2.2.3 Assistant Utterance Generation

Assistant-side responses are generated iteratively in a role-playing setup, where one LLM assumes the *assistant role* and another the *user role*. For each sub-plan, the assistant LLM is conditioned on the conversation seed (Section 2.2.1), prior sub-plans, a summary of the last M turns, and a compressed summary of earlier ones (using the prompt in Listing 37 in Appendix B); for 10M-token conversations, additional summaries of prior plans are provided. The assistant first generates a response to the user's most recent question (line 9 in Algorithm 3 in Appendix B.3.5), which is analyzed by a *question-detection module* (line 11 in Algorithm 3 in Appendix B.3.5, using the prompt in Listing 35 Appendix B) to determine the presence of a counter-question. If detected, the response is passed to the user LLM, which generates a contextually consistent reply based on the current and prior sub-plans, relevant history, and conversation summaries (using the prompt in Listing 38 in Appendix B, line 14 in Algorithm 3 in Appendix B.3.5). This loop continues until no further assistant questions are detected or the threshold $\delta_1 = 2$ is reached, balancing realism and avoiding infinite cycles. In addition, a *follow-up detection module* (line 21 in Algorithm 3 in Appendix B.3.5, using the prompt in Listing 36 in Appendix B) evaluates whether a clarifying or elaborative user follow-up is warranted, based on factors such as subject complexity, ambiguity, or incomplete responses. When required, the module generates a follow-up query conditioned on the seed, current and prior sub-plans, the most recent M turns, and earlier summaries (using the prompt in Listing 39 in Appendix B), which is then passed back to the assistant LLM. The number of follow-up exchanges is limited by a threshold $\delta_2 = 2$, analogous to δ_1. Together, these modules yield dialogues with bidirectional dynamics, contextual referencing, and realistic clarifications, approximating human–AI interactions. The details of this procedure are provided in Appendix B.3.4.

2.3 Probing Questions Generation

After constructing conversations, we generate probing questions to evaluate memory abilities. The pipeline combines automated synthesis with human validation: an LLM first produces candidate probes, which annotators review to select valid ones. Probes are derived from both the conversation plan and chat to ensure each targets a specific ability, is grounded in dialogue turns, and includes explicit provenance. The process begins by passing the plan to GPT-4.1-mini (OpenAI, 2025b), which selects candidate bullet points conditioned on the ability under evaluation. For example, knowledge-update probes require bullet pairs encoding an initial fact and its later revision, while summarization and event-ordering probes span multiple bullets. Each bullet is linked to its corresponding user and assistant turns through indices introduced during user-assistant turn generation, enabling retrieval of the precise dialogue segments in which the content was created. Candidate bullet selection is performed using prompts 1–9, one per memory ability. For abstention, candidate selection is unnecessary; probes are created directly from the plan using the prompt shown in Listing 14 (Appendix G).

Given the selected bullet points and aligned dialogue snippets, GPT-4.1-mini generates the probing question, a candidate answer, and source identifiers citing the specific messages containing the answer. For 10M-token dialogues, candidate selection and synthesis are performed with a sliding window across the ten interlocking plans, processing a limited number at a time to preserve topical locality and scalability. Probe generation uses prompts 10–19 for each memory ability, mapping candidate bullet points and contexts into fully formed questions. Finally, a human evaluator reviews the generated candidates and selects those that are valid and consistent with the conversation. Samples of probing questions are provided in Appendix D, items 1–10.

2.4 Evaluation

We evaluate LLMs on the probing questions using nugget evaluation, a common approach for long-form text assessment (Pradeep et al., 2024; 2025). Each probing question is manually validated: invalid or unsupported questions are discarded, and minor inconsistencies are corrected. From the validated set, two questions per memory ability are chosen for each conversation, yielding 20 probing questions per conversation. Rubric nuggets are then derived for each question. A nugget is an atomic, self-contained criterion that a system response must satisfy. Annotators decompose the ideal reference answer into minimal semantic units, ensuring each nugget is both atomic and self-contained. System responses are scored against these nuggets by an LLM judge (Listing 20,