ATOD: An Evaluation Framework and Benchmark for Agentic Task-Oriented Dialogue Systems

Anonymous ACL submission

Abstract

LLM-based task-oriented dialogue (TOD) agents must coordinate interleaved goals, maintain long-horizon state, and act proactively under asynchronous execution, but current benchmarks do not jointly assess these capabilities with turn-level lifecycle supervision. We introduce **ATOD**, a benchmark of 1,000 synthetic dialogues annotated for six advanced-TOD properties, and **ATOD-Eval**, whose dependency-aware metrics measure completion, memory recall, and proactivity. We instantiate ATOD-Eval with a symbolic–vector memory evaluator. Against five prompt-only judges and six matched-backbone memory baselines, it is the only configuration above 90% in both goal detection and status tracking on medium dialogues; on complex dialogues, it improves status accuracy over the next-best memory baseline by 3.68 points while recording the lowest per-turn latency. A cross-family judge swap and a manual audit with 93.8% agreement confirm measurement reliability.

1 Introduction

Task-oriented dialogue (TOD) has traditionally been studied as short, sequential interactions that resolve user requests turn by turn (Rastogi et al., 2020; Budzianowski et al., 2018). Agentic LLM assistants such as ChatGPT (OpenAI, 2024) and Gemini (Comanici et al., 2025) have shifted user expectations toward a richer setting we call **Advanced TOD**, where the agent coordinates concurrent goals, suspends and resumes them, respects evolving dependencies, maintains long-horizon memory, handles asynchronous execution, and acts proactively. Table 1 shows such an agentic interaction with interlocked flight, hotel, and restaurant bookings.

Evaluating advanced TOD is more difficult than slot-filling: an agent can complete individual goals yet fail to manage their dependencies, or track status for detected goals while missing others.

Table 1: Illustrative dialogue showing the six advanced-TOD capabilities targeted by ATOD; an annotated ATOD excerpt is provided in Appendix A.1.

User Utterance	Agent Response	Key Capabilities
Book a flight to Chicago for next Monday.	I found several options. Would you like me to confirm one now and also arrange a hotel near your destination?	Multi-goal concurrency
Yes, please. Also remind me to pack my passport the night before.	Flight confirmed and hotel reserved. A reminder has been set for Sunday night to pack your passport.	Long-horizon memory, Multi-goal concurrency
What's the weather in Chicago on Monday?	The forecast for Monday in Chicago is rainy, with a high of 12°C.	Interleaved workflows
Also book a dinner at an Italian restaurant for Monday night after my flight.	Your flight lands at 6:30pm. A table at *Luigi's* is reserved for 8:00pm.	Dependency management, Multi-goal concurrency
Arrange a ride from the airport to the hotel.	Ride scheduled for 6:45pm. Do you want the driver's details now or closer to the date?	Asynchronous execution, Proactivity
Closer to the date, please.	Got it. I will share the driver's info the day before your trip.	Long-horizon memory, Asynchronous follow-up

Existing benchmarks—MultiWOZ (Budzianowski et al., 2018), SGD (Rastogi et al., 2020), and τ-Bench (Yao et al., 2024)—do not provide turn-level lifecycle annotations across this combination of behaviors; TOAD (Liu et al., 2024b) and LUCID (Stacey et al., 2024) target parts of it. Likewise, general LLM judges (Liu et al., 2023; Zheng et al., 2023) and memory benchmarks such as LoCoMo (Maharana et al., 2024) do not evaluate dependency-aware goal trajectories. Thus, no existing benchmark supports fine-grained evaluation of the advanced-TOD capabilities studied here.

We close this gap with three contributions. **(i) ATOD**, a benchmark of 1,000 synthetic dialogues that jointly covers all six advanced-TOD properties at turn-level granularity, together with its full generation pipeline (§4). **(ii) ATOD-Eval**, a multi-dimensional evaluation framework whose dependency-aware metrics (dGCR, MRA, PE) complement traditional success rates with offline and online signals about goal completion, mem-

ory recall, proactivity, and response quality (§6). **(iii)** An **agentic memory evaluator** with a dual symbolic–vector store and a four-stage turn-level pipeline (extract / match / update-or-insert / audit) that achieves the strongest joint detection and tracking on complex dialogues at the lowest latency among matched-backbone baselines (§5, §8).

2 Related Work

2.1 TOD Systems Evaluation

G-Eval (Liu et al., 2023), AlpacaEval (Dubois et al., 2024), and MT-Bench (Zheng et al., 2023) target open-domain dialogue and emphasize fluency over goal- and memory-driven behavior. In TOD, evaluation has evolved from turn-level user satisfaction (Walker et al., 2000) to dialogue-level quality estimation (Deng et al., 2022; Komma et al., 2023), zero-shot LLM judges (Kazi et al., 2024), and simulator-based evaluation (Davidson et al., 2023). Recent benchmarks (τ-Bench (Yao et al., 2024), AutoEval-ToD (Jain et al., 2025), TD-EVAL (Acikgoz et al., 2025)) still focus on inform/success rates rather than advanced-TOD capabilities; their metric schemas (slot-level inform/success or rubric scoring) are not directly comparable to the turn-level lifecycle metrics used here, so we treat them as motivating prior work rather than as direct baselines.

2.2 TOD Datasets and Benchmarks

Human-curated datasets (MultiWOZ (Budzianowski et al., 2018), SGD (Rastogi et al., 2020), RADDLE (Peng et al., 2020), τ-Bench (Yao et al., 2024), MS-TOD (Du et al., 2025)) support dialogue state tracking and task completion but are largely single-session. Synthetic datasets (TOPDIAL (Wang et al., 2023), TOAD (Liu et al., 2024b), LUCID (Stacey et al., 2024), SynthDST (Kulkarni et al., 2024)) add personalization or proactivity but lack joint coverage. In broader agent evaluation, AgentBench (Liu et al., 2024a), T1 (Chakraborty et al., 2026), and AgentChangeBench (Rana et al., 2025) target environments, tool-use planning, and goal-shift adaptation, but none provides turn-level goal lifecycle annotations with dependency-aware metrics.

2.3 Memory for Dialogue Systems

RAG (Lewis et al., 2020), MemoChat (Lu et al., 2023), and MemoryBank (Zhong et al., 2024) support session-level recall; MemGPT (Packer et al., 2023), A-Mem (Xu et al., 2025), Mem0 (Chhikara et al., 2025), and MemOS (Li et al., 2025) introduce structured long-term memory; Chronos (Sen et al., 2026) and SGMem (Wu et al., 2025) use event-based representations; HierTOD (Mo et al., 2024) and TACT (Yoon et al., 2025) target hierarchical and proactive TOD. Benchmarks like LoCoMo (Maharana et al., 2024) and THEANINE (Ong et al., 2024) study memory temporality but not dependency-aware goal evaluation, which ATOD complements.

3 Problem Formulation

3.1 Characteristics of Advanced TOD

Prior TOD benchmarks (Budzianowski et al., 2018; Rastogi et al., 2020; Yao et al., 2024) typically isolate one or two of the behaviors below, while realistic agentic interactions exhibit them jointly. We therefore characterize advanced TOD by six properties that any evaluation must support together: *multi-goal concurrency*, where users pursue multiple objectives in parallel; *interleaving*, where goals are suspended and resumed rather than confined to contiguous turn spans; *dependency management*, where goals carry prerequisite or blocking relations (e.g., PAYMENT depends on BOOKING); *long-horizon memory*, where state is preserved across many turns and goal interruptions; *asynchronous execution*, where delayed goals remain PENDING until external conditions hold, including system-initiated follow-ups; and *proactivity*, where the system offers reminders or suggestions without explicit prompting.

3.2 Task Formulation

We formalize advanced TOD as a corpus $\mathcal{D} = \{(\mathcal{G}_i, \mathcal{C}_i)\}_{i=1}^{N}$, where $\mathcal{C}_i = \{c_{i,t}\}_{t=1}^{T_i}$ is the turn sequence of dialogue i and $\mathcal{G}_i$ is its goal set with explicit inter-goal dependencies. In traditional TOD a goal occupies a contiguous span of turns, whereas an advanced-TOD goal $g \in \mathcal{G}_i$ may span disjoint intervals of $\mathcal{C}_i$ as it is initiated, suspended, and resumed. We therefore represent each goal by its turn-aligned *status trajectory* $\{\mathrm{Status}(g, t)\}_{t=1}^{T_i}$ (e.g., OPEN→PENDING→COMPLETED), capturing a non-contiguous lifecycle that prior single-span representations cannot. The evaluation objective is to measure how well a system maintains and updates these trajectories in both *offline* (over completed dialogues) and *online* (turn by turn) settings.

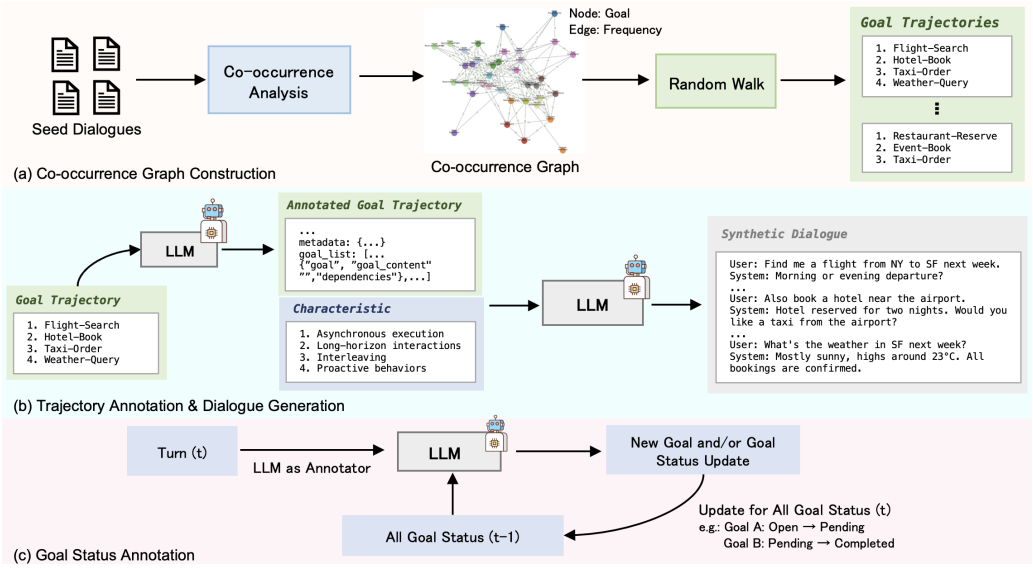

Figure 1: ATOD dataset curation pipeline. (a) **Co-occurrence Graph & Trajectory Sampling**: build a goal co-occurrence graph from seed dialogues and sample diverse multi-goal trajectories via random walks; (b) **Trajectory Annotation & Dialogue Generation**: annotate slot values, dependencies, and complexity, then generate dialogues conditioned on trajectories; (c) **Goal Status Annotation**: label active goals and lifecycle states at each turn for fine-grained tracking of dialogue progress.

4 ATOD: A Synthetic Dialogue Dataset and Generation Pipeline

We present **ATOD**, a synthetic dataset and generation pipeline for benchmarking the advanced-TOD characteristics in §3.1. ATOD contains 1,000 richly annotated multi-turn dialogues (428 medium, 572 complex) spanning 16 domains and 36 intents, with an average of 26.6 turns and 6.5 goals per dialogue. Synthetic construction does not make the benchmark trivial: the generator has access to the global trajectory, but the evaluated system must recover goal lifecycles from dialogue evidence alone. As shown in Figure 1, construction includes staged quality assurance: a verifier-judge pass separate from generation filters or regenerates inconsistent instances, and an author manually screens generated dialogues and annotations for evident defects. Appendix A.6 documents these checks and Appendix A.2 reports automated ratings. ATOD will be released as a zero-shot, test-only benchmark upon publication.

4.1 Co-occurrence Graph Construction and Goal Trajectory Sampling

Independent goal sampling yields implausible combinations and fixed templates limit diversity. We therefore construct a goal co-occurrence graph $G = (V, E)$ from a seed dialogue dataset, where nodes are goals and weighted edges capture empirical co-occurrence frequency. Candidate goal sets $S = \{g_1, \ldots, g_k\}$ are sampled by stratified random walks of varying lengths over G (Figure 1(a)), preserving realistic correlations while introducing diversity. The procedure is dataset-agnostic; we instantiate it on Schema-Guided Dialogue (SGD) (Rastogi et al., 2020), yielding trajectories with rich domain variation and naturally multi-goal, interleaved, long-horizon structure.

4.2 Annotation of Goal Trajectories and Complexity Categorization

Each sampled set S is instantiated into a concrete trajectory via LLM annotation (Figure 1(b)), yielding slot values, inter-goal dependencies D_S (e.g., prerequisites), and natural-language descriptions. Trajectories are tagged with a complexity label $c(S)$ derived from quantitative attributes (goal count, dependency density) and qualitative factors (interleaving, proactivity opportunities); see Appendix A.4. Quality control combines duplicate filtering, retries, and a dedicated verifier judgment of trajectory validity and generated-dialogue consistency (Appendix A.6).

4.3 Dialogue Generation

Dialogue synthesis is conditioned on the annotated trajectory $\tau = (S, D_S, c(S))$. Goals, dependencies, and complexity are combined into structured prompts (Appendix A.7) that produce a multi-

turn conversation $\mathcal{C} = \{c_t\}_{t=1}^{T}$ with interleaving, asynchrony, proactive assistance, and dependency-aware coordination.

4.4 Turn-level Goal Status Annotation

Finally, an LLM annotator labels each goal at every turn (Figure 1(c)) with an active goal set $\mathcal{A}_t \subseteq S$ and statuses $\mathrm{Status}(g, t) \in \{$NOT_MENTIONED, OPEN, PENDING, COMPLETED, FAILED, ABANDONED$\}$. The six-state schema captures the non-contiguous lifecycles that single-span representations miss (§3), yielding a turn-aligned `status_history` for benchmarking interleaved, asynchronous, and multi-goal trajectories.

4.5 Dataset Coverage

ATOD spans diverse domains and goal complexities, from two-goal cases to interdependent long-horizon workflows. Table 2 highlights the operational dimensions for which comparable annotations or behaviors are identifiable in published benchmark descriptions; ATOD combines these with concurrency and long-horizon goal trajectories. We use the full collection as a held-out benchmark and will release the dataset, generation pipeline, evaluation harness, and prompt templates upon publication. Table 3 reports the evaluation split.

5 Agentic Memory System

We instantiate ATOD-Eval with an *agentic memory system* (Figure 2) that maintains and updates goal trajectories from dialogue text alone. A naive LLM-only approach re-reads the entire history at each turn, while flat memory loses goal structure. We instead combine *symbolic precision* (lifecycle state and dependency links) with *semantic flexibility* (retrieval under lexical variation) through a dual store (§5.1) and a turn-level pipeline (§5.2).

5.1 Dual Memory Store

As shown in Figure 2(b), the memory system maintains (i) a *structured goal database* $\mathcal{D}_{\mathrm{sym}}$ for symbolic state and dependency metadata, and (ii) a *semantic vector store* $\mathcal{D}_{\mathrm{vec}}$ (e.g., FAISS (Douze et al., 2024)) for similarity-based retrieval. Each goal stores an identifier, lifecycle state, turn-level status history, textual description, dependency links, and embedding. This dual representation preserves precise symbolic state while still supporting retrieval under lexical variation, which matters in long-horizon dialogues where the same objective is often revisited in different wording.

5.2 Turn-Level Processing Pipeline

At each turn t, given utterance u_t and context c_t, the system extracts candidate goals $\mathcal{G}_t = \mathrm{extract}(u_t, c_t)$. Each candidate $g_t^{(i)}$ is then matched to an existing goal and updated, or inserted as a new goal and evolved in the dependency graph; a final auditing step keeps active states consistent:

$$(u_t, c_t) \xrightarrow{\text{extract}} \mathcal{G}_t,$$
$$g_t^{(i)} \xrightarrow{\text{match}} \begin{cases} \mathrm{update}(g^*), & \mathrm{Match}{=}1, \\ \mathrm{insert{+}evolve}, & \mathrm{Match}{=}0. \end{cases}$$

Stage 1: existence checking. For each candidate $g_t^{(i)}$, we retrieve top-k neighbors from $\mathcal{D}_{\mathrm{vec}}$ above a cosine similarity threshold τ, then ask an LLM judge whether any neighbor is semantically equivalent. This two-stage design favors recall during retrieval and precision during verification: the threshold τ keeps unrelated goals out of the LLM call, while the judge filters surface-level matches that share keywords but differ in intent. Matched candidates go to Stage 2; novel ones to Stage 3.
Stage 2: update. On a match, the lifecycle is advanced (e.g., PENDING→COMPLETED), slot values are refreshed, and dependencies are preserved.
Stage 3: insert and evolve. On no match, the goal is inserted into $\mathcal{D}_{\mathrm{sym}}$ and $\mathcal{D}_{\mathrm{vec}}$, and an LLM classifies its relation to each top-k neighbor as a soft *link*, a hard *dependency*, or unrelated, updating the dependency graph. This step is essential because new goals in advanced TOD are often conditioned on existing ones.
Stage 4: proactive auditing. A background process periodically inspects OPEN/PENDING goals against the latest context and triggers valid transitions, preventing stale states for asynchronously evolving goals whose status may change without explicit user restatement.

Together, these four stages maintain consistent, dependency-aware goal states throughout the dialogue and form the substrate on which ATOD-Eval's metrics (§6) are computed.

6 Evaluation Metrics and Framework

ATOD-Eval assesses both *whether* a system completes tasks and *how* it manages complex, interdependent dialogues, along three dimensions: task

Table 2: Selected operational dimensions of representative TOD benchmarks; a coverage summary, not a quality comparison. "Goal Status Anno." marks explicit per-turn lifecycle labels. ✓: present, ✗: absent.

Dataset	Avg. Turns	Async.	Dependency	Interleaving	Proactivity	Goal Status Anno.
MultiWOZ (Budzianowski et al., 2018)	13	✗	✗	✗	✗	✗
SGD (Rastogi et al., 2020)	20	✗	✗	✗	✗	✗
TOPDIAL (Wang et al., 2023)	12	✗	✗	✗	✓	✗
MS-TOD (Du et al., 2025)	7	✗	✓	✓	✗	✓
TOAD (Liu et al., 2024b)	5	✗	✓	✓	✓	✗
LUCID (Stacey et al., 2024)	21	✓	✓	✗	✗	✓
ATOD (Ours)	**27**	✓	✓	✓	✓	✓

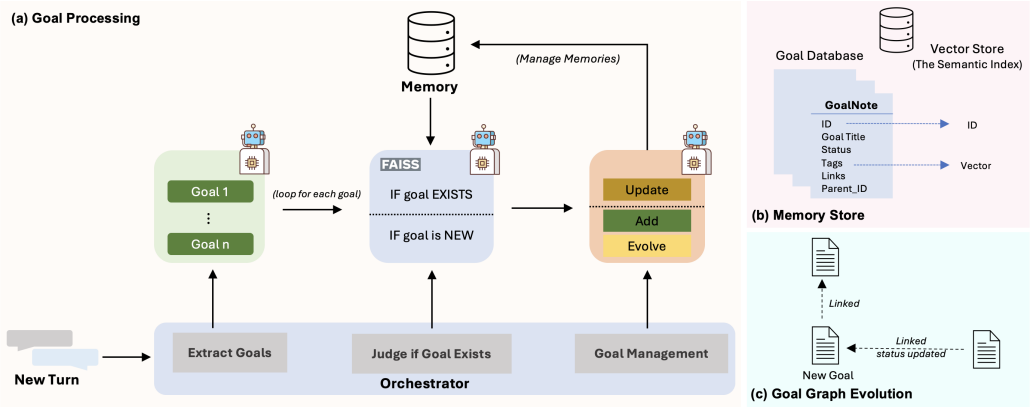

Figure 2: Architecture of the agentic memory system. (a) Turn-level goal extraction, existence checking, updating/inserting, and proactive auditing. (b) Dual memory store with symbolic metadata and semantic embeddings. (c) Dependency graph evolution for newly inserted goals, with explicit links and status transitions.

Table 3: ATOD statistics by dialogue complexity.

Statistic	Medium	Complex	Overall
Dialogues	428	572	1,000
Avg. turns	17.4	33.5	26.6
Avg. goals	3.7	8.5	6.5
With dependencies	18.7%	73.4%	50.0%

completion and efficiency (§6.1), agentic capability (§6.2), and response quality (§6.3). The three dimensions are complementary: task completion alone does not reveal whether a system manages goals proactively or recalls memory faithfully, and conversational quality alone does not certify dependency-aware progression. The framework supports both offline and online evaluation.

6.1 Task Completion and Efficiency

We evaluate whether goals are accomplished and how efficiently they progress through the dialogue.

Dependency-Aware Goal Completion Rate (dGCR). Conventional completion metrics penalize systems for goals that remain unresolved, even when those goals are blocked by unmet prerequisites. We therefore evaluate only *decided* goals (terminal status Completed or Failed). Let $S(g)$ denote the terminal status recorded in the evaluator's symbolic store $\mathcal{D}_{\text{sym}}$ and $\mathcal{T}$ the two terminal states; dGCR thus reflects how the dialogue resolves under the evaluator's reconstruction and is read alongside Goal Detection F1 and Status Tracking Accuracy (Table 4), which compare against ground truth:

$$\mathcal{T} = \{\text{Completed}, \text{Failed}\},$$
$$\text{dGCR} = \frac{|\{g \in \mathcal{U} : S(g) = \text{Completed}\}|}{|\{g \in \mathcal{U} : S(g) \in \mathcal{T}\}|}.$$

A goal is *blocked* if any declared prerequisite remains non-terminal; it is *abandoned* when the user explicitly cancels or replaces it. Blocked goals remain non-terminal, and abandoned goals are excluded because cancellation is not a system failure. For example, if g_1 is completed, g_2 failed, and g_3 pending (blocked by g_2), then the denominator is $\{g_1, g_2\}$ and $\text{dGCR} = 1/2$.

Turns to Completion (NTC). The average number of turns between goal initiation and terminal status, complementing dGCR with an efficiency signal. NTC is best interpreted relative to dialogue length; Table 6 shows the expected increase in NTC on complex dialogues, where goals are more fre-

quently suspended and resumed.

6.2 Agentic Capability Metrics

Beyond task success, we assess agentic behaviors such as memory recall and proactivity.

Memory Recall Accuracy (MRA). At each turn where a goal status changes, we construct a retrieval query q_t from the affected goal description and recent context using a fixed template (Appendix C.3). For each $q_t \in \mathcal{Q}$, an LLM judge compares the system retrieval $\hat{M}_t$ against a ground-truth memory snapshot M_t derived from the annotation and returns $r_t \in \{0, 1\}$ based on sufficiency and semantic consistency. $\text{MRA} = \frac{1}{|\mathcal{Q}|} \sum r_t$, which favors useful memory over surface-form overlap.

Proactivity Effectiveness (PE). A response is flagged as proactive if it references a goal not in the current user turn but present in the tracked state (e.g., a reminder about a pending booking). For each such response $p_t \in \mathcal{P}$, an LLM judge returns $b_t \in \{0, 1\}$ indicating whether the action is both *grounded* (supported by context) and *beneficial* (advances an unresolved goal without redundancy). $\text{PE} = \frac{1}{|\mathcal{P}|} \sum b_t$. Judge templates are in Appendix C.3.

6.3 Response Quality Metrics

Following (Liu et al., 2023; Zheng et al., 2023), *turn-level relevance* scores each response against local context, current request, and tracked goal state (averaged over turns), and *dialogue-level coherence* judges the full conversation for consistency and natural progression across interleaved goals. Together, these scores indicate whether improved goal management comes at the cost of interaction quality. Templates: Appendix C.3.

6.4 Evaluation Framework

ATOD-Eval is evaluator-agnostic: any system producing per-turn goal states can be scored, in either an **offline** setting on completed dialogues or an **online** setting that tracks evolving trajectories in real time. The online setting is enabled by metrics that depend only on per-turn snapshots, allowing monitoring of deployed agents without waiting for end-of-dialogue.

7 Experimental Setup

We evaluate along three axes: *capability*—goal detection F1 and status tracking accuracy at the final state and across normalized dialogue progress; *validity*—Pearson's r and Spearman's ρ between dGCR and the other metrics; and *efficiency*—per-turn latency and token usage. Implementation details: Appendix C.1.

Baselines. All baselines are evaluated zero-shot on the same 1,000-dialogue set. (i) *LLM-based judges*—DeepSeek-V3.2 (Liu et al., 2025), Kimi-K2.5 (Team et al., 2026), MiniMax-M2.5, Mistral-Large-3, GLM-5 (Zeng et al., 2026)—are prompted to infer goal status from full dialogue context with no external memory (Kazi et al., 2024). (ii) *Memory-based evaluators*—Mem0 (Chhikara et al., 2025), MemoChat (Lu et al., 2023), MemoryBank (Zhong et al., 2024), RAG (Lewis et al., 2020), A-Mem (Xu et al., 2025), LLM-Rsum (Wang et al., 2025)—are adapted to our schema by prompting the shared backbone to classify each memory unit into our six lifecycle states, preserving each method's original retrieval logic.

Architecture-isolated comparison. Every memory-based method, including ours, uses the same backbone LLM and embedding model (`all-MiniLM-L6-v2`); only the memory architecture varies. We use Claude-Sonnet-4 as the backbone for all memory-based evaluators (details in Appendix C.1). The prompt-only judges are themselves the LLM under test, serving as a capacity reference for what prompting alone achieves at varying model scales. All LLM calls use temperature 0 with a fixed seed, making results deterministic; we therefore report single-run numbers without confidence intervals.

Mitigating LLM circularity. Because the evaluator and the metric judges may share a model family, we provide two checks: a cross-family judge swap with Qwen3-32B (Yang et al., 2025) and a manual audit of judge decisions on a stratified subset (§8.4).

8 Results

8.1 Evaluation of the Memory System

Table 4 reports goal detection F1 and status tracking accuracy by complexity. The two columns must be read jointly: status accuracy is computed only over detected goals, so a system can miss many goals yet still appear competitive on the second column. DeepSeek-V3.2 (52.12/81.72 on medium) is the clearest case. Mistral-Large-3 (55.73/85.98 on complex) and RAG (85.59/94.85 on medium) show the same pattern more subtly. Among the eleven baselines, only our evaluator simultaneously

Table 4: Goal detection F1 and status tracking accuracy (%) by dialogue complexity. Status tracking is measured over detected goals; interpret jointly with detection quality. Shaded rows indicate memory-based methods sharing the same backbone LLM. **Bold**: our method; underline: best baseline in each column.

Category	Method	Medium		Complex	
		Goal Det. F1	Status Acc.	Goal Det. F1	Status Acc.
LLM-based	DeepSeek-V3.2	52.12	81.72	45.96	65.16
	Kimi-K2.5	66.70	92.90	66.90	82.80
	MiniMax-M2.5	72.20	90.20	66.60	83.90
	Mistral-Large-3	60.51	89.85	55.73	85.98
	GLM-5	71.42	93.52	64.84	83.42
Memory-based	Mem0 (Chhikara et al., 2025)	76.80	75.60	61.10	71.20
	MemoChat (Lu et al., 2023)	80.38	73.88	58.07	66.83
	MemoryBank (Zhong et al., 2024)	82.56	94.23	76.86	78.50
	RAG (Lewis et al., 2020)	85.59	94.85	87.13	77.83
	A-Mem (Xu et al., 2025)	87.30	89.50	62.00	80.60
	LLM-Rsum (Wang et al., 2025)	93.83	89.47	89.13	69.95
	Ours	**91.92**	**92.31**	**86.49**	**84.28**

exceeds 90% on both axes for medium dialogues. On complex dialogues, no baseline exceeds ours on both metrics simultaneously (86.49/84.28).

Architecture-isolated block. All memory-based methods share the same backbone and embedding (§7), so differences reflect memory architecture alone. LLM-Rsum obtains higher complex F1 (89.13 vs. 86.49) but much lower status accuracy (69.95 vs. 84.28); A-Mem and Mem0 drop sharply in complex detection. The F1 gap arises because our existence-checking stage merges semantically equivalent goals that flat retrieval treats as distinct, trading detection recall for status precision. These patterns support separating lifecycle state from semantic retrieval.

Capacity reference. The strongest prompt-only F1 trails the best memory-based F1 by more than 20 points in both splits. Table 5 isolates backbone capacity: memory adds >10 F1 points; status accuracy is within ±1, since the backbone already classifies status well once goals are detected.

Table 5: Same-backbone comparison between prompt-only judging and our agentic memory evaluator.

Method	Medium		Complex	
	F1	Acc.	F1	Acc.
Prompt-only	78.95	93.26	75.58	84.26
Ours	**91.92**	**92.31**	**86.49**	**84.28**
Δ	+12.97	−0.95	+10.91	+0.02

Online behavior. Figure 3 shows both metrics remain stable as the dialogue unfolds, with mild degradation only on complex cases, enabling ATOD-Eval's online setting (§6.4) without waiting for end-of-dialogue. Overall, no baseline dominates both axes: LLM-Rsum detects more goals but trails in complex status tracking and incurs >7× latency (§8.2); the higher medium status accuracy of RAG comes with lower detection F1. Our evaluator achieves the strongest joint performance on complex dialogues at the lowest per-turn cost.

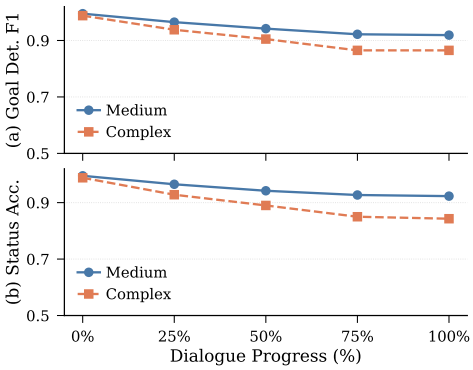

Figure 3: Goal detection F1 (top) and status tracking accuracy (bottom) versus normalized dialogue progress on medium and complex dialogues.

8.2 Efficiency Analysis

Figure 4 reports per-turn latency and token usage. Our evaluator is the most efficient on both axes (<25 s per turn on complex dialogues; ∼2,000/2,400 tokens for medium/complex). RAG and A-Mem follow with moderate latency, Mem0 incurs higher overhead, and LLM-Rsum exceeds 180 s with more than twice the tokens because it re-summarizes the full history at each turn, placing

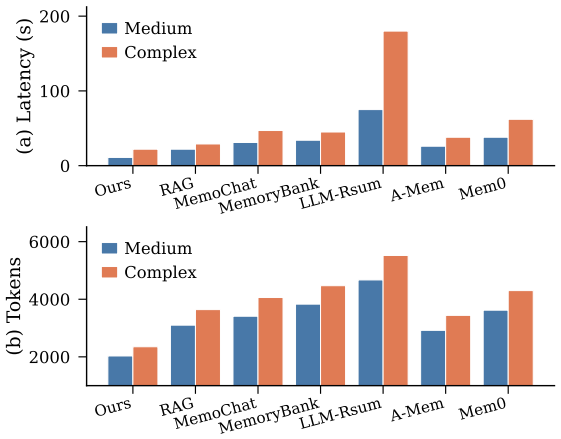

Figure 4: Per-turn update latency (top) and token usage (bottom) across methods.

our method on the Pareto frontier of accuracy and latency.

8.3 Metric Validity Analysis

Table 6: Average results of the proposed evaluation metrics across medium and complex dialogues. All metrics are on a 0–1 scale except Dialogue-level Coherence (1–5) and NTC (turn count).

Metric	Medium	Complex
dGCR	0.967	0.930
NTC	7.04	10.50
MRA	0.913	0.743
PE	0.619	0.586
Turn-level Relevance	0.752	0.766
Dialogue-level Coherence	4.40	4.45

Table 6 summarizes the proposed metrics; medium dialogues yield higher MRA and lower NTC than complex ones, confirming that the metrics respond differently to complexity. Table 7 reports correlations with dGCR (internal alignment, not independent ground truth). MRA correlates most strongly; PE is only marginally correlated and serves as a proactivity diagnostic rather than a task-success signal.

8.4 Validity of LLM Judges

Manual audit of LLM judge decisions. We manually inspected 2,880 judge outputs from 100 stratified dialogues (50 medium, 50 complex), marking each decision as correct or incorrect against the dialogue context and reference annotation. Table 8 reports 93.8% overall agreement with the LLM judge. Agreement is highest on goal status transitions (94.2%) and dependency relations (96.8%), and lower on proactivity *benefit* (81.6%), consistent with the inherent subjectivity of that dimension. The audit protocol is detailed in Appendix B.1.

Table 7: Correlation of proposed metrics with dGCR, reported as Pearson's r and Spearman's ρ under Medium and Complex settings.

Metric	Medium		Complex	
	r	ρ	r	ρ
NTC	+0.08	+0.16	+0.20	+0.05
MRA	+0.75	+0.60	+0.44	+0.43
PE	−0.05	−0.03	+0.16	+0.12
Turn-level Relevance	+0.22	+0.29	+0.08	+0.09
Dialogue-level Coherence	−0.11	−0.08	+0.13	+0.25

Table 8: Agreement between LLM judge decisions and manual audit on 100 stratified dialogues (2,880 items).

Metric Dimension	# Items	Audit Agr.
Goal Status Transition	1,850	94.2%
Dependency Relation	650	96.8%
Proactivity (grounded)	190	95.3%
Proactivity (beneficial)	190	81.6%
Overall	2,880	**93.8%**

Cross-family judge swap. Replacing the default backbone judge with `Qwen3-32B` (Yang et al., 2025) shifts our scores by −1.82/−1.71 on medium F1/Acc and −3.39/+1.02 on complex. This tests the robustness of the evaluator scores across judge families; it is not a reranking of all baselines.

Table 9: Judge swap for our evaluator (F1 / status accuracy).

Judge	Medium		Complex	
	F1	Acc.	F1	Acc.
Default LLM Judge	91.92	92.31	86.49	84.28
`Qwen3-32B`	90.10	90.60	83.10	85.30
Δ	−1.82	−1.71	−3.39	+1.02

9 Conclusions

We introduced **ATOD** and **ATOD-Eval** for evaluating interleaved, dependency-aware TOD with turn-level lifecycle annotations, and a symbolic–vector memory evaluator that achieves balanced detection and tracking on complex dialogues at the lowest per-turn latency among matched-backbone baselines. A cross-family judge swap and a manual audit of 2,880 decisions confirm measurement validity.

Limitations

LLM-based construction and evaluation. ATOD relies on LLMs at multiple stages, with some risk of circularity. Dataset construction uses a verifier pass and author-performed manual screening, while evaluation robustness is checked through a cross-family judge swap and a separate single-author audit of metric judgments. These checks reduce evident errors but do not constitute independently replicated multi-annotator validation.

Limited component-level ablations. We report a top-k sweep and a same-backbone comparison that together isolate the contribution of the memory layer, but do not yet ablate proactive auditing or the dependency graph in isolation.

Scope. ATOD is a held-out synthetic benchmark of English, text-only dialogues derived from SGD domains. The benchmark measures goal-trajectory recovery rather than live user success, and does not model personalization or multimodal inputs.

Ethical Considerations

ATOD is a synthetic benchmark built from the public Schema-Guided Dialogue (SGD) dataset and contains no real user conversations or PII. However, LLM-generated dialogues may inherit biases present in the underlying models. Staged quality control is applied, but downstream users should account for these limitations.

The benchmark rewards proactive agent behavior, which carries dual-use risks: systems optimized for proactivity could become intrusive or manipulative without proper user controls. We recommend evaluating proactive behaviors jointly with user preference sensitivity. Additionally, the notion of "beneficial" proactivity may not generalize across cultural contexts or to user populations with different communication norms. The dataset is intended solely for research purposes.

Use of AI Assistants

Generative AI was used for language editing during manuscript revision. All claims, citations, and results were checked by the authors.

References

Emre Can Acikgoz, Carl Guo, Suvodip Dey, Akul Datta, Takyoung Kim, Gokhan Tur, and Dilek Hakkani-Tür. 2025. Td-eval: Revisiting task-oriented dialogue evaluation by combining turn-level precision with dialogue-level comparisons. *arXiv preprint arXiv:2504.19982*.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. Multiwoz–a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. *arXiv preprint arXiv:1810.00278*.

Amartya Chakraborty, Paresh Dashore, Nadia Bathaee, Anmol Jain, Anirban Das, Shi-Xiong Zhang, Sambit Sahu, Milind Naphade, and Genta Indra Winata. 2026. T1: A tool-oriented conversational dataset for multi-turn agentic planning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

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

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Sam Davidson, Salvatore Romeo, Raphael Shu, James Gung, Arshit Gupta, Saab Mansour, and Yi Zhang. 2023. User simulation with large language models for evaluating task-oriented dialogue. *arXiv preprint arXiv:2309.13233*.

Yang Deng, Wenxuan Zhang, Wai Lam, Hong Cheng, and Helen Meng. 2022. User satisfaction estimation with sequential dialogue act modeling in goal-oriented conversational systems. In *Proceedings of the ACM Web Conference 2022*, pages 2998–3008.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The faiss library.

Yiming Du, Bingbing Wang, Yang He, Bin Liang, Baojun Wang, Zhongyang Li, Lin Gui, Jeff Z Pan, Ruifeng Xu, and Kam-Fai Wong. 2025. Bridging the long-term gap: A memory-active policy for multi-session task-oriented dialogue. *arXiv preprint arXiv:2505.20231*.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. 2024. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*.

Arihant Jain, Purav Aggarwal, Rishav Sahay, Chaosheng Dong, and Anoop Saladi. 2025. Autoeval-tod: Automated evaluation of task-oriented dialog systems. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human*

Language Technologies (Volume 1: Long Papers), pages 10133–10148.

Taaha Kazi, Ruiliang Lyu, Sizhe Zhou, Dilek Hakkani-Tür, and Gokhan Tur. 2024. Large language models as user-agents for evaluating task-oriented-dialogue systems. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, pages 913–920. IEEE.

Abishek Komma, Nagesh Panyam Chandrasekarasastry, Timothy Leffel, Anuj Goyal, Angeliki Metallinou, Spyros Matsoukas, and Aram Galstyan. 2023. Toward more accurate and generalizable evaluation metrics for task-oriented dialogs. *arXiv preprint arXiv:2306.03984*.

Atharva Kulkarni, Bo-Hsiang Tseng, Joel Ruben Antony Moniz, Dhivya Piraviperumal, Hong Yu, and Shruti Bhargava. 2024. Synthdst: Synthetic data is all you need for few-shot dialog state tracking. *arXiv preprint arXiv:2402.02285*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Zhiyu Li, Shichao Song, Hanyu Wang, Simin Niu, Ding Chen, Jiawei Yang, Chenyang Xi, Huayi Lai, Jihao Zhao, Yezhaohui Wang, et al. 2025. Memos: An operating system for memory-augmented generation (mag) in large language models. *arXiv preprint arXiv:2505.22101*.

Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, et al. 2025. Deepseek-v3. 2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. 2024a. Agentbench: Evaluating LLMs as agents. In *The Twelfth International Conference on Learning Representations*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: Nlg evaluation using gpt-4 with better human alignment. *arXiv preprint arXiv:2303.16634*.

Yinhong Liu, Yimai Fang, David Vandyke, and Nigel Collier. 2024b. Toad: Task-oriented automatic dialogs with diverse response styles. *arXiv preprint arXiv:2402.10137*.

Junru Lu, Siyu An, Mingbao Lin, Gabriele Pergola, Yulan He, Di Yin, Xing Sun, and Yunsheng Wu. 2023. Memochat: Tuning llms to use memos for consistent long-range open-domain conversation. *arXiv preprint arXiv:2308.08239*.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. 2024. Evaluating very long-term conversational memory of llm agents. *arXiv preprint arXiv:2402.17753*.

Lingbo Mo, Shun Jiang, Akash Maharaj, Bernard Hishamunda, and Yunyao Li. 2024. Hiertod: A task-oriented dialogue system driven by hierarchical goals. *arXiv preprint arXiv:2411.07152*.

Kai Tzu-iunn Ong, Namyoung Kim, Minju Gwak, Hyungjoo Chae, Taeyoon Kwon, Yohan Jo, Seungwon Hwang, Dongha Lee, and Jinyoung Yeo. 2024. Towards lifelong dialogue agents via timeline-based memory management. *arXiv preprint arXiv:2406.10996*.

OpenAI. 2024. Introducing chatgpt agent. https://openai.com/index/introducing-chatgpt-agent/. Accessed: July 2025.

Charles Packer, Vivian Fang, Shishir_G Patil, Kevin Lin, Sarah Wooders, and Joseph_E Gonzalez. 2023. Memgpt: Towards llms as operating systems.

Baolin Peng, Chunyuan Li, Zhu Zhang, Chenguang Zhu, Jinchao Li, and Jianfeng Gao. 2020. Raddle: An evaluation benchmark and analysis platform for robust task-oriented dialog systems. *arXiv preprint arXiv:2012.14666*.

Manik Rana, Calissa Man, Anotida Expected Msiiwa, Jeffrey Paine, Kevin Zhu, Sunishchal Dev, Vasu Sharma, et al. 2025. Agentchangebench: A multi-dimensional evaluation framework for goal-shift robustness in conversational ai. *arXiv preprint arXiv:2510.18170*.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8689–8696.

Sahil Sen, Elias Lumer, Anmol Gulati, and Vamse Kumar Subbiah. 2026. Chronos: Temporal-aware conversational agents with structured event retrieval for long-term memory. *arXiv preprint arXiv:2603.16862*.

Joe Stacey, Jianpeng Cheng, John Torr, Tristan Guigue, Joris Driesen, Alexandru Coca, Mark Gaynor, and Anders Johannsen. 2024. Lucid: Llm-generated utterances for complex and interesting dialogues. *arXiv preprint arXiv:2403.00462*.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, SH Cai, Yuan Cao, Y Charles, HS Che, Cheng Chen, Guanduo Chen, et al. 2026. Kimi k2. 5: Visual agentic intelligence. *arXiv preprint arXiv:2602.02276*.

Marilyn Walker, Candace Kamm, and Diane Litman. 2000. Towards developing general models of usability with paradise. *Natural Language Engineering*, 6(3-4):363–377.

Jian Wang, Yi Cheng, Dongding Lin, Chak Tou Leong, and Wenjie Li. 2023. Target-oriented proactive dialogue systems with personalization: Problem formulation and dataset curation. *arXiv preprint arXiv:2310.07397*.

Qingyue Wang, Yanhe Fu, Yanan Cao, Shuai Wang, Zhiliang Tian, and Liang Ding. 2025. Recursively summarizing enables long-term dialogue memory in large language models. *Neurocomputing*, 639:130193.

Yaxiong Wu, Yongyue Zhang, Sheng Liang, and Yong Liu. 2025. Sgmem: Sentence graph memory for long-term conversational agents. *arXiv preprint arXiv:2509.21212*.

Wujiang Xu, Kai Mei, Hang Gao, Juntao Tan, Zujie Liang, and Yongfeng Zhang. 2025. A-mem: Agentic memory for llm agents. *arXiv preprint arXiv:2502.12110*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. 2024. tau-bench: A benchmark for tool-agent-user interaction in real-world domains. *arXiv preprint arXiv:2406.12045*.

Yejin Yoon, Yuri Son, Namyeong So, Minseo Kim, Minsoo Cho, Chanhee Park, Seungshin Lee, and Taeuk Kim. 2025. Beyond task-oriented and chitchat dialogues: Proactive and transition-aware conversational agents. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 13291–13317.

Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, et al. 2026. Glm-5: from vibe coding to agentic engineering. *arXiv preprint arXiv:2602.15763*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

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

A ATOD Dataset: Statistics, Examples, and Construction Details

A.1 Annotated Dialogue Example

Table 10 presents a representative medium-complexity dialogue from ATOD, illustrating interleaving, dependency management, and goal abandonment. The dialogue involves four goals across three domains (Calendar, Services, Flights). Goal 2 (salon appointment) depends on Goal 1 (availability check); Goal 3 (SF flight) is abandoned mid-dialogue when the user changes plans to Chicago (Goal 4). Turn-level status annotations track each goal's lifecycle progression.

A.2 Synthetic Dataset Quality Analysis

As aggregate evidence for the construction-time verifier, we assess synthetic dialogues along five dimensions using an LLM quality rater following Liu et al. (2023). Table 11 provides automated quality-control scores, complementing the author-performed manual screening described in Appendix A.6.

A.3 Goal Extraction, Co-occurrence Graph Statistics, and Sampling Strategy

We first extract goal sequences from the SGD dataset, where each sequence is an ordered list of user goals (domain–intent pairs) within a dialogue. Table 12 summarizes the extraction results. All 10,739 sequences are multi-domain, with an average length of 3.90 goals and a range of 2–8 goals. These sequences span 16 unique domains and 37 unique intents (of which 36 appear in the final ATOD benchmark), providing a rich basis for building the co-occurrence graph.

We then construct a goal co-occurrence graph where each node is a unique goal and edges represent co-occurrence within the same sequence. Table 13 shows its statistics. The graph contains 52 nodes and 396 edges, forming a single connected component with relatively high density (0.2986) and average degree (15.23), indicating frequent goal co-occurrence across dialogues. This structure supports a diverse sampling of multi-goal trajectories, including high-degree hubs (up to 29) and rare goals (degree as low as 2).

We sample goal trajectories from this graph by selecting connected subgraphs that satisfy the desired complexity criteria (§A.4), ensuring diversity in goal count, domain coverage, and dependency patterns.

Table 10: Annotated ATOD dialogue excerpt (medium complexity, 4 goals, 3 domains). Status changes shown per turn. Goals: G1=check availability, G2=book salon (depends on G1), G3=book SF flight, G4=book Chicago flight.

Turn	Speaker	Utterance	Status Changes
1	User	I need to check Sarah Johnson's availability on October 12th, between 2 and 5 PM.	G1→OPEN
2	System	Let me look that up for you right away.	G1→PENDING
3	User	While you're checking that, I also need a one-way flight from Boston to San Francisco on June 12th.	G3→OPEN
4	System	I'll work on both. Sarah has availability from 2–4 PM. I've also started searching flights for you.	G1→COMPL., G3→PENDING
5	User	Great—I'd like to book a haircut with her at Bella Salon for 3:30 PM on October 12th.	G2→OPEN
6	System	I'll book that appointment. Meanwhile, I found several Boston–SF flights.	G2→PENDING
7	User	Actually, change the flight to Chicago O'Hare instead, same date. Is there a United flight?	G3→ABAND., G4→OPEN
8	System	Your salon appointment is confirmed for Oct 12 at 3:30 PM. Searching Chicago flights now.	G2→COMPL., G4→PENDING
11	User	Yes, book that United flight for me.	G4→COMPL.
15	User	No need to book SF flights. So I have the salon on Oct 12 and the Chicago flight on June 12?	—
17	User	That covers everything. Thanks!	—

Table 11: LLM-based quality ratings on a 1–5 Likert scale.

Dimension	Medium	Complex
Coherence	4.04	3.92
Fluency	5.00	5.00
Consistency	4.42	4.04
Relevance	4.58	4.62
Naturalness	4.04	4.04

Table 12: Summary statistics for extracted goal sequences.

Statistic	Value
Total Goal Sequences	10,739
Avg. Sequence Length	3.90
Length Range	2–8
Unique Domains	16
Unique Intents	37

Table 13: Summary statistics for the co-occurrence graph.

Statistic	Value
Total Nodes (Unique Goals)	52
Total Edges (Co-occurrences)	396
Graph Density	0.2986
Average Degree	15.23
Max Degree	29
Min Degree	2

Table 14: Criteria for medium vs. complex dialogues. Columns: Goals, Turns, Async. (asynchronous), Inter. (interleaving), Dep. (dependencies), Proac. (proactivity), Def. (defectiveness). ✓ = present, ✗ = absent. Ambiguous cases are resolved using domain diversity, dependency depth, and behaviors.

Compl.	Goals	Turns	Async.	Inter.	Dep.	Proac.	Def.
Medium	2–8	8–35	✓	✓	≤ 2	✗	✗
Complex	7+	30+	✓	✓	≥ 2	✓	✓

A.4 Complexity Criteria

Our pipeline (§ 4) uses a two-category complexity system (*medium* vs. *complex*), combining quantitative thresholds with qualitative LLM analysis for balanced distribution. Table 14 shows the criteria based on goals, turns, domains, and advanced agentic behaviors.

For the categorization process, we follow a three-step procedure. First, *goal sampling* draws trajectories under a two-category distribution (default: 65% medium, 35% complex; the final dataset composition may differ slightly due to filtering and quality control). Second, *annotation* enriches sampled goals with slots, dependencies, and realistic characteristics. Third, *hybrid classification* assigns complexity using pre-defined rules combined with LLM analysis, considering quantitative factors (goal count, domain diversity, dependency structures), qualitative factors (goal interdependence and coordination complexity), and realistic dialogue requirements such as interleaving and proactivity needs.

A.5 Annotated Trajectories and Metadata Specification

To represent complex goal structures and agentic behaviors in ATOD, we define a formal schema for dialogue trajectories and metadata (§4.2), shown in Listing 1. The schema captures interleaved goals, slot-filling states, and explicit inter-goal dependencies that arise in advanced TOD settings. Metadata fields encode global dialogue attributes and execution characteristics (e.g., interleaving, proactivity, and asynchronous actions), which are later used to condition dependency-aware and turn-level evaluation signals. Each goal entry specifies its intent, slots, and dependency relations, enabling systematic analysis of goal initiation, suspension, and resumption across multi-goal interactions.

```
{
  "dialogue_id": "string",
  "complexity_class": "medium | complex"
      ,
  "metadata": {
    "num_goals": "integer",
    "estimated_turns": "integer",
    "async_execution": "boolean",
    "interleaving": "boolean",
    "proactivity": "boolean"
    },
  "goal_list": [
    {
      "id": "string",
      "domain": "string",
      "intent": "string",
      "slots": ["string", ...],
      "slot_values": {
        "slot_name_1": "value1",
        "slot_name_2": "value2"
      },
      "dependencies": ["goal_id", ...],
      "content": "string",
      "core_content": "string",
      "classification_method": "
          pre_defined | model_based",
      "dependency_label": "boolean",
      "defectiveness_label": "boolean"
    }
    // ...more goals
  ]
}
```

Listing 1: Annotation schema for dialogue trajectories and goal-level metadata.

A.6 ATOD: Quality Control

We apply layered screening during construction. First, before dialogue generation, a verifier judge checks whether each annotated trajectory has clear goals, plausible slots, complete fields, and consistent dependencies. Second, after dialogue generation and status annotation, a dedicated verification pass separate from the generation call checks whether the dialogue realizes its trajectory, whether lifecycle annotations agree with the utterances, and whether the interaction remains coherent and plausible; flagged instances are retried or filtered. Finally, an author manually screens generated dialogues and their status annotations for these failure types. The pre-generation verifier prompt is shown below.

A.7 Dialogue Generation Prompt

Below, we present the prompt template used in §4.3 for LLM-based dialogue generation. Placeholders are filled from the annotated trajectory metadata (§A.5).

Quality Assessment Prompt
`You are a quality judge for annotated goal trajectories.`
Input: TRAJECTORY ({num_goals} goals, {complexity} complexity): {goals_text}
Task: Assess whether this trajectory is ready for dialogue generation. Check:
- Goal descriptions are clear and specific - Slot values are realistic (no placeholders)
- All required fields are present - Annotations are logically consistent
Output format: Respond with exactly one word: `PASS` or `FAIL`

Table 15: Prompt template for quality assessment of annotated trajectories.

Dialogue Generation Prompt Template
`Generate a realistic task-oriented dialogue between USER and SYSTEM.`
Requirements: Complexity: {complexity} Length: {estimated_turns} turns Goals: {goal_descriptions} Attributes: {agentic_attrs}
{combined_guidance} {outcome_guidance}
Dialogue Structure: (1) User introduces goals naturally; (2) System works under realistic constraints; (3) Natural obstacles and preference changes may occur; (4) Dialogue ends at a natural stopping point; (5) Goal completion may be partial or conditional.
Natural Conversation Patterns: Users express needs as they arise; system handles practical constraints; users may add, revise, or abandon goals; availability or technical limitations may surface.
Format: Alternating USER/SYSTEM turns, starting with USER.

Table 16: Prompt template for dialogue generation.

A.8 Goal Status Annotation Prompt

Below, we present the prompt template used in §4.4 for turn-level goal status annotation. Listing 2 provides a sample annotated dialogue instance.

```
{
  "dialogue_id": "...",
  "complexity_class": "complex",
  "metadata": {
    "num_goals": ...,
    "num_turns": ...,
    "async_execution": true,
    "interleaving": true,
    "proactivity": true
  },
  "goal_list": [...],
  "turns": [
    {
      "turn_id": 1,
      "speaker": "USER",
      "utterance": "I need to book a
          hotel in Chicago.",
      "goal_status_changes": [
```

Goal Status Annotation Prompt Template
`You are tracking goal status in a task-oriented dialogue. Analyze only the current turn and update statuses based on what actually happens.`
Current Turn: {last_turn} **Goals and Current Statuses:** {goal_descriptions}
Status Meanings:
- `NOT_MENTIONED`: not yet appeared - `OPEN`: mentioned, no action - `PENDING`: system working
- `COMPLETED`: successfully finished - `FAILED`: failed - `ABANDONED`: user cancelled
Critical Rules: (1) Change status only if something definitive occurs; (2) `PENDING` → only `COMPLETED/FAILED/ABANDONED`; (3) Preserve status if no clear change.
Terminal States: {goal_id: current_status, . . . } **JSON Template:** {json_template}
Instruction: Respond with only the JSON, updating only goals whose status clearly changes.

Table 17: Prompt template for turn-level goal status annotation.

```
        {"goal_id": "g1", "new_status":
            "open"}
      ],
      "all_goals": {
        "g1": "open",
        "g2": "not_mentioned",
        "g3": "not_mentioned"
      }
    }
    // ... remaining turns omitted
  ]
}
```

Listing 2: Sample annotated ATOD dialogue with turn-level status tracking.

B Robustness and Sensitivity Analyses

This appendix details the manual audit of metric judgments and top-k retrieval sensitivity. The cross-family judge-swap results are reported in §8.4. All analyses use the same 1,000-dialogue test set unless otherwise noted.

B.1 Manual Audit of Metric Judgments

Protocol. We sample a stratified subset of 100 dialogues (50 medium, 50 complex) and extract 2,880 individual judge decisions across four dimensions: goal status transitions, dependency relations, proactivity grounding, and proactivity benefit. Each decision is inspected against the dialogue context, reference annotation, and metric definition, and marked as acceptable or not. Table 18 reports agreement between the LLM decisions and this manual audit.

Table 18: Agreement between LLM judge decisions and manual audit on 100 stratified dialogues (2,880 items).

Metric	# Items	Audit Agr.
Goal Status Transition	1,850	94.2%
Dependency Relation	650	96.8%
Proactivity (grounded)	190	95.3%
Proactivity (beneficial)	190	81.6%
Overall	2,880	93.8%

Findings. The audit agrees with the LLM judge on 93.8% of items overall. Agreement is highest on goal status transitions (94.2%) and dependency relations (96.8%), while proactivity *benefit* is lowest (81.6%), consistent with the subjectivity of that dimension. These results validate the metric judges and are distinct from the construction-time screening in Appendix A.6.

B.2 Top-k Retrieval Sensitivity

The top-k retrieval hyperparameter controls how many neighbors are considered during existence checking (Stage 1 in §5.2). We sweep $k \in \{1, 3, 5, 7, 9\}$ to assess sensitivity. All other settings remain at their default values (Table 19).

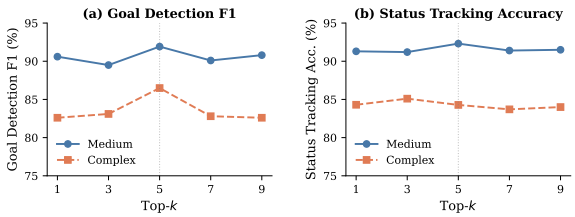

Figure 5: Top-k retrieval sensitivity. Goal detection F1 (left) and status tracking accuracy (right) as functions of top-k on the 1,000-dialogue test set. The default k=5 is marked by the dotted vertical line.

Performance is stable across all values of k, with goal detection F1 varying by less than 2.5 percentage points on medium dialogues and status tracking accuracy varying by less than 1.5 points. The default k=5 achieves the best overall performance, particularly on complex dialogues where it outperforms other settings by 3–4 points in goal F1. Smaller values (k=1, 3) risk missing relevant neighbors during existence checking, while larger values (k=7, 9) introduce noise from less relevant candidates. Overall, these results indicate that the evaluator is not brittle to this hyperparameter choice.

C Evaluator Implementation and Prompt Templates

C.1 Implementation Details

Our memory system is instantiated with `Claude-Sonnet-4` (accessed via the Amazon Bedrock API) as the primary LLM judge for goal extraction, existence checking, relationship analysis, and proactive status auditing. For embedding-based retrieval, we use `all-MiniLM-L6-v2` (384-dim) sentence embeddings indexed with FAISS. Inter-goal links are determined by an LLM call over the top-k semantically related goals rather than by a fixed similarity cutoff. Table 19 lists the final implementation settings used for all reported results.

All evaluation runs use Claude-Sonnet-4 via the Amazon Bedrock API (temperature 0, seed 42). The prompt-only baselines in Table 4 are themselves the LLM under test, while every memory-based baseline (Mem0, MemoChat, MemoryBank, RAG, A-Mem, LLM-Rsum) is run with Claude-Sonnet-4 as the backbone LLM and `all-MiniLM-L6-v2` as the embedding model, identical to our evaluator. This isolates architectural differences from raw model capacity within the memory-based block. To address potential circularity between the evaluator and the metric judges, we report a cross-family judge swap with Qwen3-32B and a manual audit on a stratified subset in §8.4; protocol details are in Appendix B.1. FAISS retrieval is run on a single NVIDIA A10 GPU; LLM calls dominate end-to-end latency.

Both hyperparameters were selected on a small held-out development set of 50 dialogues drawn from the ATOD generation pipeline using the same procedure as the test split, then frozen before the 1,000-dialogue evaluation. The existence-match threshold τ=0.40 was tuned to favor recall over precision during goal matching: a lower threshold ensures semantically related but differently phrased goals are linked, with the subsequent LLM verification step filtering false positives. Top-k=5 retrieval balances coverage against prompt length; most dialogues in the development set contained fewer than 10 concurrent goals, so k=5 retrieves the majority of relevant candidates without exceeding context limits. Sensitivity to k on the full test set is reported in Appendix B.2.

Table 19: Implementation settings for the agentic memory system.

Setting	Value
Primary LLM Judge	Backbone LLM (see text)
Embedding Model	`all-MiniLM-L6-v2` (384-d)
Vector Store / Index Type	FAISS `IndexFlatIP`
Similarity Metric	Cosine (IP on L2-normalized)
Top-k Retrieval	5
Existence-Match Threshold τ	0.40
Proactive Auditing Frequency	Every turn

C.2 Agentic Memory System Templates

As detailed in §5, the agentic memory system is implemented through a set of modular LLM prompt templates. We present three templates in sequence, corresponding respectively to (i) goal extraction from individual conversation turns, (ii) turn-level goal status classification, and (iii) goal graph evolution for establishing inter-goal links and dependencies. Together, these templates support structured, consistent, and interpretable memory management across multi-turn dialogues. The prompts list five active labels; the sixth state `NOT_MENTIONED` is the default for goals that have not yet appeared in the dialogue and is assigned implicitly (no LLM call required).

Goal Extraction Prompt

`Extract user goals from this conversation turn. Use standardized core_content patterns.`

Conversation Turn:
User: {user_utterance}
System: {system_response}

Core Content Patterns (examples):
- "book hotel" — hotels, rentals - "book flight" — flights
- "book ticket" — bus, concert, train
- "check account" — balance, account info - "search restaurant" — find restaurants

Status Labels: `OPEN`, `PENDING`, `COMPLETED`, `FAILED`, `ABANDONED`

Output format (JSON array):
`[{"goal_content": "...", "core_content": "...", "status": "OPEN"}, ...]`

Table 20: Prompt template for goal extraction in the agentic memory system.

C.3 Evaluation Prompt Templates

To make the evaluation protocol explicit, we also provide the prompt templates used for LLM-based judgment of memory recall, proactive behavior, and response quality. In all cases, the judge is given the dialogue context, the system output under evaluation, and the corresponding benchmark reference derived from ATOD annotations.

Goal Status Classification Prompt

`Analyze this conversation turn and classify the status of the specific goal below.`
Goal to classify: "{goal_content}"

Conversation Turn:
User: {user_utterance} System: {system_response}

Status Definitions:
- `OPEN`: mentioned, no action yet - `PENDING`: system processing
- `COMPLETED`: successfully achieved - `FAILED`: explicitly failed - `ABANDONED`: cancelled by user

Transition Examples:
"book a flight" → OPEN "which dates?" → PENDING "flight booked" → COMPLETED
"no flights available" → FAILED "never mind" → ABANDONED

Output format (JSON): `{"status": "STATUS"}`

Table 21: Prompt template for per-turn goal status classification.

Goal Evolution Prompt

`Analyze relationships between a new goal and existing related goals.`
New Goal: Content: {new_goal.content} Core Content: {new_goal.core_content}
Related Goals (top-k by semantic similarity): {related_goals_context}
Task: For each related goal, determine the relationship type:
- `"link"`: semantically related but independent
- `"dependency"`: new goal depends on the related goal
- `"none"`: no significant relationship

Output format (JSON): `{"goal_id_1": "relationship_type", "goal_id_2": "relationship_type"}`

Table 22: Prompt template for goal graph evolution.

Memory Recall Accuracy Prompt

`You are evaluating whether a retrieved memory output correctly matches the benchmark memory state.`
Dialogue Context: {dialogue_context}
Retrieval Query: {memory_query}
Ground-Truth Memory State: {gold_memory_json}
Retrieved Memory Output: {predicted_memory}

Judging Criteria:
- Retrieved output contains information required to answer the query
- Goal statuses, slot values, and historical facts are semantically consistent with gold state
- No contradiction is introduced

Output format (JSON): `{"match": 0 or 1}`

Table 23: Prompt template for Memory Recall Accuracy evaluation.

Proactivity Effectiveness Prompt

`You are evaluating whether a system's proactive action is contextually grounded and genuinely helpful.`
Dialogue Context: {dialogue_context}
Current System Action: {system_action}
Relevant Goal State: {goal_state_json}

Judging Criteria:
- The action is not explicitly requested by the user in the current turn
- The action is grounded in prior dialogue and memory state
- The action advances an unresolved or dependency-unlocked goal, or provides a timely reminder
- The action is not irrelevant, redundant, or distracting

Output format (JSON): `{"beneficial": 0 or 1}`

Table 24: Prompt template for Proactivity Effectiveness evaluation.

Response Quality Prompt

`You are evaluating the response quality of a task-oriented dialogue system.`
Dialogue Context: {dialogue_context}
Response Under Evaluation: {system_response}
Goal State / Dialogue Reference: {goal_state_or_full_dialogue}
Turn-Level Relevance: addresses current request; consistent with tracked goals; useful and natural
Dialogue-Level Coherence: globally consistent; interleaved goals progress coherently; no contradictions
Output format (JSON): `{"turn_quality": 1-5, "dialogue_quality": 1-5}`

Table 25: Prompt template for Response Quality evaluation.