequence length	8,192 tokens
Precision	BF16
Quantization	None
Gradient checkpointing	Enabled
Sequence packing	Disabled
Random seeds	42, 43, 44

Batch configuration. We maintain an effective batch size of 16 across all models and training settings by adjusting the per-device batch size and gradient accumulation steps to the hardware configuration. The effective batch size is the product of the per-device batch size, the number of data-parallel processes, and the number of gradient accumulation steps. All settings use one training epoch; because their supervision datasets differ in size, their numbers of optimizer updates may differ.

Input processing. Training examples are represented as multi-turn conversations in ShareGPT format. During preprocessing, trajectories are divided into context windows. Sequence packing is disabled, and any resulting example exceeding the 8,192-token limit is truncated by the training framework. We enable `mask_history`, so that only the final assistant turn in each example contributes to the training loss; preceding turns serve as context.

Hardware. Training runs use NVIDIA RTX A6000 (48 GB), RTX 6000 Ada (48 GB), RTX PRO 6000 Blackwell (96 GB), H100 (80 GB), and H200 (141 GB) GPUs. Hardware configurations vary across runs, while the effective batch size and the hyperparameters in Table 6 remain fixed. Training

Table 7: Ask-budget ablation on Valid-Seen: success rate (%) as a function of the inference-time question cap N. N=0 mutes asking entirely; N=∞ places no limit. $\pm$ denotes std over 3 seeds.

Model	S	B=0	B=1	B=3	B=5	B=∞
Qwen3-4B	41.7 ± 1.0	42.6 ± 0.5	47.0 ± 0.3	48.5 ± 0.6	48.6 ± 0.7	48.1 ± 0.5
Qwen3-8B	43.4 ± 1.4	42.7 ± 0.4	46.5 ± 1.0	47.5 ± 1.6	48.0 ± 1.2	47.6 ± 0.9
Qwen3-14B	42.8 ± 0.8	44.2 ± 1.0	49.0 ± 1.2	50.9 ± 0.5	51.5 ± 0.9	50.3 ± 1.1
Qwen3-32B	43.6 ± 1.5	43.2 ± 0.4	49.1 ± 1.0	50.1 ± 1.8	50.3 ± 1.3	50.1 ± 1.4
Llama3.1-8B	43.0 ± 0.7	43.6 ± 0.4	46.5 ± 0.7	48.2 ± 1.0	47.5 ± 1.2	45.8 ± 1.5
Ministral3-8B	41.6 ± 2.6	42.6 ± 1.1	43.9 ± 3.1	47.7 ± 2.1	46.7 ± 3.3	26.3 ± 17.2
Ministral3-14B	41.4 ± 0.6	41.7 ± 0.2	37.5 ± 11.1	40.5 ± 11.0	38.0 ± 13.6	19.9 ± 16.7
Gemma3-4B	39.6 ± 0.9	37.5 ± 0.5	32.6 ± 7.3	34.9 ± 6.1	31.5 ± 8.7	17.6 ± 12.6
Gemma3-12B	41.8 ± 1.2	41.4 ± 2.5	43.5 ± 2.7	44.7 ± 2.2	44.8 ± 2.8	35.6 ± 7.6
Gemma3-27B	40.9 ± 1.0	41.6 ± 0.8	46.2 ± 0.9	49.5 ± 2.5	50.7 ± 3.9	46.7 ± 2.2

uses BF16 without weight quantization, with gradient checkpointing enabled to reduce memory usage.

C Question Budget Analysis

Table 7 reports Valid-Seen task success for the **SQ** agents under question budgets $B \in \{0, 1, 3, 5, \infty\}$, alongside the corresponding **S** baselines. For each trained agent, we vary only the question budget while keeping the response oracle fixed. Setting $B = 0$ disables asking, whereas $B = \infty$ removes the question cap but retains the overall limit of 50 agent decisions per episode. Results are reported as mean and standard deviation over three seeds where available; entries based on fewer seeds are marked in the table.

D Trigger Calibration Details

Cross-split calibration. We calibrate all external triggers without using statistics from the split on which they are evaluated. Parameters for Valid-Seen are determined using Valid-Unseen, and parameters for Valid-Unseen are determined using Valid-Seen. For each calibration split, we run the frozen **SQ** agent with its native `Ask:` output disabled and greedy decoding, producing *never*-asking trajectories. These trajectories provide the decision prefixes and uncertainty signals used for calibration. No trigger parameter is subsequently adjusted using the evaluation split.

KnowNo-style conformal trigger. We adapt the conformal help-seeking mechanism of KnowNo (Ren et al., 2023) to our free-form action space. Whereas the original formulation operates over an explicit set of candidate actions, our agent directly generates free-form `Act` commands. We therefore approximate the candidate distribution by sampling $K = 16$ continuations from the same agent with temperature 1.0 and extracting the first predicted `Act` command from each continuation.

The calibration set is constructed from successful *never*-asking trajectories on the calibration split. Each assistant decision prefix forms one calibration instance, and the first `Act` executed in the successful trajectory is treated as the reference action y. Prefixes without a valid `Act` are excluded, and we sample up to 2,000 decision points for calibration. Let

$$f(c) = \frac{1}{K} \sum_{k=1}^{K} \mathbb{I}[c_k = c]$$

denote the empirical frequency of candidate action c among the sampled continuations. We define the nonconformity score of the reference action as

$$s = 1 - f(y).$$

Given n calibration instances, the conformal threshold $\hat{q}$ is the $\lceil (n+1)(1-\epsilon) \rceil / n$ empirical quantile of the calibration scores, using the higher-order statistic, with $\epsilon = 0.05$.

At evaluation time, we construct the prediction set

$$\mathcal{C} = \{c : f(c) \geq 1 - \hat{q}\}.$$

The agent proceeds without asking when $|\mathcal{C}| = 1$ and triggers a question otherwise. Unlike the rate-matched triggers below, this KNOWNO-style baseline is calibrated for conformal coverage rather than to match the learned gate's interaction rate.

Token-level uncertainty triggers. We additionally consider two lightweight uncertainty signals computed from the predicted free-form `Act` command. Let p_j denote the probability assigned to the jth token among the m tokens in the predicted action. We define

$$u_{\text{MeanNLL}} = -\frac{1}{m} \sum_{j=1}^{m} \log p_j,$$

$$u_{\text{MinProb}} = \min_{1 \leq j \leq m} p_j.$$

ACT-MEANNLL captures the average token-level negative log-likelihood, whereas ACT-MINPROB captures the confidence of the least-confident action token. These aggregations are motivated by prior work using token-level confidence to determine when external information should be acquired (Han et al., 2024; Jiang et al., 2023). A question is triggered when $u_{\text{MeanNLL}} \geq \tau$ for ACT-MEANNLL, or $u_{\text{MinProb}} \leq \tau$ for ACT-MINPROB.

For each trigger, the threshold τ is selected using only the calibration split. We first compute uncertainty signals at every eligible decision point in the corresponding *never*-asking trajectories. The target interaction rate is defined as the average number of questions per episode produced by the learned SQ gate on the same calibration split. For each candidate threshold, we simulate triggering over the *never*-asking decision sequences while enforcing the same interaction constraints as online evaluation, including a maximum of three questions per episode and at most one question per decision step. We select the threshold whose simulated questions-per-episode most closely matches the learned gate's interaction rate.

Random trigger. For the random baseline, each eligible decision point independently triggers a question with probability p. As with ACT-MEANNLL and ACT-MINPROB, p is selected on the calibration split to match the learned gate's average questions per episode while applying the same three-question budget and per-step interaction constraint. The resulting probability is fixed before evaluation on the opposite split.

Realized interaction rates. Although the token-level and random triggers are calibrated to match the learned gate's interaction rate on the calibration trajectories, their realized question frequencies can differ during online evaluation because asking changes the subsequent trajectory and therefore the distribution and number of future decision points. This discrepancy is particularly relevant for the KNOWNO-style trigger, whose calibration objective targets conformal coverage rather than interaction-rate matching. We therefore report the realized number of questions per episode together with task success in Table 3.

E RESPONDENT EVALUATION DETAILS

E.1 OFFLINE EVALUATION SETUP

We evaluate response sources on 432 held-out RESQ-Q questions, each paired with three independently collected human responses from RESQ-A; the held-out trajectories are disjoint from training. Model respondents receive the task instruction, interaction history up to the question, the question and uncertainty type, and the recorded privileged simulator context used during response collection.

We compare the human RESQ-A responses, the Qwen3-4B RESQ oracle fine-tuned on the training split and decoded greedily, and GPT-5.5 under zero-shot and four-shot prompting. For four-shot prompting, we retrieve four training questions by TF–IDF similarity over the question text and prepend their full context panels and corresponding human responses.

E.2 OFFLINE METRICS

We characterize respondents along three complementary axes: pairwise agreement with human response polarity (Pol.), factual support for the direct answer (AnsFact), and response scope (Extra).

Polarity agreement (Pol.). A response expresses polarity when it begins with an explicit "yes" or "no." For each model response, we compare its polarity with each human response to the same question when both express polarity; Pol. is the fraction of matching pairs. The Human row applies the same procedure between annotators and reports inter-annotator agreement (89.0% over 501 comparable pairs). Because humans often answer implicitly, this metric is conditioned on expressed polarity; restricting evaluation to the 231 questions with at least one comparable human–human pair changes all system scores by less than 0.7 points.

Answer factuality (AnsFact). AnsFact evaluates only the direct answer—the portion that answers, clarifies, or directly corrects the current question, including brief justification or information necessary to identify or reach the queried entity. Claude Opus 5 judges whether this direct answer is supported by, or logically follows from, the recorded privileged context and task goal; empty responses or responses without a direct answer are scored false.

Response scope (Extra). Extra indicates whether a response contains task-relevant information beyond what is needed to resolve the current question, such as additional state information, planning guidance, future actions, or later subgoals. Brief justification, information necessary to identify or reach the queried entity, direct correction, and the first action when the agent explicitly asks what to do next are not counted. Extra is descriptive rather than a quality metric.

We use Claude Opus 5 as the primary judge for AnsFact and Extra because GPT-5.5 is itself one of the evaluated response sources. Repeating the evaluation with GPT-5.5 as the judge yields 89.4% agreement on AnsFact and 91.7% on Extra, while preserving the respondent-level patterns reported in the main text. Each human response is judged independently, and the Human row reports the mean across the three annotators.

Listing 3: Prompt for LLM-based respondent evaluation.

```
# Role
You are evaluating a response to a household robot's question.
The respondent has access to the privileged context provided below.

Evaluate only the response to the robot's current question.
Use only the provided privileged context as evidence.
Do not assume task-specific facts that are not provided.
Do not reward or penalize length or style.

# Response Decomposition
First, mentally split the response into two parts:

- DIRECT ANSWER: the part that answers, clarifies, or directly corrects
    the issue raised in the robot's current question, including a brief
    justification and information necessary to identify or reach the
    entity asked about.

- ADDITIONAL PART: everything beyond the DIRECT ANSWER, including
    additional state information, planning guidance, future actions, or
    later subgoals.

# Criteria

## 1. answer_factual
Is the DIRECT ANSWER correct given the privileged context?

Return true if its task-specific factual claims, corrections, and
    guidance are supported by, or logically follow from, the privileged
    context and the task goal.
```

Table 8: Offline respondent evaluation on 432 held-out questions (%). Claude Opus 5 is the primary judge for AnsFact and Extra; GPT-5.5 provides cross-judge validation. Pol. is judge-independent.

		Claude Opus 5		GPT-5.5	
Respondent	Pol.	AnsFact	Extra	AnsFact	Extra
Human	89.0	84.3	57.2	75.7	57.6
RESQ oracle	85.6	87.0	31.0	78.7	30.6
GPT-5.5 0-shot	78.8	95.8	51.2	94.9	46.1
GPT-5.5 4-shot	80.0	97.7	58.1	94.7	56.7

```
Judgments that logically follow from the task goal also count as
    supported. For example, stating that a detail does not matter is
    supported when the task goal places no constraint on that detail.

Return false if any claim in the DIRECT ANSWER conflicts with the
    privileged context or asserts information that is not supported by
    it.

Return false if the response does not provide a direct answer to the
    question, or is empty.

General statements that make no task-specific factual claim do not by
    themselves make the direct answer non-factual.

## 2. extra
Does the response contain an ADDITIONAL PART as defined above?

Return true if it provides task-relevant information or guidance beyond
    what is needed to resolve the robot's current question.

Do NOT count as extra:
- a brief explanation that only justifies the direct answer,
- information necessary to identify or reach the entity asked about,
- a direct correction necessary to resolve the current question, or
- the first action when the robot explicitly asks what it should do next.

Return false if the response stays within the DIRECT ANSWER.

# Output Format
Return ONLY valid JSON in exactly this format:
{"answer_factual": true|false, "extra": true|false}
```

E.3 FULL OFFLINE RESULTS

The two judges differ in absolute AnsFact scores but preserve the main patterns: GPT-5.5 receives higher judge-rated AnsFact, the RESQ oracle shows the highest Pol. among model respondents, and the oracle provides additional information substantially less often. Human Extra also varies considerably across annotators (34.5/68.1/70.3 under the GPT-5.5 judge), indicating substantial variation in human response scope.

Polarity–factuality disagreement. We manually inspect 38 questions in which GPT-5.5 0-shot contradicts the human polarity consensus despite receiving a factual direct-answer judgment. Deictic confirmation questions form the largest category (17/38), such as "Is *this* the bowl you meant?", where resolving the referent from the task goal rather than the preceding interaction history can reverse the answer. The GPT-5.5 judge exhibits a stronger asymmetry in favor of GPT-5.5 responses on these cases than the Claude judge, further motivating our use of Claude as the primary judge and Pol. as a complementary judge-independent measure.

Table 9: Online respondent comparison per cell on Valid-Seen (SR | Conv, %; mean±std over three training seeds).

Agent	Size	ReSQ oracle	GPT-5.5 0-shot	GPT-5.5 4-shot
SQ	4B	48.5±0.5 | 19.5±2.7	49.0±1.0 | 21.9±3.1	48.5±0.9 | 19.4±3.9
SQ	8B	47.5±1.6 | 18.4±4.2	49.3±2.5 | 24.7±6.7	48.3±1.6 | 21.3±3.8
SQ	14B	50.9±0.5 | 25.4±1.7	51.9±0.7 | 29.6±0.8	50.9±0.9 | 26.2±1.2
SQ	32B	50.1±1.8 | 23.4±4.8	51.7±1.9 | 28.7±4.8	51.4±1.7 | 28.0±3.8
SQR	4B	52.4±0.9 | 29.5±0.4	52.7±1.2 | 30.5±2.7	52.4±1.5 | 29.1±3.3
SQR	8B	51.3±0.9 | 28.1±0.5	53.3±1.1 | 35.8±1.5	51.8±1.5 | 30.3±3.1
SQR	14B	53.0±0.3 | 27.3±2.0	52.9±1.5 | 29.0±1.6	52.6±1.9 | 28.1±0.5
SQR	32B	53.8±2.1 | 29.7±5.1	54.7±1.5 | 31.4±2.4	53.7±1.2 | 29.6±1.8

Human reference. Human AnsFact is a reference rather than a direct estimate of human accuracy because ReSQ-A annotators could inspect the simulator when the initially provided context was insufficient, whereas the offline judge sees only the recorded context panel. We therefore do not interpret GPT-5.5's higher AnsFact as evidence that it is more accurate than human respondents.

F Online Respondent Evaluation Details

F.1 Online Evaluation Setup

Online comparisons keep the agent, its greedy decoding, and the question budget fixed and swap only the model that answers the agent's questions. Because decoding is greedy, an episode's trajectory is identical across respondents until the first question, and whether the agent asks is decided before any answer is received. Episodes without a question are therefore unaffected by the respondent; for each agent (model size $\times$ training seed) we re-run only the episodes in which the agent asks (226–293 of 820 Valid-Seen episodes) and carry over the remaining outcomes from the main run. The set of asked episodes is identical across respondents within a seed, so Conv is computed on the same episodes for every respondent. Zero-shot and four-shot prompts are identical to the offline evaluation (Appendix E.1); GPT-5.5 is sampled at its API default, and a matched ReSQ-oracle run at temperature 1.0 changes its results within seed noise.

F.2 Full Online Results

Table 9 reports SR and Conv for every model size and training setting; Table 4 in the main text averages each column over the four sizes.

Four-shot prompting. Four-shot GPT-5.5 yields lower SR and lower Conv than zero-shot GPT-5.5 in all eight cells, so the averaged differences in the main text reflect a consistent effect rather than aggregation, even though individual cells lie within seed variation. Offline, the two prompts differ in scope (Extra 58.1 vs. 51.2) but not in answer factuality (97.7 vs. 95.8), so the online gap isolates the effect of providing information beyond the question.

Respondent invariance. Under every respondent, the ordering of SQ model sizes is the same (14B $\approx$ 32B $>$ 4B $\approx$ 8B in both SR and Conv), and the change from SQ to SQR has the same sign and comparable magnitude for a given size (Conv at 4B: $+10.0$, $+8.6$, $+9.7$ for the ReSQ oracle, zero-shot, and four-shot GPT-5.5; at 14B: $+1.9$, -0.6, $+1.9$). Changing the respondent shifts any cell by at most 2.0 points in SR and 7.7 in Conv without changing which agent is better or whether recovery supervision helps.

Convergence under recovery supervision. Under SQ, Conv rises with model size for every respondent, whereas under SQR it collapses to a narrow band independent of size (27.3–29.7 for the ReSQ oracle, 28.1–30.3 for four-shot GPT-5.5, 29.0–31.4 for zero-shot GPT-5.5 excluding 8B). The SQR gain is therefore largest where the SQ starting point is lowest (4B, 8B) and near zero at 14B, whose SQ Conv already lies in the band. The 8B agent with zero-shot GPT-5.5 (SQR Conv 35.8)

is the one cell above its band; we note its large SQ-level seed variance (6.7) but offer no further explanation.

Flip. Agents ask predominantly on episodes they cannot solve alone, so the Conv denominator $|U_0|$ is large (665–769 pooled over seeds) while the Flip denominator $|S_0|$ is small (21–75), and a single episode moves Flip by 1.3–4.8 points. We therefore report only Conv for respondent comparisons; the observation that asking can break otherwise-solved episodes holds for every respondent. Which episodes convert is dominated by trajectory-level stochasticity: conversion-set overlap between different respondents (Jaccard 0.48–0.59) is indistinguishable from that between sampling seeds of the same oracle (0.37–0.55), and 61–68% of U_0 episodes are converted by no respondent.