ADAR: LEARNING TO SEARCH FOR IMPLICIT PREFERENCES IN DIALOGUE MEMORY

Anonymous authors
Paper under double-blind review

ABSTRACT

Retrieving a user's implicit preferences requires finding supporting dialogue turns that may share few words with the current request. We study how evidence annotations available during training can teach a compact model to search for such indirect support without annotations at inference. We introduce ADAR, a dialogue search agent trained on restricted-oracle demonstrations: a teacher uses annotated evidence to explore search trajectories, filters reject paths with overt leakage, incorrect answers, or missing evidence, and the student learns from the resulting observable traces. During search, an Abductive Distance Score (ADS) summarizes lexical coverage and semantic competition as interpretable feedback alongside retrieved text. Under a shared backbone, ADAR improves accuracy over the strongest evaluated search baseline by 14.94 percentage points on PersonaMem-v2, 4.38 on held-out LongMemEval, and 2.63 on LoCoMo. Withholding numerical ADS feedback from the same checkpoint while preserving autonomous stopping lowers accuracy on all three datasets. These results support converting source evidence annotations into search supervision that improves preference answering and transfers to new dialogue histories. Our code is available at https://anonymous.4open.science/r/adar27-E72B.

1 INTRODUCTION

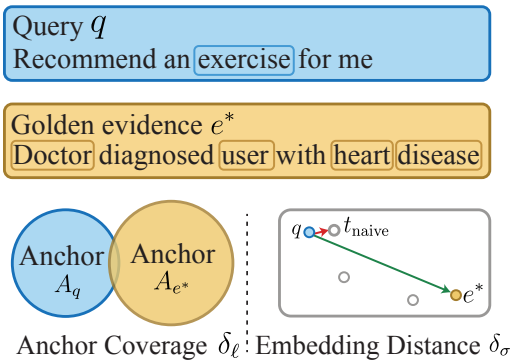
A dialogue assistant may remember what a user said yet fail to retrieve what a new request makes relevant. Consider a user asking for exercise advice. Earlier mentions of swimming or fitness reveal interests, but another turn may say, "My doctor diagnosed me with a heart condition." That turn need not mention exercise, yet it may constrain which recommendations are appropriate. The assistant must connect the request to relevant personal circumstances and retrieve the supporting turns. This illustrative example captures the shift from factual recall to implicit personalization (Maharana et al., 2024, Wu et al., 2025, Jiang et al., 2025b, Zhao et al., 2025).

We study **query-time retrieval for implicit-preference answering**. The input is a request and a turn-indexed dialogue history. The output is an answer supported by retrieved turns. The central learning problem is how to choose productive search directions when the request does not name its supporting evidence. This makes the route from a request to its supporting turns a central target of supervision.

The challenge is to infer which parts of the history could matter, then verify them through retrieval. In this example, relevant health constraints are a possible search direction. The diagnosis is explicit in the history, while its relevance to exercise advice must be inferred. We use *abductive* to describe this hypothesis-guided search for support. Teaching that connection is the purpose of ADAR's trajectory supervision.

Existing memory systems organize histories before a particular query arrives (Packer et al., 2024, Zhong et al., 2024, Xu et al., 2025, Chhikara et al., 2025, Fang et al., 2025). Active search methods instead reformulate queries and interleave retrieval with reasoning (Trivedi et al., 2023, Li et al., 2025, Jin et al., 2025, Zheng et al., 2025). Both are useful starting points. Our question is what supervision helps an active retriever bridge indirect query-to-evidence relationships under a fixed deployment model. A terminal answer reward evaluates the end of a search, whereas a demonstration also specifies intermediate queries and stopping decisions. ADAR uses this intermediate structure to teach the search process directly.

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.



Lexical gap. A request about exercise need not share informative anchors with a supporting medical-history turn. The lexical term measures missing anchor coverage.

Semantic competition. A superficially similar turn can outrank the supporting evidence. The semantic term compares evidence similarity with the strongest distractor.

Figure 1: Illustration of the two ADS components. Boxed terms denote anchors. The evidence e^* and a competing turn t_{naive} illustrate why lexical coverage and semantic competition are distinct.

Abductive Distance-Aware Retrieval (ADAR) converts source evidence annotations into demonstrations of how to search. A teacher explores candidate trajectories with access to annotated support, and filtering selects paths that retrieve evidence and produce correct answers while rejecting overt leakage. The student learns from observable traces with privileged inputs removed. This provides intermediate supervision for query formulation and evidence assessment, which the deployed model must perform using unannotated dialogue history.

To give the search policy a compact view of its current retrieval state, we combine missing informative query terms with the reference set's semantic margin over competing turns (Figure 1). Applied to the accumulated retrieval pool, this score supplies feedback without evidence labels. Applied to annotated evidence, it also describes query-to-evidence indirectness offline. The policy reads the score alongside the retrieved content when choosing another search or producing an answer.

Our contributions are threefold:

1. **Evidence-supervised search for implicit preferences.** We train a compact agent on filtered teacher demonstrations, converting evidence annotations into search behavior that retrieves supporting dialogue turns without gold labels at deployment.
2. **Interpretable feedback during search.** We formulate the Abductive Distance Score (ADS) to summarize lexical coverage and semantic competition in the growing retrieval pool. Annotated evidence also permits offline analysis of this score.
3. **Effectiveness, transfer, and feedback utility.** ADAR outperforms shared-backbone search baselines on three benchmarks, including held-out LongMemEval. Same-checkpoint ablations support the utility of ADS feedback while preserving autonomous stopping.

2 Evidence-Supervised Dialogue Search

2.1 Task and Framework Overview

Figure 2 summarizes the pipeline. An instance contains a query q, a history $H = \{t_1, \ldots, t_n\}$ of dialogue turns, and a reference response y^*. Annotated source instances additionally provide an evidence subset $E^* \subseteq H$. A policy issues search queries u_r, retrieves turns, accumulates a deduplicated pool P_r, and chooses to continue or answer.

Gold evidence and answers support offline analysis and demonstration construction. Privileged input fields are removed before student SFT. Deployment uses only the query, unannotated history, and retrieval feedback, with evaluation references restricted to the external judge.

Both stages share an interaction loop: formulate a query, retrieve turns, inspect evidence and feedback, then search again or answer. Section 2.2 describes trajectory construction and student training, Section 2.3 defines the accompanying feedback, and Section 2.4 specifies online search decisions.

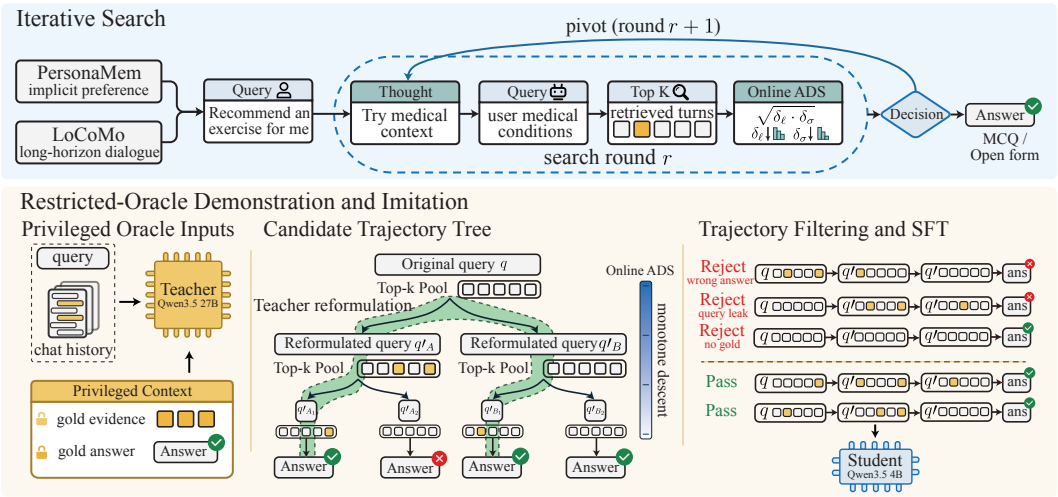


Figure 2: Overview of ADAR. Top: query-time iterative retrieval, online ADS feedback, and commit/pivot decisions. Bottom: a 27B teacher uses privileged source annotations to explore shared-prefix trajectories. Leakage, outcome, and evidence-presence filters select demonstrations for the 4B student. Privileged inputs and the search tree are used only offline. The benchmark labels illustrate the common dialogue-retrieval interface.

2.2 Offline Trajectory Construction and Student Training

Offline training consists of teacher exploration, trajectory filtering, and student SFT. The 27B teacher follows the search-and-decision protocol in Section 2.4, with additional source annotations used only during construction. The teacher turns annotated support into examples of intermediate queries, retrieved observations, and decisions. The student then learns how to progress from a request to supporting turns using the information available at deployment.

Restricted oracle and branching. A private `BACKGROUND` block gives the teacher the reference preference/answer and a short supporting excerpt. Instructions prohibit quoting, paraphrasing, or revealing this block in emitted text. At each search state, two sampled query-generation actions create alternative branches that reuse the parent prefix. Root-to-commit paths form candidate demonstrations. Branching exposes alternative continuations from a shared observable context while reusing previous generations. Branching is confined to offline construction. Privileged fields are removed and prompts replaced with their student-facing versions before SFT, so each demonstration presents the interaction available to the deployed policy. The selection constraints below screen the emitted trajectory as well as its final answer.

Three selection constraints. The anti-leak filter checks assistant messages against eight forbidden-phrase families and rejects search queries with at least two multi-character gold-excerpt tokens absent from the original query. The outcome filter rejects answers judged incorrect under the relevant MCQ or open-form protocol. The evidence-presence filter rejects a path unless its commit-time pool contains at least one annotated evidence turn. Together, these criteria connect the training target to three requirements: observable search behavior, a correct judged answer, and retrieval of annotated support. No explicit trajectory-length filter is applied.

The evidence criterion matters because an answer label alone leaves the route to that answer unspecified. A model may produce a plausible response while retrieving only topically related turns. Such a path offers little supervision for finding the personal information that makes the response appropriate. Requiring annotated support in the pool makes successful evidence acquisition part of trajectory selection. Section 3.3 examines how often outcome-passing candidates fail this additional criterion.

Student objective. The surviving demonstrations form the training corpus $\mathcal{D}_{\text{filtered}}$. The student minimizes assistant-token negative log-likelihood on the observable traces,

$$\mathcal{L}_{\text{SFT}}(\theta) = -\sum_{\tau \in \mathcal{D}_{\text{filtered}}} \sum_{j \in \mathcal{A}(\tau)} \log p_\theta(x_j \mid x_{<j}), \tag{1}$$

where $\mathcal{A}(\tau)$ indexes assistant tokens. Retrieved observations and user/system inputs are masked from the loss. Supervision covers query formulation, evidence assessment, and commitment. Each query is learned in the context of the request and observations preceding it. Later actions are conditioned on the evidence obtained along the path, allowing the same objective to teach both reformulation after retrieval and commitment when the teacher judges the pool sufficient.

The transferable object is therefore a sequence of search decisions. Source annotations guide demonstration construction, while the student learns the observable interaction with the retrieval environment. At deployment, the policy can apply these decisions to a new history through the same turn-retrieval interface. Held-out LongMemEval tests this transfer in Section 3.2. The baseline comparison evaluates this supervision recipe under matched student capacity, with training conditions specified in Section 3.1.

2.3 ADS AS RETRIEVAL FEEDBACK

The search policy receives the retrieved text together with a compact numerical description of its coverage and competition. ADS combines two signals: informative query terms missing from a reference set E, and the similarity of its best match relative to competing history turns (Figure 1).

Let A_x contain the named entities, content-noun lemmas, and content adjectives extracted by spaCy (Honnibal et al., 2020), lowercased and stop-word filtered, and let $A_E = \bigcup_{t \in E} A_t$. Using smoothed IDF weights over a corpus of query and turn anchor sets, define missing lexical coverage as

$$\delta_\ell(q, E) = \frac{\sum_{w \in A_q \setminus A_E} \text{idf}(w)}{\sum_{w \in A_q} \text{idf}(w)}. \tag{2}$$

Here $\text{idf}(w) = \log((N+1)/(\text{df}(w)+1))$, where N is the number of anchor sets in that corpus. Each query anchor is counted once, so repeated words receive no term-frequency factor. For unit-normalized Qwen3-Embedding-0.6B embeddings (Zhang et al., 2025), the semantic term compares the best in-set and out-of-set matches,

$$s_{\text{in}}(q, E) = \max_{t \in E} \cos(\text{emb}(q), \text{emb}(t)), \qquad s_{\text{out}}(q, E) = \max_{t \in H \setminus E} \cos(\text{emb}(q), \text{emb}(t)), \tag{3}$$

$$\delta_\sigma(q, E) = 1 - \big(s_{\text{in}}(q, E) - s_{\text{out}}(q, E)\big), \qquad \text{ADS}(q, E) = \sqrt{\delta_\ell(q, E)\, \delta_\sigma(q, E)}. \tag{4}$$

IDF emphasizes informative anchors, while the semantic margin captures competition that absolute cosine omits. The geometric mean is large when both coverage and separation are poor. A small value on either axis suppresses the aggregate. These complementary axes make the retrieval state interpretable and provide a concise feedback signal.

The reference set determines the use of the score. With annotated evidence $E = E^*$, ADS describes indirectness for offline analysis. With the accumulated pool $E = P_r$, it provides online feedback without evidence labels. Pool expansion makes ADS nonincreasing on its defined domain, so the agent assesses answer support from the retrieved content and uses ADS as an accompanying signal. Commitment remains a learned decision. The expression assumes positive anchor weight, nonempty sets in the maxima, and a nonnegative semantic term. Appendix A gives the domain, monotonicity proof, and interpretation details.

2.4 ONLINE SEARCH AND DECISION MAKING

The Qwen3.5-4B student (Qwen Team, 2026) executes the learned protocol without the teacher, gold annotations, or branching. Each round has two phases. In the search phase, it emits a short `THOUGHT` and a search `QUERY` u_r. The retriever selects the top five history turns by embedding similarity to u_r, and newly retrieved turns are added to the deduplicated pool P_r. The student receives these observations together with $\text{ADS}(q, P_r)$.

In the decision phase, the student emits a content-based `judgment` and chooses `commit` or `pivot`. A commit produces an answer from the available evidence. A pivot starts another search round using the accumulated observations. Commitment can occur at any round, with a common maximum of five rounds and mandatory commitment at that cap. There is no fixed ADS stopping threshold. MCQ options are hidden throughout search and used only for final answer mapping.

The retrieval query u_r can change between rounds, but ADS is always computed relative to the original request q. Similarity to u_r assesses the current search direction, while ADS describes the accumulated pool against a fixed information need. In the exercise illustration, an initial search can retrieve swimming or fitness discussions. The subsequent decision concerns whether those turns provide enough personal context for the request. A further query about relevant health circumstances could retrieve the disclosed diagnosis and change the final response. This illustrates the behavior taught by the trajectories: use the current observations to choose a next search direction or commit to an answer. The score and retrieved text inform that decision together. Caching and feedback-computation details are in Appendix A, with stage-level resource use in Appendix G.

3 EXPERIMENTS

We organize the experiments around three questions, progressing from overall effectiveness to the contribution of the training and inference design, then to the interpretation of ADS:

RQ1: Does ADAR improve dialogue-memory answering and transfer beyond its training sources?

RQ2: Do trajectory supervision and online ADS feedback contribute to answer accuracy?

RQ3: What retrieval conditions does ADS describe beyond its online use?

3.1 EXPERIMENTAL SETUP

Datasets and scope. PersonaMem-v2 evaluates implicit preferences through four-choice questions (Jiang et al., 2025b). We retain the five retrieval-focused types, approximately 69% of the benchmark, excluding sensitive-information redaction and requests to forget. LoCoMo (Maharana et al., 2024) uses a conversation-disjoint split following Mem0 (Chhikara et al., 2025), with one conversation used for synthesis. These two benchmarks provide the annotated sources for teacher demonstrations. LongMemEval (Wu et al., 2025) contributes 470 non-abstention questions out of 500 and is entirely held out from synthesis and student training. All methods within a comparison use the same selected evaluation set. The reported scores cover these retrieval subsets, detailed in Appendix F.

Models, training, and budgets. The primary comparison uses Qwen3.5-4B for ADAR, Native RAG, IRCoT, Search-o1, Search-R1, and DeepResearcher, with a shared Qwen3-Embedding-0.6B retriever. ADAR is fully fine-tuned for two epochs at learning rate 10^{-5}, effective batch size eight, and context length 12,288. The two outcome-RL baselines use full-parameter GRPO (Shao et al., 2024) for 500 steps with learning rate 10^{-6} and four rollouts per step on the same source training corpus. They share ADAR's five-round, top-five search cap. Native RAG uses one top-ten retrieval. IRCoT uses up to five top-three retrievals. The trained search agents thus share a retrieval cap, while the training-free methods retain their specified retrieval schedules. Supplementary adaptations tested dialogue-memory prompts for Search-R1 and replaced web search with history retrieval plus format retry for DeepResearcher. The comparison evaluates training recipes at a shared deployment capacity. ADAR uses intermediate trajectory supervision, while the RL baselines optimize terminal outcomes. Training compute is not matched.

Answer evaluation and uncertainty. PersonaMem-v2 hides answer options during search and uses an MCQ letter judge for the final answer. LongMemEval and LoCoMo use a frozen Qwen3.5-122B YES/NO judge. Both judges score the answer without seeing the search transcript, using the prompts in Appendix J. Search-agent results are reported as mean $\pm$ standard deviation over three decoding seeds. The open-form scores reflect consistency with the supplied reference under this judging protocol, rather than benchmark-official factual QA scoring.

Table 1: Accuracy (%) on the selected retrieval subsets. Parentheses give the rank by mean accuracy within each dataset across all listed methods (1 is best). The search-agent comparison uses a shared 4B backbone.

Method	Approach	PersonaMem-v2	LongMemEval	LoCoMo
Search-agent baselines				
Native RAG	Training-free	29.32 ± 1.38 (4)	44.02 ± 1.48 (3)	29.91 ± 0.88 (5)
IRCoT	Training-free	32.37 ± 3.51 (2)	37.90 ± 2.59 (5)	30.60 ± 1.50 (4)
Search-o1	Training-free	26.71 ± 2.05 (6)	27.63 ± 0.64 (6)	27.67 ± 0.94 (6)
Search-R1	Outcome RL	16.64 ± 0.66 (8)	14.74 ± 2.25 (8)	22.95 ± 0.42 (8)
DeepResearcher	Outcome RL	21.63 ± 0.68 (7)	19.44 ± 3.23 (7)	33.31 ± 1.26 (3)
Memory-system baselines				
AssoMem	Memory system	26.77 ± 3.77 (5)	41.43 ± 5.07 (4)	34.51 ± 3.45 (2)
LightMem	Memory system	31.50 ± 3.99 (3)	46.27 ± 5.20 (2)	23.48 ± 2.80 (7)
ADAR	Trajectory SFT	$\mathbf{47.31} \pm 2.42$ (1)	$\mathbf{48.40} \pm 3.57$ (1)	$\mathbf{35.94} \pm 0.57$ (1)

3.2 RQ1: Does ADAR improve answering and transfer?

To answer RQ1, we compare answer accuracy with search-agent and memory-system baselines (Table 1). PersonaMem-v2 tests the target implicit-preference task, while LoCoMo and held-out LongMemEval test broader dialogue-memory performance. ADAR reaches 47.31% on PersonaMem-v2, compared with 32.37% for IRCoT. On LongMemEval it reaches 48.40%, versus 44.02% for Native RAG, without using target-dataset annotations for training. The LoCoMo improvement is smaller: 35.94% versus 33.31% for DeepResearcher. Relative to the strongest search-agent baseline on each dataset, the gains are 14.94, 4.38, and 2.63 percentage points.

The memory-system rows extend this comparison to AssoMem (Zhang et al., 2026) and LightMem (Fang et al., 2025). ADAR numerically exceeds both on all three tasks, with narrower margins of 2.13 points over LightMem on LongMemEval and 1.43 points over AssoMem on LoCoMo. Appendix I gives the category breakdowns.

Which capabilities improve? The category results in Appendix I make the aggregate gains more concrete. ADAR exceeds the strongest search baseline in each of the five retained PersonaMem-v2 categories, with gains ranging from 7.20 to 16.67 points. The benefit therefore spans different preference contexts rather than a single category. On held-out LongMemEval, ADAR reaches 43.43% on multi-session questions and 33.61% on temporal questions, versus the strongest search-agent scores of 37.37% and 20.17%. These gains of 6.06 and 13.44 points show transfer relative to search agents on questions involving information across sessions or time. On LoCoMo, it also improves single-hop, multi-hop, and temporal scores, although multi-hop accuracy remains only 9.34%.

Other baselines lead LoCoMo's open-domain category and LongMemEval's single-session and knowledge-update categories. The gains are concentrated in preference answering and questions involving information across sessions or time. This pattern connects the source task to the transfer setting: both can require finding relevant personal context beyond the most direct query match. RQ1 thus supports the usefulness of the learned search policy across these settings. RQ2 examines the training and inference choices behind that policy.

Sensitivity to the evaluation judge. ADAR remains above DeepResearcher with both GLM5 and DeepseekV4 on LongMemEval and LoCoMo, although absolute scores and the LoCoMo margin vary. Appendix D reports this pairwise judge-sensitivity check.

3.3 RQ2: Do trajectory supervision and ADS feedback help?

RQ1 evaluates the complete system. To answer RQ2, Table 2 separates two comparisons: removing student SFT tests the training recipe, while withholding ADS from a fixed checkpoint tests feedback during inference. The filtering statistics then examine which demonstrations enter training, and online plots describe score behavior during search.

Table 2: Accuracy (%) for supervision and feedback comparisons (top) and component comparisons (bottom). Parentheses show relative decreases from Full ADAR, computed as $100(a_{\text{best}} - a)/a_{\text{best}}$. Without online ADS feedback, the same checkpoint receives no numerical ADS feedback but retains autonomous commit/pivot decisions and the five-round retrieval cap.

Variant	PersonaMem-v2	LongMemEval	LoCoMo
Full ADAR	47.31 (0.00%)	48.40 (0.00%)	35.94 (0.00%)
Without online ADS feedback	40.06 (↓15.32%)	45.43 (↓6.14%)	31.76 (↓11.63%)
Without student SFT	30.33 (↓35.89%)	42.92 (↓11.32%)	28.67 (↓20.23%)
Lexical feedback only	44.52 (↓5.90%)	47.71 (↓1.43%)	33.56 (↓6.62%)
Semantic feedback only	42.16 (↓10.89%)	46.63 (↓3.66%)	34.46 (↓4.12%)

Trajectory supervision. Removing student SFT lowers accuracy by 16.98 points on PersonaMem-v2, 5.48 on LongMemEval, and 7.27 on LoCoMo. The largest change occurs on the target preference task, and the reduction also extends to the held-out dataset. These results support training on demonstrations of the search process. The training target includes the actions that precede the answer: how to reformulate a query after observing retrieved turns, assess the available support, and decide when to commit. The comparison measures the value of this trajectory-training recipe as a whole.

Evidence-based trajectory selection. The selection funnel provides a closer view of the demonstrations used to teach that behavior. In a diagnostic sample of 300 candidates, the filters reject 121 for leakage, a further 57 for incorrect answers, and 28 for missing evidence, leaving 94 survivors. Relative to this sampled pool, these are 40.3%, 19.0%, 9.3%, and 31.3%.

The decisive observation is that 28 of the 122 candidates passing the first two checks, approximately 23%, still fail to retrieve annotated support. Selecting demonstrations solely by their judged answers would retain these trajectories. Evidence-based selection instead requires a successful search to appear in the demonstration, giving the student examples of reaching relevant history turns as well as producing an accepted answer. This distinction explains why the training pipeline checks evidence acquisition separately from answer correctness. Appendix E gives source-level counts and the scope of the leakage inspection.

Online ADS feedback. To test feedback in the online loop of Section 2.4, we omit numerical ADS from the same trained checkpoint and retrieval interface (Table 2). The agent still reads retrieved turns and decides autonomously whether to commit or pivot. Both conditions retain top-five retrieval, a five-round cap, and early commitment. Withholding feedback lowers accuracy by 7.25, 2.97, and 4.18 percentage points on PersonaMem-v2, LongMemEval, and LoCoMo, respectively. Because the checkpoint and decision options are preserved, this comparison tests the usefulness of numerical feedback to the deployed policy. The gain occurs while the agent remains free to answer early or search again, linking ADS to the learned search process rather than an externally imposed stopping schedule.

The score contributes no new turn or answer content. It compresses properties of the current pool that would otherwise have to be inferred from many retrieved turns and their relation to the remaining history. The student still has to read those turns to assess support. The comparison shows that the trained policy benefits from the feedback interface, but does not identify whether the benefit comes primarily from query reformulation, evidence assessment, or commitment timing.

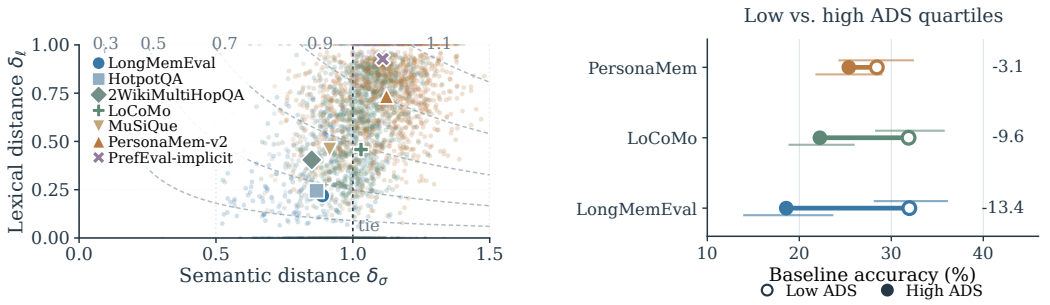
Feedback components. Joint lexical and semantic feedback has the highest reported accuracy among the component variants on all three datasets. We treat these rows as descriptive comparisons and use the same-checkpoint removal of all numerical feedback for the controlled test of online feedback utility.

Online behavior. The trajectory plots in Appendix C show how feedback evolves over rounds and how accuracy varies with commit-time ADS. Pool expansion explains the descending scores, while the final-score groups describe their empirical association with answer quality. The same-checkpoint removal above directly tests whether providing the score helps the policy.

Together, the training and inference comparisons answer RQ2: evidence-based trajectories teach a useful search policy, and numerical pool feedback helps that policy make decisions at deployment.

3.4 RQ3: What retrieval conditions does ADS describe?

RQ2 tests whether feedback helps the deployed policy. RQ3 instead checks what the score describes when annotated evidence is available for analysis. The two components locate lexical gaps and strong competing turns. We examine their distribution across benchmarks and whether high-scoring questions tend to challenge existing search methods. These descriptive checks do not establish that ADS is an optimal predictor of answer errors.



(a) Cross-benchmark difficulty plane

(b) Low- versus high-ADS questions

Figure 3: Offline ADS analyses using annotated evidence. (a) Query-to-evidence coverage and competition across seven benchmarks. (b) For each benchmark, questions are ranked by ADS and the lowest and highest quartiles are compared. Open and filled circles show their respective mean accuracy across five search baselines, joined by a colored segment. Thin horizontal lines show question-level bootstrap 95% confidence intervals for each mean. The number at right is the high-minus-low accuracy difference in percentage points.

Lexical and semantic retrieval gaps. Figure 3(a) places each question by two aspects of its gap from annotated evidence. The vertical axis measures the IDF-weighted share of query terms missing from the evidence. The horizontal axis rises when other history turns compete more strongly with the evidence in embedding similarity. PersonaMem-v2 and PrefEval have larger average gaps on both axes than the factual reference benchmarks (Yang et al., 2018, Ho et al., 2020, Trivedi et al., 2022).

Accuracy across ADS quartiles. Figure 3(b) compares the lowest and highest ADS quartiles separately within each dialogue benchmark. Across five search baselines, accuracy in the high-ADS group is lower by 13.4 points on LongMemEval, 9.6 on LoCoMo, and 3.1 on PersonaMem-v2. Forming quartiles within each benchmark compares relatively direct and indirect questions without assuming a common ADS cutoff across datasets. The largest separation occurs on LongMemEval, showing that this retrieval difficulty also matters for factual memory questions. The smaller PersonaMem-v2 difference shows that the association varies by task. These baseline averages characterize question groups rather than ADAR's own accuracy gain.

Embedding robustness and online feedback. Using a different embedder preserves the benchmark-level ADS ordering on six shared datasets, and the per-question scores remain highly correlated (Appendix B). The same appendix compares ADS with cosine distance, dense rank, and BM25 rank, which sometimes correlate more strongly with baseline errors. Those comparisons assess error association, while ADAR uses ADS to summarize the state of an ongoing search. Figure 3 uses annotated evidence E^* to describe query-to-evidence gaps offline. During search, the student instead receives ADS for its retrieved pool P_r, without access to evidence labels. It reads that score alongside the retrieved turns when deciding whether to search again or answer. RQ2 tests the value of this online feedback through the same-checkpoint removal comparison.

4 Related work and scope

Dialogue-memory benchmarks span factual recall (Maharana et al., 2024, Wu et al., 2025), preference following (Zhao et al., 2025), and implicit persona inference (Jiang et al., 2025a,b). ADAR connects preference answering with dialogue retrieval by learning how to search for supporting turns.

Memory architectures such as MemGPT, MemoryBank, A-MEM, Mem0, and LightMem structure, consolidate, or maintain information over time (Packer et al., 2024, Zhong et al., 2024, Xu et al., 2025, Chhikara et al., 2025, Fang et al., 2025). ADAR instead learns query-conditioned search over a turn index. Structured memory and a learned search policy address complementary parts of this pipeline: how history is represented and how it is explored for a particular request. This also clarifies the two types of comparison in Table 1. Memory-system rows compare complete approaches to using dialogue history, including their storage choices. Shared-backbone search-agent rows hold the model and retriever fixed, giving a more focused comparison of how the retrieval policy is trained and used.

Query rewriting and active retrieval already address gaps between questions and evidence (Gao et al., 2023, Trivedi et al., 2023, Asai et al., 2024, Jeong et al., 2024, Li et al., 2025). Outcome-trained agents further learn search behavior (Jin et al., 2025, Zheng et al., 2025, Song et al., 2025). ADAR differs in where supervision enters. Source evidence guides the teacher's exploration and filters completed paths, so the student sees both a successful answer and the sequence of searches that reached supporting turns. This distinction matters when a plausible final answer can be produced without retrieving the relevant turn. The learned actions remain executable through the same retrieval interface available at deployment.

Learning from a privileged teacher. The information asymmetry connects ADAR to learning using privileged information (Vapnik & Izmailov, 2015): richer training inputs help construct a predictor with restricted deployment inputs. It shares the teacher-student motivation of knowledge distillation (Hinton et al., 2015), transferring filtered search actions and observations rather than soft class distributions. ADAR applies these ideas to indirect dialogue retrieval, requiring demonstrated searches to reach annotated turns and excluding explicit oracle fields from the student view. The evidence annotations guide the teacher toward useful search paths, while the student is trained on the resulting queries, observations, and decisions. This connects privileged supervision to the sequential behavior required for dialogue retrieval.

Scope of transfer. LongMemEval supplies no training trajectories and asks factual memory questions beyond the main source task's implicit-preference format. Its multi-session and temporal gains test whether the learned search behavior extends beyond that question format. The dialogue-turn index, embedder, and student capacity remain fixed, so the evidence concerns task transfer through a common retrieval interface.

Limitations. ADAR uses annotated source evidence and answers to construct training trajectories offline, while the deployed student needs only unannotated history. Extending trajectory construction to weakly supervised or unannotated logs is a natural next step. The experiments use one student-teacher configuration and a turn-level retrieval interface, leaving other model scales and memory representations for future study. Our evaluation targets retrieval-oriented questions rather than privacy, forgetting, or abstention behavior. Primary open-form accuracy uses a fixed model judge, and human assessment of response usefulness would complement this metric.

5 Conclusion

ADAR turns annotated dialogue evidence into search demonstrations for a compact student deployed without evidence labels. Filtered demonstrations supervise query formulation, evidence assessment, and commitment through the same interface used at deployment. Gains span all five evaluated preference categories and extend to held-out multi-session and temporal questions. ADS supplies interpretable pool feedback, and withholding it while preserving autonomous search decisions reduces accuracy. These results support teaching the path to supporting evidence and providing interpretable feedback along that path, while assessing answer support from the retrieved content.

ETHICS STATEMENT

Inferring preferences from dialogue can expose sensitive attributes or treat a tentative inference as a stable user trait. ADAR is evaluated as a retrieval mechanism. It does not implement consent, access control, deletion, or safe personalization. Deployment should respect explicit user instructions and avoid treating retrieved behavioral cues as authoritative diagnoses or immutable preferences. The excluded privacy and forgetting categories are outside the reported evaluation, not requirements that can be ignored. Source dialogue data and derived artifacts must be handled according to their licenses and access conditions.

REPRODUCIBILITY STATEMENT

The appendix specifies benchmark subsets, training hyperparameters, search budgets, per-category results, and synthesis, inference, and judging prompts. It also reports ADS diagnostics, trajectory filtering, and an alternative-judge comparison. The main comparison uses the selected retrieval subsets and the evaluation protocol described in Section 3.1.

AI USE STATEMENT

Generative language models produced candidate search trajectories for training and judged open-form answers, as detailed in the method and appendix. An AI assistant aided code inspection and small implementation changes, analysis and figure preparation, and drafting and polishing the manuscript. The authors retain responsibility for the final text, code, numerical results, and claims.

REFERENCES

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building production-ready AI agents with scalable long-term memory. *arXiv preprint arXiv:2504.19413*, 2025.

Jizhan Fang, Xinle Deng, Haoming Xu, Ziyan Jiang, Yuqi Tang, Ziwen Xu, Shumin Deng, Yunzhi Yao, Mengru Wang, Shuofei Qiao, Huajun Chen, and Ningyu Zhang. LightMem: Lightweight and efficient memory-augmented generation. *arXiv preprint arXiv:2510.18866*, 2025.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. Precise zero-shot dense retrieval without relevance labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1762–1777. Association for Computational Linguistics, 2023.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics (COLING)*, pp. 6609–6625. International Committee on Computational Linguistics, 2020.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spaCy: Industrial-strength Natural Language Processing in Python, 2020.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong C. Park. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 7036–7050. Association for Computational Linguistics, 2024.

Bowen Jiang, Zhuoqun Hao, Young-Min Cho, Bryan Li, Yuan Yuan, Sihao Chen, Lyle Ungar, Camillo J. Taylor, and Dan Roth. Know me, respond to me: Benchmarking LLMs for dynamic user profiling and personalized responses at scale. In *Conference on Language Modeling (COLM)*, 2025a.

Bowen Jiang, Yuan Yuan, Maohao Shen, Zhuoqun Hao, Zhangchen Xu, Zichen Chen, Ziyi Liu, Anvesh Rao Vijjini, Jiashu He, Hanchao Yu, Radha Poovendran, Gregory Wornell, Lyle Ungar, Dan Roth, Sihao Chen, and Camillo Jose Taylor. PersonaMem-v2: Towards personalized intelligence via learning implicit user personas and agentic memory. *arXiv preprint arXiv:2512.06688*, 2025b.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-R1: Training LLMs to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. Search-o1: Agentic search-enhanced large reasoning models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5420–5438. Association for Computational Linguistics, 2025.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. Evaluating very long-term conversational memory of LLM agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13851–13870. Association for Computational Linguistics, 2024.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. MemGPT: Towards LLMs as operating systems. In *Conference on Language Modeling (COLM)*, 2024.

Qwen Team. Qwen3.5. https://huggingface.co/collections/Qwen/qwen35, 2026.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. R1-Searcher: Incentivizing the search capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2503.05592*, 2025.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10014–10037. Association for Computational Linguistics, 2023.

Vladimir Vapnik and Rauf Izmailov. Learning using privileged information: Similarity control and knowledge transfer. *Journal of Machine Learning Research*, 16(61):2023–2049, 2015. URL https://www.jmlr.org/papers/v16/vapnik15b.html.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. LongMemEval: Benchmarking chat assistants on long-term interactive memory. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-MEM: Agentic memory for LLM agents. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2369–2380. Association for Computational Linguistics, 2018.

Kai Zhang, Xinyuan Zhang, Ejaz Ahmed, Hongda Jiang, Caleb Kumar, Kai Sun, Zhaojiang Lin, Sanat Sharma, Shereen Oraby, Aaron Colak, Ahmed A. Aly, Anuj Kumar, Xiaozhong Liu, and Xin Luna Dong. AssoMem: Scalable memory QA with multi-signal associative retrieval. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=ZCjWUBwCwE.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 Embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025.

Siyan Zhao, Mingyi Hong, Yang Liu, Devamanyu Hazarika, and Kaixiang Lin. PrefEval: Do LLMs recognize your preferences? evaluating personalized preference following in LLMs. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025.

Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei Liu. DeepResearcher: Scaling deep research via reinforcement learning in real-world environments. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 414–431. Association for Computational Linguistics, 2025.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. MemoryBank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 19724–19731, 2024.

A ADS DOMAIN, MONOTONICITY, AND INTERPRETATION

Domain of the reported expression. The lexical denominator must be positive and the sets E and $H \setminus E$ must be nonempty. For unrestricted cosine, each similarity lies in $[-1, 1]$, their difference in $[-2, 2]$, and the raw semantic term in $[-1, 3]$. Thus $\sqrt{\delta_\ell \delta_\sigma}$ is real-valued only when its argument is nonnegative. On the nonnegative semantic domain its maximum is $\sqrt{3}$, not universally $\sqrt{2}$. If all relevant cosine values instead lie in $[0, 1]$, the semantic term lies in $[0, 2]$ and the latter bound follows. Unit normalization alone does not ensure that condition.

Boundary cases. For an empty pool, ADS returns $\sqrt{2}$. It sets $\delta_\ell = 0$ for an empty query-anchor set and $\delta_\ell = 1$ when its IDF weights sum to zero, uses $s_{\text{out}} = 0$ for an empty complement, and clips a negative semantic term to zero. These conventions keep the score real-valued in $[0, \sqrt{3}]$. The monotonicity result below concerns nonempty proper pools, where both maxima in the formula are defined.

Monotonicity on the stated domain. Let $E \subseteq F \subseteq H$, holding q, the history, anchors, IDF weights, and embeddings fixed. Since $A_E \subseteq A_F$, the missing-anchor numerator cannot increase, so $\delta_\ell(q, F) \leq \delta_\ell(q, E)$. A maximum over the expanding evidence set cannot decrease, whereas a maximum over its shrinking complement cannot increase. Hence $\delta_\sigma(q, F) \leq \delta_\sigma(q, E)$. If both semantic terms are nonnegative, multiplying the two nonnegative inequalities and taking square roots proves $\text{ADS}(q, F) \leq \text{ADS}(q, E)$. The proof concerns each individual growing pool, not averages over a changing population of active trajectories.

Why saturation does not certify evidence. Suppose a query asks for exercise recommendations and a supporting memory reports the user's cardiovascular diagnosis. A distractor that literally lists "exercise recommendations" can cover all query anchors while offering only generic advice. Once this distractor is retrieved, $\delta_\ell = 0$ and ADS is zero wherever the semantic term is defined, even if the medical-history turn remains outside the pool. This schematic counterexample shows why the agent must inspect content. Likewise, a pool containing some annotated evidence can omit other necessary turns, or an answer can ignore retrieved evidence. Neither a low score nor the evidence-presence filter establishes full grounding.

Relation to standard retrieval signals. ADS combines familiar lexical and embedding ingredients over a reference set of turns. Anchors are sets, so repetition adds no weight, and the IDF-weighted lexical term measures normalized missing coverage. The semantic term compares the best in-set match with the strongest out-of-set competitor. At a tie, $\delta_\sigma = 1$. Absolute cosine reports similarity without this competition, whereas gold-evidence rank needs evidence labels. The reference-set formulation permits both annotated offline analysis and unlabeled pool feedback. Its interpretation depends on the extractor, IDF corpus, embedder, and evidence convention. The geometric mean is an interpretable aggregation choice rather than an experimentally established optimum. ADS is neither a calibrated error probability nor a certificate of answer support.

Computation. History anchors and embeddings can be cached. Exact s_{out} refers to unretained history turns, so feedback is not a pool-only computation. For a fixed original query, computing similarities to all n turns takes $O(nd)$ work for embedding dimension d. After caching these similarities, a sorted list and pool membership can support updates without recomputing every cosine each round. Search queries change with the policy and still require retrieval. Offline gold-based analysis is outside the deployment inference path. The reported construction and training/evaluation times are in Appendix G.

B ADDITIONAL ADS DIAGNOSTICS

We assess whether the offline score changes with the embedding model and compare its association with baseline errors against simpler retrieval measures. These diagnostics use the case counts shown below. Agreement in benchmark ordering does not imply identical per-query scores.

Table 3: Embedding sensitivity: Spearman correlation of per-query full ADS between Qwen3-Embedding-0.6B and BAAI/bge-small-en-v1.5. The six-centroid ordering has correlation 1.000.

Benchmark	Cases	Correlation
HotpotQA	20	0.997
2WikiMultiHopQA	20	0.997
MuSiQue	20	0.982
LongMemEval	100	0.998
PersonaMem-v2	199	0.990
PrefEval implicit-choice	20	0.980

Table 4: Spearman correlation with mean error across five baselines in the supplementary diagnostic. Higher values indicate stronger error association. ADS is not consistently the strongest predictor.

Dataset / embedder	ADS	Lexical	Semantic	Cosine dist.	Dense rank	BM25 rank
PersonaMem / Qwen	.043	.045	.132	.241	.293	.044
LongMemEval / Qwen	.166	.150	.291	.341	.264	.217
PersonaMem / BGE	.046	.045	.026	.038	.090	.044
LongMemEval / BGE	.157	.150	.257	.220	.067	.217
LoCoMo / BGE	.159	.155	.287	.392	.389	.389

Scope of component diagnostics. The lexical-only and semantic-only rows in Table 2 describe performance under component feedback. The controlled inference comparison uses the same trained checkpoint with all numerical ADS feedback removed while preserving autonomous commit/pivot decisions and the retrieval cap. It measures whether the policy benefits from feedback, without isolating individual query and stopping decisions.

C Online score plots

Figure 4 shows ADS over search rounds and answer accuracy grouped by commit-time ADS. These plots describe search behavior and its association with accuracy. The same-checkpoint feedback comparison in Table 2 separately tests the effect of providing numerical feedback.

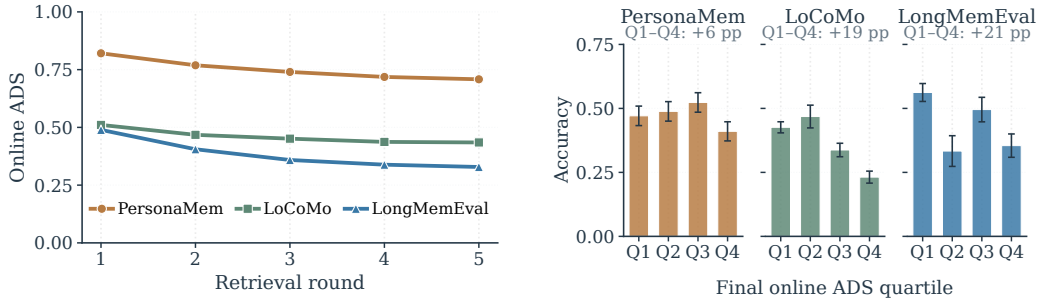


Figure 4: ADS across search rounds (left) and accuracy grouped by final ADS (right). Pool growth lowers the score where the formula is defined, while the relation to answer accuracy is measured empirically.

D Supplementary judge results

Table 5 reports how the ADAR and DeepResearcher scores change under two alternative judges. The Original column rounds the corresponding means in Table 1 to one decimal place. This comparison checks the ordering of these two methods under each judge.

Table 5: Accuracy (%) under alternative judges. Original rounds the corresponding Table 1 mean to one decimal place.

Dataset	Method	Original	GLM5	DeepseekV4
LongMemEval	ADAR	48.4	46.9	45.2
	DeepResearcher	19.4	18.0	19.6
LoCoMo	ADAR	35.9	37.4	34.5
	DeepResearcher	33.3	35.2	32.8

E CONSTRUCTION-STAGE SELECTION

Table 6: Sampled sequential filtering funnel. Each rejection count is additional at that stage, rather than an independently measured filter failure rate. These 300 candidates are a diagnostic sample, not the full synthesis corpus.

Source	Candidates	Anti-leak	Outcome	Evidence	Surviving
PersonaMem-v2	240	98	32	27	83
LoCoMo	60	23	25	1	11
Total	300	121	57	28	94

The sequential funnel characterizes selection in this 300-candidate sample. Each row records the additional candidates removed at each filter stage, so the counts do not measure the independent effect of a filter on answer accuracy. The anti-leak checks and removal of privileged input fields reduce direct leakage channels, while subtle teacher-induced shortcuts remain possible.

F BENCHMARK DETAILS

We evaluate on three benchmarks chosen as representative points of the explicit-to-implicit spectrum.

PersonaMem-v2 is a four-choice MCQ benchmark for implicit preference inference in task-grounded dialogue. Its queries span seven preference types, and we evaluate the memory-recall slice of five behavioural types (neutral, anti-stereotypical, therapy-background, health-and-medical, stereotypical) and drop two types that do not test memory recall. `sensitive_info` (about 10% of items) places a PII string in the query itself and asks the model to redact it, so correctness requires masking rather than retrieval. `ask_to_forget` (about 21%) asks the model to honour a prior "forget X" instruction, so correctness requires suppressing a stated fact, again not retrieval. The five retained types form the abductive behavioural-retrieval core.

LongMemEval tests long-horizon factual recall over multi-session chat. We evaluate on the 470 memory-recall questions, excluding the 30 knowledge-abstention questions whose correct answer is to decline to answer, since these test calibration rather than memory retrieval. We report four families: single-session, multi-session, temporal, and knowledge-update. LongMemEval is used only for evaluation, never for training.

LoCoMo is an open-form question-answering benchmark over long-term, multi-session conversations. We follow the conversation-level split of Mem0 (Chhikara et al., 2025), with disjoint conversations for trace synthesis, validation, and evaluation and zero overlap between synthesis and test. We report single-hop, multi-hop, open-domain, and temporal questions.

G TRAINING AND INFERENCE DETAILS

Student training. We full-fine-tune the Qwen3.5-4B base model (Qwen Team, 2026) on filtered teacher demonstrations with effective batch size eight, learning rate 1×10^{-5} under a cosine schedule, two epochs, and a $12{,}288$-token context. The corpus contains $4{,}960$ trajectories, with $4{,}666$ from PersonaMem-v2 and 294 from the LoCoMo training conversation. Multiple trajectories can originate

from one source instance. Loss is causal-LM cross-entropy with role masking, so gradients flow only on assistant tokens. The two RL baselines start from the same initialisation, also update all parameters, and use the same training data (Appendix H).

Answer judging. The committed `<answer>` is scored by an MCQ letter judge for PersonaMem-v2 or a frozen Qwen3.5-122B YES/NO judge for LongMemEval and LoCoMo. The judge sees only $(q, \text{options}, \text{candidate})$ or $(q, \text{gold}, \text{candidate})$, never the search transcript. Prompts appear in Appendix J.

Compute and infrastructure. Offline trajectory generation and filtering required 48 hours on two NVIDIA A100 GPUs. ADAR training and evaluation together required 72 hours on two NVIDIA A100 GPUs. These times cover ADAR's data construction and training/evaluation stages, excluding baseline development and ablations.

The offline construction cost produces a reusable corpus for student SFT. Deployment uses the 4B student and retrieval feedback without the 27B synthesis teacher. These offline costs are separate from per-query deployment costs.

H Baseline Configurations

The five search-agent baselines use the Qwen3.5-4B backbone and the Qwen3-Embedding-0.6B retriever. Each benchmark uses the same answer-scoring protocol for these baselines and ADAR.

Training-free. **Native RAG** retrieves the top-10 turns by query cosine and answers in a single call (300 max new tokens). **IRCoT** (Trivedi et al., 2023) interleaves chain-of-thought with retrieval for up to five iterations, retrieving the top-3 turns per step and terminating when a reasoning step emits "so the answer is". **Search-o1** (Li et al., 2025) runs an agentic search-and-reason loop for up to five search rounds, top-5 turns per search, with an in-documents reasoning step that condenses retrieved turns before answering.

Outcome-RL (retrained). **Search-R1** (Jin et al., 2025) and **DeepResearcher** (Zheng et al., 2025) are retrained from the Qwen3.5-4B base with full-parameter GRPO on the same trace corpus as our method, each with its own outcome reward. Both use identical training hyperparameters in the reported configuration: 500 GRPO steps, learning rate 1×10^{-6}, KL coefficient $\beta = 0.001$, and 4 rollouts per step. At inference, both retrieve up to 5 rounds with top-5 results per round. Both are trained with full parameters to match ADAR's training regime.

I Per-Category Results

The three tables below report per-category accuracy.

Table 7: PersonaMem-v2 per-category MCQ accuracy (%). Categories: Neutral, Anti-stereotypical, Therapy, Health, and Stereotypical.

Method	Neutral	A-stereo	Therapy	Health	Stereo	**Over.**
Native RAG	27.91 ±1.01	22.22 ±1.75	32.11 ±4.96	23.68 ±8.36	45.79 ±4.24	29.32 ±1.38
IRCoT	28.49 ±4.64	32.16 ±8.65	23.31 ±5.99	39.47 ±4.27	42.06 ±5.78	32.37 ±3.51
Search-o1	33.14 ±1.01	20.47 ±4.42	19.77 ±2.75	35.09 ±5.26	25.23 ±7.35	26.71 ±2.05
Search-R1	14.57 ±3.65	9.15 ±3.65	29.49 ±1.37	14.14 ±3.04	18.48 ±3.21	16.64 ±0.66
DeepResearcher	25.00 ±1.01	22.22 ±1.19	20.23 ±4.12	21.05 ±1.52	17.76 ±0.32	21.63 ±0.68
AssoMem	16.67 ±6.69	25.81 ±7.91	45.45 ±10.59	25.00 ±11.00	25.00 ±8.29	26.77 ±3.77
LightMem	33.33 ±8.57	32.26 ±8.58	18.18 ±8.19	12.50 ±8.16	50.00 ±9.42	31.50 ±3.99
ADAR	45.35 ±4.64	45.03 ±4.42	39.31 ±6.30	56.14 ±6.62	54.21 ±5.78	47.31 ±2.42

Table 8: LoCoMo per-category open-form accuracy (%). Categories: Single-hop, Multi-hop, Open-domain, and Temporal.

Method	S-hop	M-hop	O-dom	Temp	**Over.**
Native RAG	20.59 ±2.59	0.78 ±0.33	14.46 ±8.33	45.25 ±2.39	29.91 ±0.88
IRCoT	23.11 ±3.76	2.33 ±2.09	25.30 ±8.33	43.85 ±1.31	30.60 ±1.50
Search-o1	21.01 ±4.80	4.28 ±0.70	31.33 ±8.00	37.85 ±2.87	27.67 ±0.94
Search-R1	13.03 ±1.72	3.50 ±0.70	26.51 ±8.33	32.82 ±1.08	22.95 ±0.42
DeepResearcher	26.47 ±5.38	5.84 ±1.39	27.71 ±2.31	46.09 ±1.31	33.31 ±1.26
AssoMem	25.15 ±7.37	8.33 ±5.61	25.00 ±15.35	47.95 ±4.73	34.51 ±3.45
LightMem	12.20 ±5.08	4.17 ±4.11	12.50 ±11.73	23.81 ±4.11	23.48 ±2.80
ADAR	31.93 ±1.49	9.34 ±1.20	25.30 ±6.93	48.04 ±0.30	35.94 ±0.57

Table 9: LongMemEval per-category open-form accuracy (%). The single-session subcategories are merged into Single-Sess (S-Sess).

Method	S-Sess	M-Sess	Temp	K-Upd	**Over.**
Native RAG	58.89 ±3.46	37.37 ±1.44	20.17 ±5.99	62.86 ±4.81	44.02 ±1.48
IRCoT	66.00 ±1.15	15.15 ±1.44	17.65 ±4.96	44.29 ±8.33	37.90 ±2.59
Search-o1	49.33 ±1.15	13.13 ±0.12	12.04 ±1.37	28.57 ±6.36	27.63 ±0.64
Search-R1	22.67 ±3.06	11.11 ±1.44	11.76 ±0.17	8.57 ±6.36	14.74 ±2.25
DeepResearcher	22.44 ±10.39	11.11 ±2.50	15.13 ±0.21	32.86 ±4.81	19.44 ±3.23
AssoMem	50.00 ±9.17	41.41 ±10.37	21.85 ±8.58	61.43 ±13.38	41.43 ±5.07
LightMem	54.00 ±8.93	45.45 ±10.39	39.50 ±10.19	60.00 ±12.93	46.27 ±5.20
ADAR	58.67 ±4.16	43.43 ±2.89	33.61 ±4.96	58.57 ±7.22	48.40 ±3.57

J PROMPTS

The teacher synthesis system prompt is shown in Box 1. The student inference prompt uses the same protocol with the anti-leak block removed (Box 3). The `<pool_metrics>` feedback reports the current online ADS of the cumulative pool.

Box 1: Teacher system prompt (restricted-oracle synthesis)

```
You are demonstrating a personalized-memory search agent. The agent searches a user's chat history to
      answer a query.
PROTOCOL: strict two-phase round structure:
Phase A (when asked for THOUGHT/QUERY): output EXACTLY two lines:
THOUGHT: <one sentence on which angle to try next, grounded in what you can observe>
QUERY: <a single short search string>
Phase B (after the system returns observation + pool_metrics): output the judgment block, using ONLY
      these tags:
<judgment> 1-3 sentences over the online-ADS reading + content evidence </judgment>
<decision>commit|pivot</decision>
<answer> open-form answer, grounded in retrieved turns </answer>  (commit only)
pool_metrics reports the current online ADS of the cumulative pool: a lower value means the pool now
      covers the query well (retrieval has saturated). A higher value means relevant evidence is still
      missing.
You have up to 5 rounds. If you reach round 5 without committing, commit on round 5 with the strongest
      available evidence.
==== ANTI-LEAK RULES (load-bearing) ====
You may be given background context about the user's preference. Your job is to demonstrate how an
      agent (which has NO such context at inference time) would naturally reach the supporting evidence.

RULE 1  Search queries must be plausible given ONLY the user's query. Use synonyms, related concepts,
      natural pivots. Do NOT use keywords from the background unless they would naturally arise from
      the query.
```

```
  BAD : query "easy ways to cut down on sugar". Background mentions "Hebrew festival recipes". QUERY: "
      Hebrew festival low sugar".
  GOOD: QUERY: "low-sugar baking habits".
RULE 2  Forbidden words/phrases (case-insensitive). Any occurrence discards the trace: "gold", "oracle
      ", "hint", "told", "provided", "given", "target answer", "ground truth", "the context says", "
      target turn", "answer-bearing turn", and any phrasing implying out-of-band knowledge of the
      preference.
RULE 3  THOUGHT must read as an in-the-moment hypothesis, not knowledge.
  BAD : "The target turn likely contains 'visiting researchers'..."
  GOOD: "Earlier searches found general topics. Let me try the user's work context."
RULE 4  Cite retrieved turns by their turn# (seen in <observation>). Never reference a turn not in the
      current pool.
RULE 5  The QUERY must not contain a multi-word phrase that appears neither in the user's query nor in
      what you have already retrieved.
If tempted to use background info to shortcut the search, DON'T. Run the search as an authentic agent
      would, and let the retrieved evidence drive your judgment.
```

Box 2: Teacher initial user message (BACKGROUND block)

```
User query: {query}
BACKGROUND (only for your synthesis judgment, NEVER paraphrase, quote, or reference in output):
- general preference being tested: {preference}
- a turn supporting this preference exists in the chat history (you must find it through natural search
      and must NOT copy phrases from the next line)
  excerpt-for-judgment-only: '{gold_snippet_head}'
Use the BACKGROUND ONLY to verify when you have found supporting evidence. Never use it to construct
      your queries or thoughts.
Phase A: output THOUGHT and QUERY for round 1.
```

Box 3: Student inference prompt (system + round messages)

```
[system]
You are a personalized-memory search agent. You search a user's chat history to answer a query.
PROTOCOL: strict two-phase round structure:
Phase A: output EXACTLY two lines:
THOUGHT: <one sentence on which angle to try next>
QUERY: <a single short search string>
Phase B (after observation + pool_metrics): output, using ONLY:
<judgment> 1-3 sentences over the online-ADS reading + evidence </judgment>
<decision>commit|pivot</decision>
<answer> open-form answer grounded in retrieved turns </answer> (commit only)
pool_metrics reports the current online ADS of the cumulative pool (lower = pool covers the query well,
      higher = evidence still missing).
You have up to 5 rounds. At round 5 commit with the best evidence.
[user, round 1]
User query: {query}
Output THOUGHT and QUERY for round 1.
[user, after each search]
<observation>
[turn#{idx} sim={cos}] {turn text} ...
</observation>
<pool_metrics>online_ADS={value}</pool_metrics>
Judge the pool and decide commit/pivot.
[user, on pivot]
Output THOUGHT and QUERY for the next round.
```

Box 4: MCQ letter judge (PersonaMem-v2)

```
[system]
You answer a multiple-choice question based on a candidate answer. Pick the option (a/b/c/d) that best
      matches the candidate answer. Reply with exactly one letter.
[user]
Question: {query}
Options:
(a) {option a}
(b) {option b}
(c) {option c}
(d) {option d}
Candidate answer: {answer_text}
Which option (a/b/c/d) best matches the candidate answer? Reply with one letter only.
```

Box 5: YES/NO confirm-reveal judge (LongMemEval, LoCoMo)

```
You are an evaluation judge for a personalized-response benchmark.
Given the user's query, the hidden ground-truth user preference, and the model's response, decide if
      the response REFLECTS / HONORS the preference. The response need not mention the preference
      verbatim. It just needs to be consistent with what someone holding that preference would receive
      as a helpful answer.
Output exactly one token: YES or NO.
User's query: {user_query}
Hidden ground-truth user preference: {gt_preference}
Model's response: {model_response}
Decision (YES or NO):
```