ATOD: AN EVALUATION FRAMEWORK AND BENCHMARK FOR AGENTIC TASK-ORIENTED DIALOGUE SYSTEMS

Yifei Zhang *
Amazon
jimmyzyf@amazon.com

Hooshang Nayyeri
Amazon
hooshang@amazon.com

Rinat Khaziev
Amazon
rinatk@amazon.com

Emine Yilmaz
Amazon
University College London
eminey@amazon.co.uk

Gokhan Tur
Amazon
University of Illinois Urbana–Champaign
gokturt@amazon.com

Dilek Hakkani-Tür
Amazon
University of Illinois Urbana–Champaign
hakkanit@amazon.com

Hari Thadakamalla
Amazon
thadakah@amazon.com

ABSTRACT

Recent advances in task-oriented dialogue (TOD) systems, driven by LLMs with extensive API and tool integration, have expanded the scope of conversational agents beyond traditional turn-by-turn task execution. Modern systems are increasingly expected to coordinate interleaved goals, preserve long-horizon context, and provide proactive assistance under asynchronous execution. However, existing benchmarks do not systematically evaluate these agentic behaviors. To address this gap, we introduce **ATOD**, a benchmark and synthetic dialogue generation pipeline that produces richly annotated conversations requiring long-horizon reasoning. ATOD captures key characteristics of *Advanced TOD*, including multi-goal coordination, dependency management, long-horizon memory, and proactivity. Building on ATOD, we propose **ATOD-Eval**, a holistic evaluation framework that operationalizes these dimensions through fine-grained metrics and supports reproducible evaluation in both offline and online settings. We further present an agentic memory-based evaluator for benchmarking models on ATOD. Experiments show that ATOD-Eval enables comprehensive assessment of task completion, agentic capability, and response quality, and that the proposed evaluator provides a favorable accuracy–efficiency trade-off relative to strong memory-based and LLM-based baselines under this evaluation setting.

1 INTRODUCTION

Task-oriented dialogue (TOD) systems have traditionally been studied as conversational agents that satisfy user requests through sequential, turn-by-turn interactions Rastogi et al. (2020); Cheng et al. (2022); Wang et al. (2023). With the emergence of LLMs and agentic AI systems such as ChatGPT OpenAI (2024), Claude Anthropic (2024), and Gemini Comanici et al. (2025), expectations for TOD systems have increased substantially. Users now expect advanced capabilities, including the ability to manage multiple objectives concurrently (*multi-goal concurrency*), continue progress while waiting for external API or tool responses (*asynchronous execution*), and flexibly suspend or resume objectives during a dialogue (*interleaved workflows*). They also expect *proactivity*, in which the system offers useful assistance without digression while dynamically handling evolving *goal*

*Work performed during internship at Amazon.

Table 1: Illustrative dialogue that highlights key agentic capabilities of advanced TOD systems.

User Utterance	Agent Response	Capabilities
Book a flight to Chicago for next Monday.	I found several options. Would you like me to confirm one now and also reserve a hotel near your destination?	MGC
Yes, please. Also remind me to pack my passport the night before.	Flight confirmed and hotel reserved. A reminder has also been scheduled for Sunday night.	LHM; MGC
What is the weather in Chicago on Monday?	Forecast for Monday in Chicago: rainy, high of 12 °C.	INT
Also book dinner at an Italian restaurant for Monday night after my flight.	Your flight lands at 6:30 pm. A table at *Luigi Trattoria* is reserved for 8:00 pm.	DEP; MGC
Arrange transportation from the airport to the hotel.	Transportation is scheduled for 6:45 pm. Would you like the driver information now or closer to the date?	ASY; PRO
Closer to the date, please.	Understood. I will send the driver information the day before your trip.	LHM; AFU

Abbreviations. MGC: multi-goal concurrency; LHM: long-horizon memory; INT: interleaving; DEP: dependency management; ASY: asynchronous execution; PRO: proactivity; AFU: asynchronous follow-up.

dependencies. *Long-horizon memory* is equally important, because agents must integrate immediate conversational context with persistent knowledge across extended or multi-session interactions.

Table 1 shows a representative interaction that motivates the evaluation challenges studied in this work. In such interactions, agents must coordinate interdependent goals, preserve context over time, and support asynchronous progress in non-sequential dialogues. Together, these characteristics define what we refer to as *Advanced TOD*. Evaluating such systems requires assessment not only of response quality and task completion, but also of how these aspects interact in complex dialogue settings. Despite substantial progress in automatic evaluation Liu et al. (2023); Dubois et al. (2024); Zheng et al. (2023); Li et al. (2024); Yao et al. (2024); Jain et al. (2025); Acikgoz et al. (2025) and TOD dataset construction Budzianowski et al. (2018); Rastogi et al. (2020); Du et al. (2025); Wang et al. (2023); Kulkarni et al. (2024), most benchmarks do not capture the advanced characteristics described above, leaving these capabilities underexplored. In parallel, although recent work has started to evaluate dialogue systems with memory components Xu et al. (2025); Chhikara et al. (2025); Maharana et al. (2024); Ong et al. (2024), existing approaches do not provide standardized protocols for assessing long-horizon retention, adaptive updates, and the management of interleaved goals with complex dependencies. This limitation points to the need for a unified benchmark and a holistic evaluation framework for systematic assessment of advanced TOD behavior under realistic and complex interaction scenarios.

To address this need, we introduce **ATOD**, a *benchmark* and synthetic dialogue generation pipeline that produces richly annotated dialogues requiring long-horizon memory, interleaved workflows, and explicit goal dependencies. Building on ATOD, we propose **ATOD-Eval**, a holistic evaluation framework that provides standardized benchmarks and fine-grained metrics for advanced TOD capabilities. ATOD-Eval integrates benchmarking and evaluation by jointly assessing goal completion, dependency management, memory consistency, proactivity, and multi-goal coordination, translating these dimensions into reproducible metrics for both offline and online settings. We further present an **agentic memory-based evaluator** for benchmarking models on ATOD, which enables empirical comparison with strong memory-based and LLM-based baselines. Extensive experiments validate the proposed benchmark and evaluation framework, showing that the resulting metrics provide a comprehensive and consistent assessment of advanced TOD capabilities, while the proposed evaluator delivers a strong accuracy–efficiency trade-off under this evaluation setting.

2 Related Work

2.1 Evaluation of TOD Systems

Automatic evaluation frameworks such as G-Eval Liu et al. (2023), AlpacaEval Dubois et al. (2024), and MT-Bench Zheng et al. (2023) are designed for open-domain dialogue and focus primarily on fluency and coherence rather than on goal-driven or memory-dependent behavior. In TOD, early work emphasized turn-level user satisfaction Walker et al. (2000); Schmitt & Ultes (2015); Bodigutla et al. (2019), and later work extended evaluation to the dialogue level through frameworks such as RoBERTaIQ Gupta et al. (2021), USDA Deng et al. (2022), and DQM Komma et al. (2023).

Other studies evaluate task completion with zero-shot LLM judges Kazi et al. (2024) or interactive protocols that rely on user simulators Sun et al. (2021); Cheng et al. (2022); Davidson et al. (2023). More recent benchmarks, including AutoTOD Xu et al. (2024), FNCTOD Li et al. (2024), τ-Bench Yao et al. (2024), AutoEval-ToD Jain et al. (2025), and TD-EVAL Acikgoz et al. (2025), focus mainly on inform and success rates and therefore do not capture the broader capabilities required in advanced TOD.

2.2 TOD Datasets and Benchmarks

Human-curated datasets such as MultiWOZ Budzianowski et al. (2018), SGD Rastogi et al. (2020), RADDLE Peng et al. (2020), τ-Bench Yao et al. (2024), and MS-TOD Du et al. (2025) support dialogue state tracking and task completion, but provide limited support for long-horizon or multi-session memory. Most of these datasets are confined to single sessions with narrowly scoped goals. Synthetic datasets, including TOPDIAL Wang et al. (2023), TOAD Liu et al. (2024), LUCID Stacey et al. (2024), and SynthDST Kulkarni et al. (2024), introduce personalization and proactivity, but they still do not adequately support advanced agentic behavior.

2.3 Memory in Dialogue Systems

Memory mechanisms are essential for retaining context and managing goals over extended interactions. Early approaches such as RAG Lewis et al. (2020), MemoChat Lu et al. (2023), and MemoryBank Zhong et al. (2024) support session-level recall through retrieval, summarization, or history storage, but they do not maintain persistent memory across sessions. More recent agentic memory architectures, including MemGPT Packer et al. (2023), A-Mem Xu et al. (2025), mem0 Chhikara et al. (2025), and MemOS Li et al. (2025), introduce structured mechanisms for long-term retention. Other studies, such as LoCoMo Maharana et al. (2024), THEANINE Ong et al. (2024), and MAP Du et al. (2025), evaluate memory along temporal or efficiency dimensions. However, most prior work studies memory in isolation and does not provide standardized protocols that connect memory use with goal management in advanced TOD settings.

3 Problem Formulation

3.1 Characteristics of Advanced TOD

Advanced TOD differs from conventional sequential TOD in several important ways. **Multi-Goal Concurrency.** Users may pursue multiple objectives at the same time, requiring agents to track and manage parallel goals with distinct states. **Interleaving.** Goals may be suspended, resumed, and alternated across contexts rather than following a strictly sequential workflow. **Long-Horizon Memory.** Goals may span many turns, so agents must preserve consistent state and dependency information over extended interactions. **Asynchronous Execution.** Some goals are delayed, for example while waiting for external confirmation, which requires agents to maintain a Pending state and resume execution when conditions are satisfied. **Proactivity.** Agents should take initiative by reminding users of pending tasks or suggesting relevant actions in context, including **asynchronous follow-up**, in which the system later re-contacts the user about a deferred or time-sensitive goal.

3.2 Task Formulation

Formally, let $\mathcal{D} = \{(\mathcal{G}_i, \mathcal{C}_i)\}_{i=1}^{N}$ denote a dialogue corpus, where $\mathcal{C}_i = \{c_{i,t}\}_{t=1}^{T_i}$ is the ordered sequence of dialogue turns and $\mathcal{G}_i$ is the associated set of user goals with explicit dependencies. Unlike in traditional TOD settings, where each goal is usually confined to a contiguous span, in advanced TOD a single goal $g \in \mathcal{G}_i$ may span disjoint intervals of $\mathcal{C}_i$ and may be initiated, suspended, and resumed as the dialogue evolves. We represent each goal through a *goal status trajectory* $\{\mathrm{Status}(g,t)\}_{t=1}^{T_i}$, such as Open $\rightarrow$ Pending $\rightarrow$ Completed/Failed/Abandoned, which captures a non-contiguous and interleaved lifecycle over extended interactions. The evaluation objective is to assess how well a system manages interdependent goals, maintains long-horizon trajectories, coordinates asynchronous workflows, and provides proactive support in both *offline* and *online* settings.

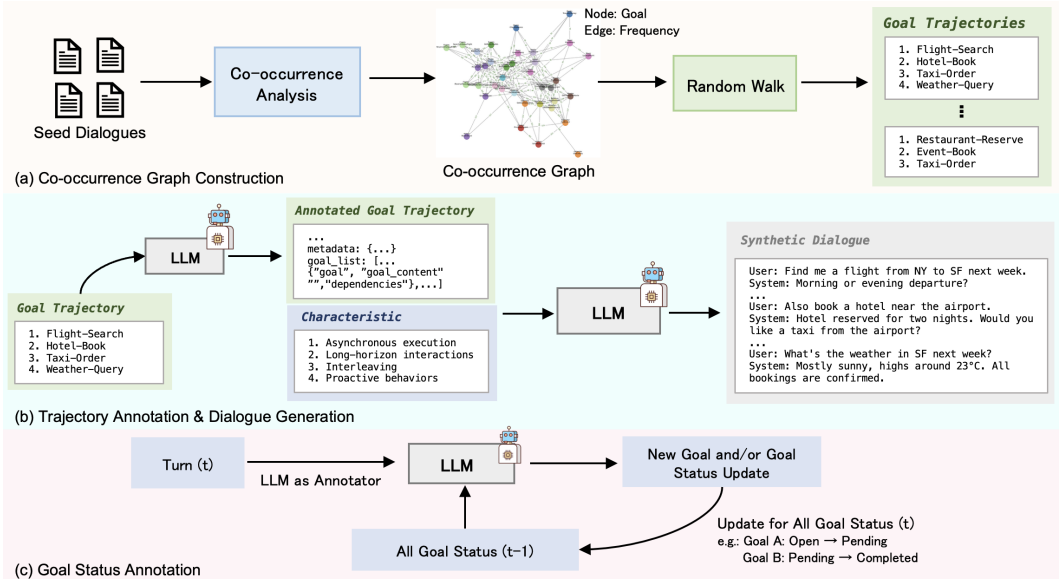


Figure 1: ATOD data curation pipeline. (a) **Co-occurrence Graph & Trajectory Sampling** (§4.1): construct a goal co-occurrence graph from seed dialogues and sample diverse multi-goal trajectories through random walks; (b) **Trajectory Annotation & Dialogue Generation** (§4.2–§4.3): an LLM annotates slot values, inter-goal dependencies, and complexity, and then generates agentic multi-turn dialogues conditioned on the trajectories; (c) **Goal Status Annotation** (§4.4): at each turn, an LLM labels active goals and updates lifecycle states, enabling fine-grained tracking of dialogue progress.

4 ATOD: A Synthetic Dialogue Dataset and Generation Pipeline

Motivated by the challenges above, we present **ATOD**, a synthetic dataset and generation pipeline designed to benchmark the advanced TOD characteristics introduced in §3.1. ATOD consists of richly annotated, memory-intensive dialogues that explicitly encode these characteristics and thereby enable systematic evaluation. As illustrated in Figure 1, the dataset is constructed through a modular LLM-driven pipeline (§4.1–§4.4) with quality control at each stage. We further analyze dataset coverage and compare ATOD with prior benchmarks in §4.5.

4.1 Co-occurrence Graph Construction and Goal Trajectory Sampling

To generate realistic synthetic dialogues, it is important to capture how goals naturally co-occur in user interactions. Independent goal sampling often produces implausible combinations, whereas fixed templates limit diversity. To address this issue, we construct a goal co-occurrence graph $G = (V, E)$ from an underlying dialogue dataset, where each node represents a unique goal and each weighted edge reflects empirical co-occurrence frequency. As shown in Figure 1(a), candidate goal sets $S = \{g_1, \ldots, g_k\}$ are sampled through stratified random walks of varying lengths over G, preserving realistic correlations while increasing diversity. This procedure is dataset-agnostic; in our experiments, we instantiate it with the Schema-Guided Dialogue (SGD) corpus Rastogi et al. (2020). The resulting dialogues exhibit richer domain variation and naturally support multi-goal, interleaved, and long-horizon interactions.

4.2 Annotation of Goal Trajectories and Complexity Categorization

As illustrated in Figure 1(b), each sampled goal set S is instantiated into a concrete trajectory through LLM-based annotation, yielding (i) slot values, (ii) inter-goal dependencies D_S that capture prerequisite or blocking relations (for example, Payment depending on Booking), and (iii) natural-language goal descriptions. Each trajectory receives a complexity label $c(S)$ that reflects both quantitative attributes, such as the number of goals and dependency density, and qualitative factors, such as interleaving or opportunities for proactivity; detailed criteria are provided in Ap-

pendix A.3. This categorization ensures coverage across complexity levels and enables structured evaluation of agentic behavior. Quality control is applied throughout the pipeline: the LLM filters duplicate or incompatible goals during sampling, and automatic retries together with LLM-based checks verify slot validity, dependency consistency, and linguistic fluency during annotation (Appendix A.5). This multi-stage quality control pipeline validates intermediate outputs step by step to reduce error propagation and improve logical coherence, as reflected in the quality scores reported in Appendix A.1.

4.3 DIALOGUE GENERATION

As shown in Figure 1(b), dialogue synthesis is conditioned on the annotated trajectory $\tau = (S, D_S, c(S))$. The goals, dependencies, and target complexity profile are combined into structured prompts for LLM-based generation (templates are provided in Appendix A.6). The LLM then generates a natural multi-turn conversation $\mathcal{C} = \{c_t\}_{t=1}^{T}$ that realizes the specified goals while exhibiting interleaving, asynchronous execution, proactive assistance, and dependency-aware coordination.

4.4 TURN-LEVEL GOAL STATUS ANNOTATION

Finally, as illustrated in Figure 1(c), an LLM annotator performs iterative turn-level analysis to label the status of each goal at every dialogue turn. Each utterance c_t is annotated with an active goal set $\mathcal{A}_t \subseteq S$ and corresponding statuses $\mathrm{Status}(g, t) \in \{$NOT_MENTIONED, OPEN, PENDING, COMPLETED, FAILED, ABANDONED$\}$. This design allows goals to be initiated, suspended, resumed, or terminated over time rather than being confined to contiguous spans. The resulting turn-aligned `status_history` provides a rich reference for benchmarking multi-goal tracking and asynchronous or interleaved progress.

4.5 DATASET COVERAGE

ATOD spans diverse domains and goal complexities, ranging from simple two-goal cases to interdependent long-horizon workflows. Through our modular pipeline, we generated a total of 1,000 richly annotated multi-turn dialogues spanning medium and complex settings. Because ATOD-Eval is a zero-shot benchmark designed to evaluate existing LLMs and agentic systems, it does not involve parameter training; accordingly, all 1,000 dialogues are used as the test set. Table 2 compares ATOD with representative prior benchmarks. Although existing datasets capture individual aspects of advanced TOD, ATOD uniquely combines multi-goal concurrency, interleaving with asynchronous execution, explicit dependency modeling, proactive behavior, and turn-level status annotation. This design makes ATOD the first dataset specifically constructed to support comprehensive evaluation of advanced TOD systems.

Table 2: Comparison of ATOD with representative TOD benchmarks. "Average Turns" denotes the per-dialogue average. The rightmost column (*Goal Status Anno.*) indicates whether explicit per-turn labeling of the lifecycle state of each goal is available (for example, PENDING, COMPLETED, or FAILED). Other columns indicate support for key agentic features: asynchronous goal management, explicit dependency modeling, interleaving, and proactive behavior (✓: present, ✗: absent).

Dataset	Average Turns	Async	Dependency	Interleaving	Proactive	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. (2024)	5	✗	✓	✓	✓	✗
LUCID Stacey et al. (2024)	21	✓	✓	✗	✗	✓
ATOD (Ours)	54	✓	✓	✓	✓	✓

5 AN AGENTIC MEMORY-BASED EVALUATOR

Building on ATOD, we introduce the *agentic memory-based evaluator* of ATOD-Eval (Fig. 2), which is implemented as an *agentic memory system*. Whereas ATOD provides annotated dialogues

for benchmarking, this evaluator assesses models directly from dialogue text by checking whether goal trajectories are maintained and updated consistently throughout the interaction. The evaluator contains two main components: (i) a **dual memory store** (§5.1) and (ii) a **turn-level processing pipeline** (§5.2).

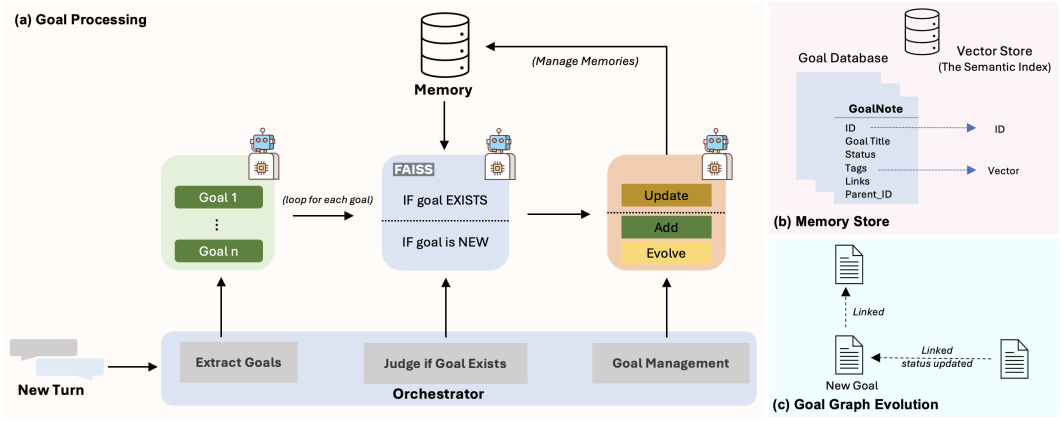


Figure 2: Architecture of the agentic memory system. (a) Turn-level pipeline for goal extraction, existence checking, updating or inserting, and proactive auditing. (b) Dual memory store with symbolic metadata and semantic embeddings. (c) Evolution of the dependency graph when new goals are inserted, with explicit links and status transitions.

5.1 Dual Memory Store

As shown in Fig. 2(b), the memory system maintains a *dual memory store* consisting of (i) a **structured goal database** $\mathcal{D}_{\text{sym}}$, which persistently records symbolic metadata such as goal content and status, and (ii) a **semantic vector store** $\mathcal{D}_{\text{vec}}$ (for example, FAISS Douze et al. (2024)), which indexes embeddings of goal metadata for similarity-based retrieval.

Each goal g is stored as {`id`, `status`, `status_history`, `goal_description`, `dependencies`, `parent_id`, `embedding`}, where `status` $\in$ {OPEN, PENDING, COMPLETED, FAILED, ABANDONED}; `status_history` records turn-level transitions; `goal_description` provides a standardized textual representation; `dependencies` and `parent_id` encode inter-goal relations; and `embedding` supports semantic retrieval during interaction. NOT_MENTIONED is used only in the per-turn annotations of ATOD for goals that have not yet appeared in the dialogue; the memory store records only instantiated goals. This dual design combines precise symbolic state tracking with flexible semantic matching and thereby supports robust memory management over long-horizon dialogues.

5.2 Turn-Level Processing Pipeline

At each dialogue turn t, the memory system applies a structured *turn-level processing pipeline* to maintain the lifecycle of all goals. The pipeline consists of five stages: (i) **goal extraction** from the current utterance and context, (ii) **existence checking** against the dual memory store, (iii) **updating** matched goals, (iv) **inserting and evolving** new goals with dependency evolution, and (v) **proactive auditing** to keep active states consistent. This design supports dynamic tracking and interleaving of multi-goal trajectories over long horizons.

Formally, given user utterance u_t and context c_t, the system extracts candidate goals $\mathcal{G}_t$. Each candidate $g_t^{(i)} \in \mathcal{G}_t$ is matched against the memory store to determine whether an existing entry should be updated or a new one should be inserted:

$$(u_t, c_t) \xrightarrow{\text{extract}} \mathcal{G}_t$$

$$g_t^{(i)} \xrightarrow{\text{match}} \begin{cases} \text{update}(g^*), & \text{if Match} = 1, \\ \text{insert} + \text{evolve}(g_t^{(i)}), & \text{if Match} = 0. \end{cases}$$

Stage 1. Goal Extraction. The system extracts candidate goals $\mathcal{G}_t$ from the current turn and dialogue context.

Stage 2. Existence Checking. For each candidate $g_t^{(i)}$, the system retrieves the top-k neighbors $\mathcal{N}_k(g_t^{(i)})$ from $\mathcal{D}_{\text{vec}}$ and applies an LLM-based judge f_{judge} for semantic verification. If the confidence score is at least τ, then $\text{Match} = 1$; otherwise, $\text{Match} = 0$.

Stage 3. Updating Existing Goals. When $\text{Match} = 1$, the *Update* module advances the lifecycle of the goal, for example from PENDING to COMPLETED, refreshes slot values and dependencies, and preserves existing inter-goal relations.

Stage 4. Adding and Evolving New Goals. When $\text{Match} = 0$, the new goal is inserted into both $\mathcal{D}_{\text{sym}}$ and $\mathcal{D}_{\text{vec}}$. The *Evolve* module links the new goal to related goals $\{g_k \mid \text{rel}(g_t^{(i)}, g_k) \geq \delta\}$ and updates the directed dependency graph $G = (\mathcal{V}, \mathcal{E})$ to support interleaved workflows. Capturing these dependencies is essential in advanced TOD, where goals are often logically conditioned on others, such as PAYMENT following BOOKING. Maintaining these relations prevents premature completion and supports faithful modeling of complex task dynamics.

Stage 5. Proactive Status Tracking. Beyond event-driven updates, a background auditing process periodically inspects active goals (OPEN, PENDING) against dialogue context and tool outputs. An LLM judge triggers valid transitions, for example from PENDING to COMPLETED, which prevents stale states and preserves coherence across dependent goals.

Together, these modules maintain consistent and dependency-aware goal states, thereby supporting concurrency and reliable evaluation in advanced TOD.

6 ATOD-EVAL: EVALUATION METRICS AND FRAMEWORK

Having introduced ATOD as a benchmark dataset (§4.5) and the agentic memory-based evaluator that tracks evolving goals (§5), we now present the evaluation framework of **ATOD-Eval**. This framework defines metrics and protocols that assess not only *whether* a system completes tasks, but also *how effectively* it manages complex and interdependent dialogues. ATOD-Eval covers three dimensions: (i) *Task Completion and Efficiency* (§6.1), (ii) *Agentic Capability Metrics* (§6.2), and (iii) *Response Quality Metrics* (§6.3). A unified evaluation framework (§6.4) supports both offline and online evaluation.

6.1 TASK COMPLETION AND EFFICIENCY

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

Dependency-Aware Goal Completion Rate (dGCR). Conventional goal completion metrics treat all goals equally and therefore unfairly penalize a system when some goals remain blocked by unmet prerequisites. We define a dependency-aware variant that evaluates only goals whose prerequisites have been satisfied. Formally, let $S(g)$ denote the status of goal g in $\mathcal{D}_{\text{sym}}$, and let $\mathcal{U}$ denote the set of *dependency-decidable* goals, , goals whose prerequisites are satisfied and can therefore be evaluated fairly. Define $\mathcal{U}_{\text{dec}} = \{g \in \mathcal{U} \mid S(g) \in \{\text{COMPLETED}, \text{FAILED}\}\}$. Then,

$$\text{dGCR} = \frac{|\{g \in \mathcal{U}_{\text{dec}} : S(g) = \text{COMPLETED}\}|}{|\mathcal{U}_{\text{dec}}|}.$$

This formulation avoids bias from dependency-locked goals and provides a faithful measure of system performance in multi-goal workflows.

Turns to Completion (NTC). For each completed goal, NTC computes the average number of turns from initiation to completion, thereby capturing execution efficiency and complementing dGCR.

6.2 AGENTIC CAPABILITY METRICS

Beyond task success, we assess whether systems exhibit agentic behavior such as memory recall and proactive action.

Memory Recall Accuracy. This metric measures how often the memory retrieved by a system matches the ground-truth state, including slot values, goal statuses, and historical context. Concretely, an LLM judge compares the retrieved memory state against the ground-truth goal metadata and determines whether the retrieved content is semantically consistent at the turn level. These judgments are then averaged across the dialogue.

Proactivity Effectiveness. We evaluate proactive behavior by identifying *proactive events*, in which the system initiates a goal or a state change without an explicit user request in the same turn, and then judging whether the action is contextually appropriate and beneficial. This metric is evaluated with an LLM judge that determines whether the proactive action meaningfully advances the task or provides necessary information without explicit user prompting. These judgments are aggregated over all proactive events in a dialogue.

6.3 Response Quality Metrics

In addition to task outcomes and agentic behavior, we assess conversational quality, focusing on *turn-level relevance* and *dialogue-level coherence*, following prior work Liu et al. (2023); Dubois et al. (2024); Zheng et al. (2023). These metrics ensure that systems maintain natural and consistent interactions while also managing goals effectively.

6.4 Evaluation Framework

Together, these metrics form a unified framework that jointly evaluates task outcomes, agentic behavior, and conversational quality. ATOD-Eval supports both **offline** benchmark analysis and **online** tracking, enabling consistent assessment across static datasets and real-time deployments.

7 Experimental Setup

We evaluate the capability, validity, and efficiency of the framework through task-oriented and cost-oriented metrics. To assess evaluator capability, we report *Goal Detection F1*, which measures coverage of correctly identified active goals, and *Status Tracking Accuracy*, which measures state classification accuracy among detected goals; these metrics are reported both at the final dialogue state and across normalized dialogue progress. Implementation details are provided in Appendix B.1. To assess metric validity, we analyze correlations with *Dependency-Aware Goal Completion Rate (dGCR)* through Pearson's r and Spearman's ρ. The evaluated metrics include *Turns to Completion (NTC)*, *Memory Recall Accuracy*, *Proactivity Effectiveness*, and subjective response quality at both the turn level and the dialogue level. To assess efficiency, we measure per-turn update latency and average token usage in order to study scalability under increasingly complex dialogue conditions.

We compare with two classes of baselines. The first class comprises **LLM-based judges**: following Kazi et al. (2024), we prompt LLMs (`Claude-3.5-Sonnet`, `Claude-3.7-Sonnet`, `Claude-4-Sonnet`, `DeepSeek-R1`) in a zero-shot setting to infer goal status and task completion. The second class comprises **memory-based evaluators**: we adapt representative memory-augmented frameworks, including RAG Lewis et al. (2020), MemoChat Lu et al. (2023), MemoryBank Zhong et al. (2024), and LLM-Rsum Wang et al. (2025). Because these architectures were developed primarily for open-domain retention, we adapt their prompting strategies to align with our goal status schema and thereby support a fair comparison.

8 Results

8.1 Evaluation of the Memory System

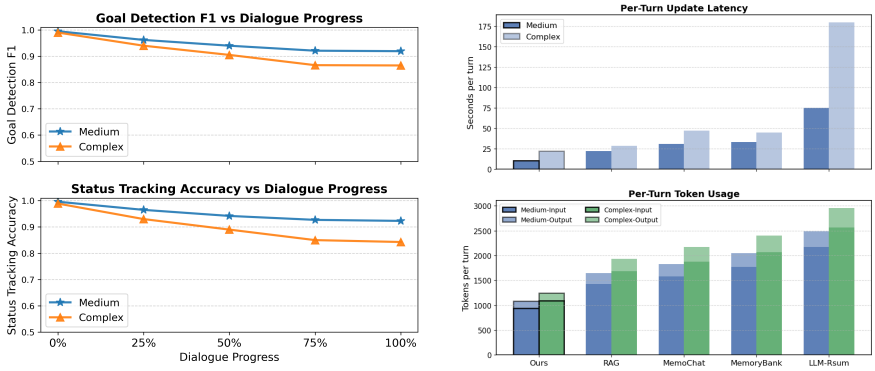
Table 3 reports goal detection and status tracking results for medium and complex dialogues. Memory-based approaches are generally stronger than direct LLM judges, which highlights the importance of explicit memory structures for advanced TOD evaluation. Relative to the baselines, our method provides a strong overall balance in medium settings and achieves the highest status tracking accuracy in complex settings, where several baselines degrade substantially. Figure 3a further analyzes performance as a function of normalized dialogue progress.

Table 3: Comparison of goal detection F1 and status tracking accuracy for each method, broken down by dialogue complexity. All results are reported as percentages (%) and averaged over the test set.

Category	Method	Medium		Complex	
		Goal Detection F1	Status Tracking Accuracy	Goal Detection F1	Status Tracking Accuracy
LLM-based	`DeepSeek-R1`	52.84	96.36	36.63	74.42
	`Claude-3.5-Sonnet`	74.08	92.94	82.92	76.10
	`Claude-3.7-Sonnet`	76.67	92.47	72.97	78.64
	`Claude-4-Sonnet`	78.95	93.26	75.58	84.26
Memory-based	RAG ((Lewis et al., 2020))	85.59	94.85	87.13	77.83
	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
	LLM-Rsum ((Wang et al., 2025))	93.83	89.47	89.13	69.95
	Ours	**91.92**	**92.31**	**86.49**	**84.28**

8.2 Efficiency Analysis

As shown in Figure 3b, our method achieves the lowest mean per-turn update latency across both settings. It also uses fewer input and output tokens in both medium and complex dialogues. This efficiency advantage arises from selective goal matching and lightweight updates, which reduce redundant LLM calls.



(a) Goal detection and status tracking over dialogue progress.

(b) Per-turn latency and token usage.

Figure 3: Online performance and efficiency analysis under medium and complex settings.

8.3 Metric Validity Analysis

We first summarize the average values of the proposed metrics in Table 4a. These metrics reflect complementary dimensions of system behavior, including efficiency, memory, proactivity, and interaction quality. Medium dialogues are shorter and yield higher memory recall, whereas complex dialogues require more turns and exhibit lower recall.

Table 4: Summary statistics and validity analysis of the proposed evaluation metrics under medium and complex settings.

(a) Average results across medium and complex settings.

Metric	Medium	Complex
dGCR	0.967	0.930
# Turns to Completion	7.04	10.50
Memory Recall Accuracy	0.913	0.743
Proactivity Effectiveness	0.619	0.586
Turn-level Quality	0.752	0.766
Dialogue-level Quality	4.40	4.45

(b) Correlation with dGCR (Pearson r, Spearman ρ).

Metric	Medium		Complex	
	r	ρ	r	ρ
Turns to Completion	$+0.08$	$+0.16$	$+0.20$	$+0.05$
Memory Recall Accuracy	$+0.75$	$+0.60$	$+0.44$	$+0.43$
Proactivity Effectiveness	-0.05	-0.03	$+0.16$	$+0.12$
Turn-level Quality	$+0.22$	$+0.29$	$+0.08$	$+0.09$
Dialogue-level Quality	-0.11	-0.08	$+0.13$	$+0.25$

To examine validity, we then analyze correlations with *dGCR* in Table 4b. Among all metrics, *Memory Recall Accuracy* shows the strongest correlation with dGCR in both settings, which highlights the role of accurate memory in dependency-aware task success. *Turns to Completion* and *Turn-level Quality* show weaker but still informative alignment, capturing efficiency and local interaction quality. *Proactivity Effectiveness* exhibits only marginal correlation, which suggests that richer proactive scenarios may be needed to reveal its value more clearly. Overall, the metrics provide complementary views of system behavior: some align closely with dependency-sensitive success, whereas others contribute efficiency-oriented and quality-oriented signals.

9 CONCLUSIONS

We introduced **ATOD**, a benchmark that captures key characteristics of advanced task-oriented dialogue, including multi-goal concurrency, dependency management, long-horizon memory, asynchrony, and proactivity, together with turn-level goal status annotations for fine-grained evaluation. Building on this benchmark, we proposed **ATOD-Eval**, a holistic evaluation framework that translates these capabilities into reproducible metrics for offline and online settings. We further presented an **agentic memory-based evaluator** for benchmarking on ATOD. Experimental results show that, under the proposed evaluation setting, the evaluator achieves a strong overall accuracy–efficiency trade-off, including competitive goal detection, the highest status tracking accuracy in complex dialogues, and substantially lower update latency and token usage than strong LLM-based and memory-based baselines.

REFERENCES

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

Anthropic. Building effective agents. `https://www.anthropic.com/engineering/building-effective-agents`, 2024. Accessed: July 2025.

Praveen Kumar Bodigutla, Longshaokan Wang, Kate Ridgeway, Joshua Levy, Swanand Joshi, Alborz Geramifard, and Spyros Matsoukas. Domain-independent turn-level dialogue quality evaluation via user satisfaction estimation. *arXiv preprint arXiv:1908.07064*, 2019.

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

Qinyuan Cheng, Linyang Li, Guofeng Quan, Feng Gao, Xiaofeng Mou, and Xipeng Qiu. Is multiwoz a solved task? an interactive tod evaluation framework with user simulator. *arXiv preprint arXiv:2210.14529*, 2022.

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.

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

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

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

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

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

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

Saurabh Gupta, Xing Fan, Derek Liu, Benjamin Yao, Yuan Ling, Kun Zhou, Tuan-Hung Pham, and Chenlei Edward Guo. Robertaiq: An efficient framework for automatic interaction quality estimation of dialogue systems. 2021.

Arihant Jain, Purav Aggarwal, Rishav Sahay, Chaosheng Dong, and Anoop Saladi. 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)*, pp. 10133–10148, 2025.

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

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

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

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

Zekun Li, Zhiyu Zoey Chen, Mike Ross, Patrick Huber, Seungwhan Moon, Zhaojiang Lin, Xin Luna Dong, Adithya Sagar, Xifeng Yan, and Paul A Crook. Large language models as zero-shot dialogue state tracker through function calling. *arXiv preprint arXiv:2402.10466*, 2024.

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

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

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

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

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

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

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

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

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

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

Alexander Schmitt and Stefan Ultes. Interaction quality: assessing the quality of ongoing spoken dialog interaction by experts—and how it relates to user satisfaction. *Speech Communication*, 74: 12–36, 2015.

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

Weiwei Sun, Shuo Zhang, Krisztian Balog, Zhaochun Ren, Pengjie Ren, Zhumin Chen, and Maarten de Rijke. Simulating user satisfaction for the evaluation of task-oriented dialogue systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2499–2506, 2021.

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

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

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

Heng-Da Xu, Xian-Ling Mao, Puhai Yang, Fanshu Sun, and He-Yan Huang. Rethinking task-oriented dialogue systems: From complex modularity to zero-shot autonomous agent. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2748–2763, 2024.

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

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

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

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 Appendix A

A.1 Synthetic Dataset Quality Analysis

We assess the quality of synthetic dialogues along five dimensions: coherence, fluency, consistency, relevance, and naturalness Liu et al. (2023). As shown in Table 5, both medium-level and complex-level dialogues achieve strong results across all criteria, particularly in fluency and relevance. These

results indicate that the generation pipeline produces realistic and high-quality conversations suitable for downstream evaluation.

Table 5: LLM-based evaluation of synthetic dialogues along five dimensions. Scores are averaged over medium-level and complex-level dialogues and are reported on a 1–5 Likert scale, where higher values are better.

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

In addition, the pipeline employs a separate LLM-based judge at multiple stages (§4.2, §4.3, and §4.4), including trajectory sampling, goal annotation & classification, and status annotation, to ensure output quality at each step of the generation process. This layered evaluation helps preserve both faithfulness and consistency throughout synthetic dialogue construction.

A.2 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, represented as domain–intent pairs, within a dialogue. Table 6 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, which provides a rich basis for constructing the co-occurrence graph.

Table 6: 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

We then construct a goal co-occurrence graph in which each node corresponds to a unique goal and edges represent co-occurrence within the same sequence. Table 7 reports the graph statistics. The graph contains 52 nodes and 396 edges, forms a single connected component, and exhibits relatively high density (0.2986) and average degree (15.23), which indicates frequent goal co-occurrence across dialogues. This structure supports diverse sampling of multi-goal trajectories, including high-degree hubs (up to 29) and rare goals (with degree as low as 2).

We sample goal trajectories from this graph by selecting connected subgraphs that satisfy the target complexity criteria (§A.3), thereby ensuring diversity in goal count, domain coverage, and dependency patterns.

A.3 Complexity Criteria

Table 7: 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

The pipeline in § 4 uses a two-category complexity system (*medium* and *complex*) that combines quantitative thresholds with qualitative LLM analysis to achieve balanced coverage. Table 8 presents the criteria based on goals, turns, domains, and advanced agentic behavior.

Table 8: Criteria for medium and complex dialogues. Columns: Goals, Turns, Async. (asynchronous), Inter. (interleaving), Dep. (dependencies), Proac. (proactivity), Defective? (whether defective cases are allowed, for example, injected infeasibility or inconsistency). ✓ = present, ✗ = absent. Ambiguous cases are resolved through domain diversity, dependency depth, and observed behavior.

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

The categorization process follows three stages. First, *goal sampling* draws trajectories under a two-category distribution (default: 65% medium, 35% complex). Second, *annotation* enriches sampled goals with slots, dependencies, and realistic interaction characteristics. Third, *hybrid classification* assigns complexity through a combination of predefined rules and LLM analysis, considering quantitative factors (goal count, domain diversity, and dependency structure), qualitative factors (goal interdependence and coordination difficulty), and realistic dialogue requirements such as interleaving and proactivity.

A.4 Annotated Trajectories and Metadata Specification

To represent complex goal structures and agentic behavior 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, such as 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, which enables 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.5 ATOD: QUALITY CONTROL

We use the following LLM-based quality control prompt to verify goal clarity, slot validity, and annotation consistency of annotated goals (§4.2) before dialogue generation.

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:

- The goal descriptions are clear and specific
- The slot values are realistic and contain no placeholders
- All required fields are present
- The annotations are logically consistent

Output format:
Respond with exactly one word: `PASS` or `FAIL`

A.6 DIALOGUE GENERATION PROMPT

Below we present the exact prompt template used in §4.3 for LLM-based dialogue generation. Placeholders (for example, {complexity}, {estimated_turns}, {goal_descriptions}, and {agentic_attrs}) are filled programmatically from the annotated trajectory metadata, as described in §A.4.

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. The user introduces goals naturally throughout the conversation
2. The system works on goals under realistic constraints and limitations
3. Natural obstacles, delays, and preference changes may occur
4. The dialogue ends at a natural stopping point
5. Goal completion may be partial or conditional, reflecting realistic scenarios

Natural Conversation Patterns:

- Users express needs and preferences as they arise
- The system responds helpfully while handling practical constraints
- Users may add, revise, or abandon goals based on new information
- Availability, pricing, or technical limitations may arise
- Conversations conclude when users are satisfied or defer decisions

Format: Alternating USER/SYSTEM turns, starting with USER.

A.7 Goal Status Annotation Prompt

Below we present the exact prompt template used in §4.4 for turn-level goal status annotation. The prompt is instantiated with the current dialogue turn, the list of goals together with their current statuses, and the expected JSON schema. Listing 2 further provides a sample annotated dialogue instance, illustrating how goal status transitions and complete goal states (`all_goals`) are tracked across turns.

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 to Analyze:
{last_turn}

Goals and Current Statuses:
{goal_descriptions}

Status Meanings:

- `NOT_MENTIONED`: the goal exists but has not appeared in the dialogue
- `OPEN`: the goal has been mentioned by the user, but no action has started
- `PENDING`: the system is actively working on the goal
- `COMPLETED`: the goal has been completed successfully
- `FAILED`: the goal failed because of system limitations or availability constraints
- `ABANDONED`: the user cancelled the goal or changed their mind

Critical Rules:

1. Change the status of a goal *only* if something definitive occurs in the current turn
2. `PENDING` goals may transition only to `COMPLETED`, `FAILED`, or `ABANDONED`
3. If no clear change occurs, preserve the existing status

Terminal States (Do Not Change):
{goal_id: current_status, . . . }

Current Statuses (JSON Template):
{`json_template`}

Instruction: Respond with *only* the JSON above, updating *only* goals whose status clearly changes in the current turn.

```
{
  "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_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 Appendix B

B.1 Implementation Details

Using the proposed generation pipeline, we construct a benchmark of 1,000 richly annotated multi-turn dialogues. The dataset follows a predefined complexity distribution, with 65% of dialogues categorized as medium and 35% as complex. Since ATOD-Eval is a zero-shot, prompt-based benchmark designed to evaluate existing LLMs and agentic systems, it does not involve parameter training. Accordingly, all 1,000 dialogues are used as the test set for evaluation. The memory system is instantiated with `Claude-3.7-Sonnet` (accessed through the Amazon Bedrock API) as the primary LLM judge. For embedding-based retrieval, we use `MiniLM-L6-v2` embeddings indexed with FAISS for efficient nearest-neighbor search. Across all experiments, we use fixed values of top-k, the similarity threshold δ, and the LLM decision threshold τ.

B.2 Agentic Memory System Templates

As described in §5, the agentic memory system is implemented through a set of modular LLM prompt templates. We present three templates in sequence, corresponding 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.

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 information
- `"search restaurant"` — restaurant discovery
- `"book restaurant"` — reservations

Status Labels (for extracted goals): `OPEN`, `PENDING`, `COMPLETED`, `FAILED`, `ABANDONED`

Note: `NOT_MENTIONED` does not appear in the extracted list because extraction returns only goals that are triggered or discussed in the current turn.

Output format (JSON array):

```
[ {"goal_content": "...", "core_content": "...", "status": "OPEN"},
...]
```

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 taken
- `PENDING`: the system is processing the goal or requesting information
- `COMPLETED`: the goal has been achieved successfully
- `FAILED`: the goal has failed explicitly
- `ABANDONED`: the goal has been cancelled by the 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"}

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`: the new goal depends on the related goal
- `none`: no significant relationship

Output format (JSON):

```
{
  "goal_id_1": "relationship_type",
  "goal_id_2": "relationship_type"
}
```