BOOKMARKS: CHECKPOINT-ACCELERATED ACTIVE GROUNDING FOR ROLE-PLAYING

Anonymous authors
Paper under double-blind review

ABSTRACT

Memory systems are critical for role-playing agents (RPAs) to maintain long-horizon consistency. However, existing RPA memory methods (e.g., profiling) mainly rely on incremental summarization, whose compression discards details which become inaccessible to subsequent grounding. To address this issue, we propose a search-based memory framework called **BOOKMARKS** for **active grounding**, which retains access to the full preceding storyline and collects task-relevant information on demand. Since summarizing the preceding storyline anew for each grounding request incurs substantial redundant computation, BOOKMARKS introduces **passive updating** to reuse earlier search results as checkpoints. Each **bookmark** represents the **content** about a particular aspect (**section**) of story information at a specific synchronization **point**. For current task, BOOKMARKS searches for only useful contents, reuses existing semantically equivalent bookmarks or initializes new ones, and synchronizes the selected ones from their stored checkpoints to the current scene. A reused bookmark thus only needs to process the newly observed storyline suffix, avoiding repeated synchronization. We evaluate BOOKMARKS across six narrative artifacts involving 47 characters and 9,537 test cases against non-active grounding baselines, covering next-action prediction and challenging, human-authored reasoning questions from a mystery game. BOOKMARKS improves next-action fidelity in a five-dimensional evaluation (emotion, intent, causality, position, and content) and raises mystery-game reasoning accuracy from 39.44% to 45.42% over the strongest baseline. Further analysis shows that (1) active grounding amplifies the benefits of additional sampling; (2) improvements span all five evaluation dimensions; and (3) reuse preserves quality at lower cost, while larger bookmark budgets yield continued gains.[1]

1 INTRODUCTION

Role-playing agents (RPAs) (Chen et al., 2024b;c) require information about characters' evolving states and behaviors to remain faithful throughout a storyline. Existing memory methods either retrieve similar observations, limiting grounding to selected examples, or incrementally summarize observations into character profiles, leaving discarded details inaccessible. Introducing search-based grounding from general search agent (Jin et al., 2025) retains access to the full preceding storyline, allowing RPAs to propose search targets based on the current scene and actively collect relevant evidence from earlier observations. A naive implementation is costly as each search processes the storyline from the beginning. Inspired by readers who leave **bookmarks** to revisit useful information, we propose **BOOKMARKS**, a memory framework that supports active grounding for RPAs and further accelerates via reusable checkpoints.

Specifically, BOOKMARKS imitates human readers by maintaining a pool of bookmarks inserted at different positions in the storyline. Each bookmark contains (1) a section title q; (2) the maintained content y; and (3) a synchronization point p marking the latest processed storyline position. A bookmark (q, y, p) represents section-specific grounding information at a specific time point.

Based on this data structure, BOOKMARKS grounds RPAs in three steps. (1) **Proposal** identifies sections that are potentially beneficial for grounding the current scene; (2) **Synchronization** reuses

[1]The code and data will be released for reproduction.

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

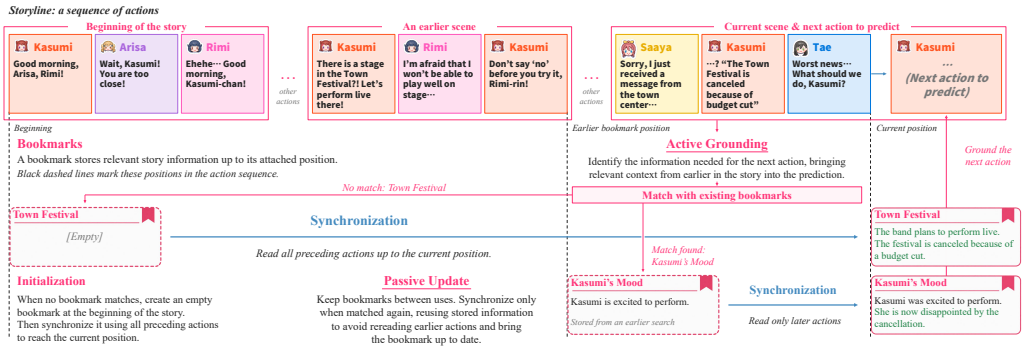

Figure 1: BOOKMARKS actively searches the preceding storyline for useful grounding information, while passive updating accelerates this search through checkpoint reuse.

an existing semantically equivalent bookmark or initializes an empty one at the beginning of the storyline, then updates its content from the stored checkpoint to the current scene; (3) **Grounding** uses the synchronized contents to support RPA acting. We present a running example of BOOKMARKS in Figure 1, with more details in Figure 2. In our implementation, each bookmark additionally has a type τ specifying how its content is represented and updated. Concept bookmarks maintain descriptions of entities or concepts. State bookmarks maintain ordered state nodes, refining the current node or appending a successor while retaining transition history. Behavioral bookmarks store actions together with their scenes and retrieve examples relevant to the current scene.

From a methodological view, conventional profile updating can be viewed as a special case of BOOKMARKS that maintains a single state bookmark through generic aggregation. In comparison, BOOKMARKS supports **active grounding** to seek information needed for the current scene. Its search cost is addressed by **passive updating**, which resumes selected bookmarks from their checkpoints and avoids repeatedly processing the same storyline prefix.

We evaluate BOOKMARKS on six diverse test artifacts drawn from an existing benchmark (Peng et al., 2026) and further extend the scope to cases made public after the evaluated model's (`gpt-5.1`) knowledge cutoff. We test each selected storyline from beginning to end with memory initialized empty, scoring next-action predictions by averaging binary match judgments across emotion, intent, causality, position, and content. We also evaluate accuracy on 284 challenging human-authored reasoning questions with definite answers from "Magical Girl Witch Trials", made public after the same cutoff. These additions extend previous evaluations to newly released narratives and authentic reasoning, testing grounding beyond familiar storylines.

BOOKMARKS improves predictions across all five dimensions and raises reasoning accuracy from 39.44% to 45.42% over the strongest baseline. The largest average gains in content and causality suggest that storyline evidence helps ground what characters do and why, while generally larger gains when evaluating the best of three predictions indicate that this evidence can support stronger candidates with additional sampling. Checkpoint reuse maintains prediction quality comparable to searching from scratch while avoiding repeated updates. Increasing the bookmark budget also yields further gains, whereas enlarging Extract-and-Aggregate (ETA) profiles gives diminishing returns, supporting memory organization around current information needs over longer summaries.

Our contributions include (1) BOOKMARKS improves role-playing fidelity through active grounding over the full preceding storyline. (2) Passive updating accelerates this grounding through bookmark reuse while preserving prediction quality. (3) Expanded benchmarks and extensive analyses validate these benefits on post-cutoff narratives and challenging human-authored reasoning questions.

2 BACKGROUND AND RELATED WORK

With the rapid development of LLMs' capability comes the ever-growing demand for more personalized interactions, to which role-playing agent (RPA) emerges as one of the central paradigms (Chen et al., 2024c; Tseng et al., 2024). These RPAs are expected to consistently produce in-character actions given story scene context, and the effective construction of such context is a *grounding* prob-

lem. As storylines lengthen, a static context would fail to provide enough relevant information, and thus dynamic *memory* systems become a key research direction. We accordingly cover **Evaluation**, **Training and Inference**, and **Memory** of role-playing agents.

Evaluation of RPAs splits by granularity into holistic judgment and per-action scoring. *Holistic judgment* aggregates over one of three character facets (identity, behavior, or knowledge) or bundles them. Identity is probed by psychiatric-style personality inventories (Wang et al., 2024b; Cheng et al., 2025), which rely on LLM-as-judge whose human alignment is known to be weak (Zhou et al., 2025). Behavior is tested in game, multi-agent, and social simulation testbeds (Yu et al., 2025; Zhou et al., 2024; Chen et al., 2024a). Knowledge is checked with factual or hallucination tests (Shen et al., 2023; Sadeq et al., 2024; Ahn et al., 2024). Broad-coverage suites bundle several facets into one composite score (Tu et al., 2024; Lu et al., 2025; He et al., 2025; Ding et al., 2025). *Per-action scoring*, in contrast, compares the next in-character action against a structured reference and is therefore facet-agnostic (Peng & Shang, 2025).

Training and Inference are two routes to adapt a model over time. The training-time route bakes character into parameters through supervised fine-tuning on character experiences (Shao et al., 2023) or large-scale synthetic dialogue (Moore Wang et al., 2024; Lu et al., 2024; Wang et al., 2025b; Yang et al., 2025a), multi-character LoRA hot-swapping (Yu et al., 2024), boundary- and personality-aware data (Tang et al., 2024; Yang et al., 2025b; Ji et al., 2025), and reinforcement-learning recipes (Wang et al., 2025e; Fang et al., 2025; Liu et al., 2025). These methods often suffer from plot scarcity, out-of-distribution hallucination, and an inability to absorb facts the storyline adds after training. The inference-time route leaves the backbone frozen and inserts structure between scene and response, like role-aware reasoning (Tang et al., 2025), strategy-conditioned dialogue (Ye et al., 2025), retrieval-augmented exemplars (Wang et al., 2024a), and activation-level persona steering (Chen et al., 2025a). Memory belongs to this same family but specifically deals with what content to store.

Memory for RPAs focuses on what is stored (a compressed profile vs. an explicit structure) and how it's updated (statically before inference vs dynamically as scenes arrive). *Static-profile methods* compress the storyline into a single profile re-attached every scene, in forms ranging from executable-function profiles (Peng & Shang, 2025) to dialogue-recursive and topic-indexed summaries (Wang et al., 2025a; Zhong et al., 2024; Lu et al., 2023). These methods guess what to keep and what to discard, and could lose critical cues that are useful later. *Static-structure methods* fix a storage scheme ahead of time, like typed memory hierarchies (Yan et al., 2023; Sun et al., 2024), event-and-relation graphs (Ran et al., 2025; Li et al., 2024; Wang et al., 2025d), and mined if-then decision trees with distilled discriminators (Peng et al., 2026). However, only a small, scene-dependent subset is needed at a time. *Dynamic-profile retrieval* picks relevant profile entries for each scene (Chen et al., 2025b; Huang et al., 2024; Wang et al., 2026), but memory pool is fixed. Outside RP, state maintenance, world-model grounding, and long-form-story reasoning (Yoneda et al., 2024; Liu et al., 2024; Gurung & Lapata, 2025; Yi et al., 2025; Xia et al., 2025) share one principle: keep enough state information to answer the next action-dependent question, and pull new information only when needed.

3 Our Bookmarks Framework

3.1 Preliminary

Storyline can be viewed as a sequence of actions from different characters, including special ones such as *"narration"* or *"environment"*. We denote the storyline as $A = [a_1, a_2, \cdots, a_N]$, where $N = |A|$. A character sequence $C = [c_1, c_2, \cdots, c_N]$ assigns each a_i to the character c_i who takes it.

Role-playing Agents RPAs aim to reproduce character behaviors by predicting the next action a_i from the current scene s_i and the target character c_i. In practice, s_i is a local context before a_i, such as several preceding actions. Thus, an RPA can be viewed as $a_i \sim \mathrm{RPA}(\cdot \mid s_i, c_i)$.

Grounding Stage Grounding augments the local scene with additional information before predicting a_i. This information may come from character profiles, retrieved examples, or storyline memory. We focus on data-driven grounding based on the preceding storyline $[a_1, \cdots, a_{i-1}]$.

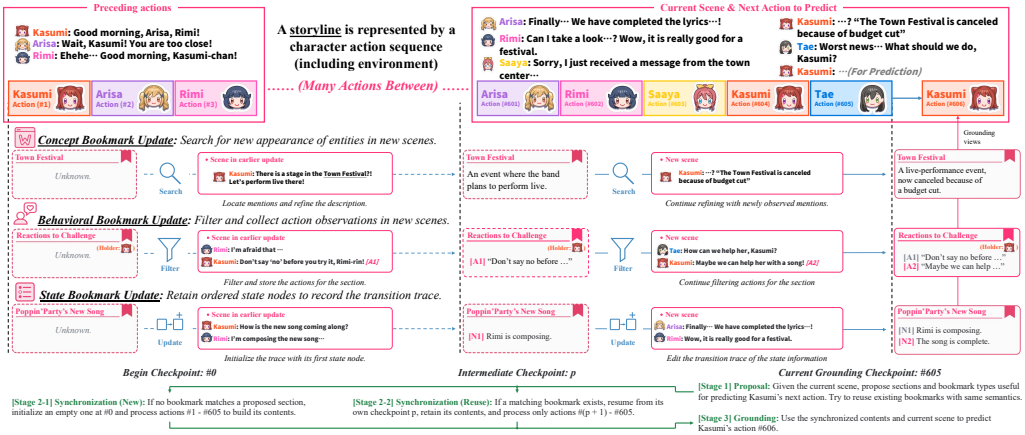

Figure 2: The role-playing grounding workflow of BOOKMARKS.

3.2 BOOKMARKS FRAMEWORK

Figure 2 illustrates the workflow of BOOKMARKS. Given a target character c_i and scene s_i, BOOKMARKS constructs a grounding context from a memory bank $\mathcal{B}$ of reusable *bookmarks*. Each bookmark maintains information under a *section*, such as a character's current goals or responses to danger, at a specific point in the storyline.

At each prediction step, BOOKMARKS proposes useful sections, matches them with existing bookmarks or initializes new ones, and synchronizes the selected bookmarks to the current story point. It then composes their contents into a grounding context g_i for prediction $a_i \sim \mathrm{RPA}(\cdot \mid s_i, c_i, g_i)$, which separates the choice of what to remember from information maintenance as the story unfolds.

Bookmark Data Structure A bookmark is represented as $b = (q, y, \tau, p)$, where q is a section title expressed as a noun phrase, y is its maintained content, τ is the bookmark type, and p is the synchronization point. The section identifies what is maintained, while its type determines how the content is represented.

We use three bookmark types, $\tau \in \{\texttt{concept}, \texttt{state}, \texttt{behavioral}\}$. **Concept** bookmarks maintain descriptions of named entities or concepts. **State** bookmarks maintain ordered traces of evolving states, such as locations, relationships, goals, or knowledge. **Behavioral** bookmarks store examples of a character's actions together with the scenes in which they occur. Each type thus preserves the information needed for grounding in a suitable form.

3.3 ACTIVE GROUNDING

The first stage selects what information would help ground the current prediction. Given (s_i, c_i), the proposal module generates a small set of sections and their types:

$$Q_i = \{(q_i^{(1)}, \tau_i^{(1)}), \cdots, (q_i^{(K)}, \tau_i^{(K)})\}.$$

Sections describe maintainable topics, such as *"Alice's current goals"* or *"Alice's responses to danger"*, so they can support later scenes as well as the current one.

This stage is active because the proposed sections depend on the current task. BOOKMARKS identifies the information relevant to the next action and uses it to select which parts of memory to access.

3.4 PASSIVE UPDATING

After proposal, BOOKMARKS resolves each section to a bookmark and synchronizes the selected bookmarks. This stage is passive because a bookmark is updated only when selected for grounding.

Matching For each proposed section, BOOKMARKS searches for a bookmark with the same type, subject, and section meaning. **Reuse** activates a matching bookmark with its contents and

synchronization point. **New** initializes an empty bookmark at $p = 0$ when no matching section is found. Each selected bookmark is then synchronized from its own checkpoint.

Synchronization An activated bookmark is synchronized from its stored point p to $i-1$ by processing the unseen suffix $[a_{p+1}, \cdots, a_{i-1}]$. A type-specific update operator U_τ produces

$$y' = U_\tau(q, y, [a_{p+1}, \cdots, a_{i-1}], C),$$

yielding the updated bookmark $(q, y', \tau, i-1)$. The section stays fixed during type-specific updates.

- **Concept.** Synchronization locates mentions of the entity or concept in the unseen suffix and uses their local contexts to refine its description:

  $$y' = \mathrm{Refine}\big(q, y, \{(s_t, a_t) : p < t < i,\ M(q, a_t) = 1\}\big),$$

  where $M(q, a_t)$ indicates whether a_t mentions the entity or concept specified by q. The updated description accumulates information about how it is characterized in the story.
- **State.** For section q, the content y records successive natural-language states:

  $$y = [y^{(1)}, \cdots, y^{(L)}], \qquad L = |y|.$$

  Each node includes its applicable time, place, or situation; the latest node $y^{(L)}$ describes the current state at p. The operator $U_{\texttt{state}}$ reads $[a_{p+1}, \cdots, a_{i-1}]$ in chronological chunks. Each chunk preserves the trace if it adds no relevant information, refines the latest node within the same conditions, or establishes a successor state under changed conditions:

  $$y \leftarrow \begin{cases} y, & \text{keep,} \\ [y^{(1)}, \cdots, y^{(L-1)}, \tilde{y}], & \text{modify current,} \\ [y^{(1)}, \cdots, y^{(L)}, \tilde{y}], & \text{append state.} \end{cases}$$

  Here $\tilde{y}$ is the state description written from q, the current trace, and the new chunk. An empty trace starts by appending its first supported state. Repeating this rule through $i-1$ yields y', with earlier nodes retained as transition history.
- **Behavioral.** Synchronization filters scene–action examples by their relevance to the section and adds the retained examples to the stored set y:

  $$y' = y \cup \mathrm{Filter}\big(q, \{(s_t, a_t) : p < t < i\}\big).$$

  Keeping each action with its scene preserves the conditions under which the behavior occurred. At grounding time, a selection operator retrieves stored examples relevant to the current scene:

  $$\mathcal{E}_i(b) = \mathrm{Select}(s_i, y') \subseteq y'.$$

  The grounding view is $\mathcal{E}_i(b)$, while y' remains available for future updates and retrieval.

Across all types, synchronization advances the bookmark's checkpoint so later updates process only the newly observed suffix. Combined with active proposal, this limits maintenance to the memory selected for the current prediction, reducing repeated storyline processing during active grounding.

Grounding Finally, BOOKMARKS constructs g_i from the synchronized bookmarks using a view suited to each type. Concept bookmarks contribute their descriptions. State bookmarks present their most recent nodes in chronological order through the current state, showing how the state evolved. Behavioral bookmarks contribute the selected scene–action examples $\mathcal{E}_i(b)$. Thus, the same behavioral section can provide different examples for different prediction contexts.

The same type-specific views are also applied to bookmarks with checkpoints near the current story point, providing recently maintained context. The resulting grounding context combines task-relevant concepts, evolving states, and situated behavioral examples to support the next action prediction.

4 BENCHMARK

Datasets We construct eight test sets from a previous benchmark (Peng et al., 2026) and artifacts released after `gpt-5.1`'s knowledge cutoff. The collection covers *The Melancholy of Haruhi*

Table 1: Examples of next-action prediction from Poppin'Party Band Story 0 (left) and human-authored multiple-choice questions from Magical Girl Witch Trials (right), with condensed context and English translation. The current scene during real benchmarking provides 10 preceding actions.

Next-Action Prediction	**Multiple-Choice Questions**
Context: Rimi had declined Kasumi's band invitation for fear of freezing onstage. When her sister's band was delayed, Kasumi and Arisa filled in, and Rimi joined them on bass.	**Context**: Each phone has a unique ID and can receive calls. Ema had called Miria. The recovered phone has Miria's distinctive case but no data; the handset inside could have been replaced.
Current scene (action sequence) **Kasumi:** A-ahaha, we got told off... **Arisa:** Of course we did! **Arisa:** And you dragged me into it! That was so embarrassing! **Rimi:** But thanks to you two, I could get onstage... I was scared, but I had fun.	**Current scene (action sequence)** **An-an:** That phone is Miria's! I never switched it! **Nanoka:** But— **An-an:** That's a baseless accusation! There is no evidence of a swap! **Ema:** The evidence that the phone was switched...
Question: What is Rimi's next action?	**Question:** Choose the correct continuation of Ema's statement.
Reference action **Rimi:** If it's not too late, I'd like to be in a band with you two...!	**A.** No longer exists. **B.** Can be found at the crime scene. **C.** Can be found by making a phone call. (**Answer:** C.)

Suzumiya (Haruhi), *JoJo's Bizarre Adventure* (JOJO), *Fullmetal Alchemist* (FMA), *A Game of Thrones* (AGOT), *Poppin'Party* (PoPiPa), and *Magical Girl Witch Trials* (Manosaba). These English and Japanese narratives span school comedy, supernatural adventure, political fantasy, music-centered daily life, and detective mystery. PoPiPa includes Band Story 0 and Event 321; the post-cutoff test cases include PoPiPa Event 321 and Manosaba. Table 1 illustrates the two evaluation tasks and Table 2 summarizes dataset statistics. Appendix E provides artifact and character backgrounds.

Next-Action Prediction We evaluate eligible character actions from beginning to end of each selected storyline, with memory initialized empty. For action a_i of character c_i, the local scene s_i contains up to 10 preceding actions, and grounding uses the observed prefix $[a_1, \cdots, a_{i-1}]$. The original action is held out as the reference. We compare predictions with references across five dimensions: **emotion** (expressed emotion), **intent** (immediate purpose), **causality** (character reasoning), **position** (stance), and **content** (central proposition or action). Each dimension receives an independent yes/no judgment, scored as $100/0$. We report the mean match score across the five dimensions and all evaluated actions. We use `gpt-4.1`, whose reliability is supported by the previous benchmark (Peng et al., 2026), as a judge distinct from all evaluated models. Our human validation of 200 cases per dimension yields a mean agreement of 93.7% with the judge, with a standard deviation of 1.36 percentage points across dimensions.

Multiple-Choice Questions We further include 284 multiple-choice questions (MCQs) from Manosaba. These human-authored questions come from the original game, have definite correct answers, and require reasoning over accumulated story evidence. The questions became public in July 2025 after `gpt-5.1`'s knowledge cutoff, providing an additional test of reasoning on newly released material. The model selects an answer from the provided options, and we report accuracy against the original correct answers.

5 Experiment

5.1 Baselines

We compare BOOKMARKS with representative grounding methods that differ in how they access and maintain storyline information.

- **Vanilla** predicts character actions without additional grounding context, relying only on the parametric knowledge of the RPA.
- **Retrieval-based In-Context Learning** (Wang et al., 2024a) **(RICL)** retrieves top-k similar past scene-action pairs as grounding examples for the current scene.
- **Extract-and-Aggregate (ETA)** (Wang et al., 2025c) incrementally maintains a single character profile by summarizing new observations and aggregating them into the previous profile.
- **3-way ETA** divides the character profile into three fixed sections, *experience*, *state*, and *behavior*. For each prediction, the RPA selects relevant sections and synchronizes only those sections through ETA-style recurrent updating.

Table 2: NAP reports the mean match score across five dimensions; MCQ reports accuracy (both in %). Best@k selects the best of k predictions per action. MCQ uses Best@1 to avoid inflated accuracy from multiple guesses. Significance testing yields $p < 0.05$.

	Artifact	Haruhi	JOJO	FMA	AGOT	PoPiPa		Manosaba	
						BS0	E321	NAP	MCQ
Info	Genre	Sci-fi	Action	Fantasy	Politics	Music		Mystery	
	#Character	5	7	5	11	5		14	
	#Action	781	1592	1351	1686	1944	306	1593	284
	Released	Before 20-01					26-02	25-07	
	Lang	en				ja			
Best@1	Vanilla	32.21	34.10	34.21	41.30	36.05	44.57	38.10	35.21
	RICL	32.86	35.52	35.28	42.28	37.50	46.82	39.82	36.62
	ETA	33.47	35.93	35.72	42.55	38.65	46.24	40.22	38.38
	3-way ETA	34.17	36.44	36.47	42.85	38.25	47.03	40.84	39.44
	BOOKMARKS	**36.37**	**38.87**	**37.88**	**44.41**	**39.87**	**49.35**	**43.97**	**45.42**
Best@3	Vanilla	40.69	45.57	42.98	50.31	39.17	48.27	41.83	Disabled by MCQ
	RICL	41.74	47.83	44.71	53.75	46.22	53.21	45.98	
	ETA	42.16	49.06	45.38	53.26	44.82	54.70	46.65	
	3-way ETA	42.52	49.85	46.84	54.82	45.89	55.63	48.19	
	BOOKMARKS	**45.04**	**52.21**	**49.18**	**56.84**	**48.89**	**59.23**	**52.47**	

Methodological Comparison ETA can be viewed as a special case of BOOKMARKS since its character profile corresponds to a single state bookmark maintained through generic aggregation. 3-way ETA forms an intermediate setting by introducing selection among three fixed sections while retaining ETA-style updates. BOOKMARKS further allows sections to be proposed as needed and their contents to be maintained through type-specific representations and update rules. This progression lets us evaluate how flexible memory organization affect grounding quality.

Implementation We primarily use `gpt-5.1` as the RPA backbone and additionally evaluate open-source DeepSeek and Qwen models to assess generalizability across model families. RICL retrieves eight examples for ICL by scene semantic similarity using `qwen3-embedding-8b`. To match BOOKMARKS's additional output-token usage per synchronization round, ETA and 3-way ETA are prompted to limit their outputs to 1,024 tokens and 256 tokens per section, respectively. BOOKMARKS proposes at most three bookmarks per round, selects eight behavioral action examples using BM25, and updates state bookmarks in blocks of 64 actions. Prompts are in Appendix B.

5.2 MAIN RESULTS

Table 2 shows that BOOKMARKS achieves the best performance across diverse storylines and both evaluation tasks, supporting active grounding that maintains storyline memory around the needs of the current scene. All methods' gains over Vanilla highlight the value of explicit storyline evidence, while the advantage over RICL supports maintaining synchronized memory beyond retrieving individual examples. The general improvement from ETA to 3-way ETA suggests that selecting and updating relevant memory sections already benefits grounding. BOOKMARKS further improves on both by allowing flexible sections with type-specific representations and updates. Its generally larger gains under Best@3 suggest that this targeted grounding also helps the model produce stronger candidate actions when multiple predictions are sampled. These gains extend to post-cutoff narratives, supporting effective grounding on uncontaminated material, a setting rarely tested in prior work. The particularly large improvement on human-authored Manosaba MCQs further suggests that active grounding is especially suited to reasoning, where each question creates an immediate need to seek specific evidence from the observed storyline, which is further validated by cases in Appendix C.

Multi-dimension Results Figure 3 shows that BOOKMARKS achieves prominent gains over all five dimensions, yielding broad improvements in role-playing fidelity. For `gpt-5.1`, the largest average gains across storylines occur in *Content* and *Causality*. This pattern aligns with the goal of BOOKMARKS since the framework provides relevant storyline evidence to ground what a character does or says and the reasoning behind it. Maintaining synchronized concepts, evolving states,

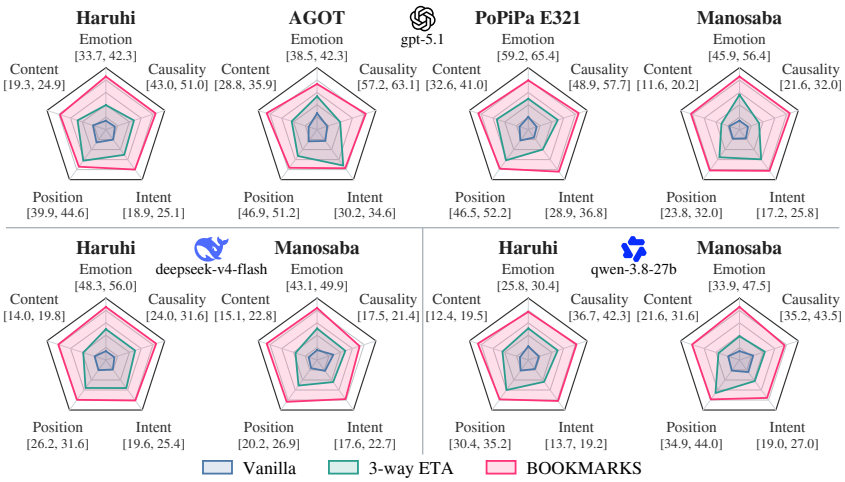

Figure 3: NAP match scores under Best@1. Bracketed intervals at each vertex give the axis limits (%) at the radar center and outer boundary, respectively, with linear scaling between them.

and situated behavioral examples supplies context that connects the next action to the character's preceding experiences and current circumstances.

Cross-Model Performance The benefits also transfer to the open-source `deepseek` and `qwen` backbones as both show substantial improvements spanning all five dimensions. This broader pattern suggests that synchronized storyline memory supports multiple aspects of character behavior for backbones with more limited capabilities. Notably, `qwen-3.8-27b` benefits consistently from BOOKMARKS on both storylines, demonstrating that even a relatively small 27B-parameter model can make effective use of the framework.

5.3 EFFICIENCY ANALYSIS

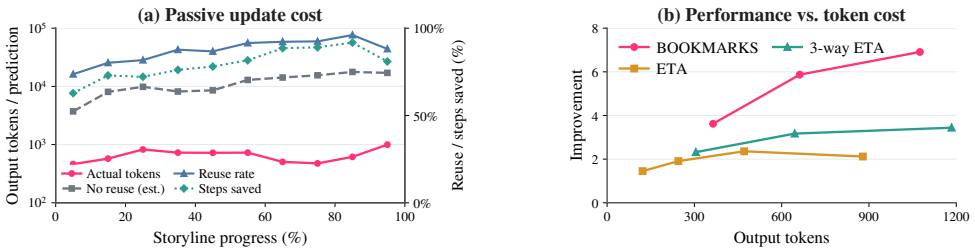

Figure 4: Efficiency analysis on Manosaba[3]. (a) Actual and estimated no-reuse output-token costs, with reuse rates and fractions of update steps saved on the right axis, averaged over 10% storyline-progress buckets. (b) Best@1 NAP improvement over Vanilla versus output tokens.

BOOKMARKS reduces the cost of maintaining storyline knowledge while continuing to benefit from a larger grounding budget. Figure 4 (left) shows that, as the storyline grows, reusing bookmarks avoids most repeated updates and keeps output-token cost low even as the estimated cost without reuse increases. Across progress buckets, the estimated output-token cost without reuse is 8.1–32.6× the actual cost, highlighting the savings from keeping reusable memories synchronized throughout a long narrative. To examine how effectively each method uses additional capacity, we vary the bookmark limit in BOOKMARKS over $\{2, 3, 5\}$ and the maximum profile-token capacity specified in ETA's prompt over $\{128, 256, 512, 1024\}$ (3-way for $\{128, 256, 512\}$ each section). Figure 4 (right) shows that allowing more bookmarks consistently improves predictions, whereas increasing (3-way)

[3]We use Manosaba for analysis because its characters were unseen before the evaluated models' knowledge cutoffs, providing a clean setting without prior character knowledge.

ETA's profile capacity yields diminishing gains and eventually a slight decline. This pattern supports organizing grounding around multiple task-relevant memories as an effective way to obtain both better predictions at lower cost and further gains when more resources are available.

5.4 Ablation Study

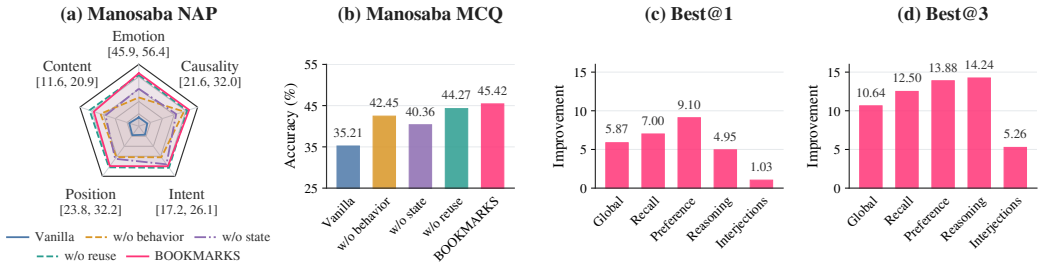

Figure 5: Manosaba ablations on (a) Best@1 NAP, with axes linearly rescaled to the labeled ranges, and (b) MCQ accuracy. (c,d) NAP improvements of BOOKMARKS over Vanilla (percentage points), overall and by action type, under Best@1 and Best@3, respectively.

The Manosaba ablations (1) evaluate the contributions of state and behavioral bookmarks to grounding and (2) test whether reuse accelerates memory updates without compromising prediction quality. As shown in Figure 5(a,b), removing either memory type lowers performance across all five NAP dimensions and on MCQ. The NAP losses reflect their different roles, as removing behavioral bookmarks hurts *Emotion* and *Intent* more, whereas removing state bookmarks has a larger effect on *Causality* and *Content*. This pattern is consistent with behavioral examples grounding how a character responds and state traces preserving the evolving circumstances behind an action. On MCQ, removing state and behavioral bookmarks reduces accuracy by 5.06 and 2.97 percentage points, respectively, further supporting the value of both memory types for reasoning over storyline evidence. Disabling reuse yields performance comparable to the full framework on both NAP and MCQ. Combined with the cost savings in Figure 4, this supports reuse as an efficiency mechanism that preserves grounding quality, while the specialized memories supply the complementary evidence needed for stronger predictions.

5.5 Improvement Analysis

Figure 5(c,d) further divides improvement gain by action type, which shows that BOOKMARKS's gains are concentrated in actions that depend on storyline knowledge. *Recall* and *Preference* both improve more than the global average under Best@1 and Best@3, highlighting the contribution of remembered events and character-specific preferences. Trivial *Interjections* benefit much less, as they typically require little storyline evidence. The substantially larger gains on *Reasoning* under Best@3 suggest that Best@1 underestimates BOOKMARKS's contribution as a memory system, since performance also depends on the model's ability to derive the final conclusion. Best@3 better reveals the value of the reasoning-relevant information BOOKMARKS collects, supporting targeted access to storyline evidence as the main source of its gains.

6 Conclusion and Future Work

We present BOOKMARKS, a search-based memory framework that improves role-playing fidelity through active grounding and accelerates this process through passive updating. We show that BOOKMARKS outperforms retrieval-based grounding and incremental profiling on next-action prediction and storyline reasoning. As future work, we plan to extend BOOKMARKS to character-specific knowledge tracking, integrate it with self-refinement, and develop update policies tailored to different bookmark types and narrative structures.

AI USE STATEMENT

We used large language models (LLMs) only for copy-editing: improving grammar, clarity, and overall writing style to enhance readability. The LLMs did not contribute to or modify the scientific content, core ideas, methodology, or experimental results. The authors take full responsibility for the manuscript's final content and accuracy.

REFERENCES

Jaewoo Ahn, Taehyun Lee, Junyoung Lim, Jin-Hwa Kim, Sangdoo Yun, Hwaran Lee, and Gunhee Kim. Timechara: Evaluating point-in-time character hallucination of role-playing large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 3291–3325, 2024.

Hongzhan Chen, Hehong Chen, Ming Yan, Wenshen Xu, Gao Xing, Weizhou Shen, Xiaojun Quan, Chenliang Li, Ji Zhang, and Fei Huang. Socialbench: Sociality evaluation of role-playing conversational agents. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 2108–2126, 2024a.

Jiangjie Chen, Xintao Wang, Rui Xu, Siyu Yuan, Yikai Zhang, Wei Shi, Jian Xie, Shuang Li, Ruihan Yang, Tinghui Zhu, Aili Chen, Nianqi Li, Lida Chen, Caiyu Hu, Siye Wu, Scott Ren, Ziquan Fu, and Yanghua Xiao. From persona to personalization: A survey on role-playing language agents, 2024b. URL https://arxiv.org/abs/2404.18231.

Jiangjie Chen, Xintao Wang, Rui Xu, Siyu Yuan, Yikai Zhang, Wei Shi, Jian Xie, Shuang Li, Ruihan Yang, Tinghui Zhu, et al. From persona to personalization: A survey on role-playing language agents. *Transactions on Machine Learning Research*, 2024c.

Runjin Chen, Andy Arditi, Henry Sleight, Owain Evans, and Jack Lindsey. Persona vectors: Monitoring and controlling character traits in language models. *arXiv preprint arXiv:2507.21509*, 2025a.

Weishu Chen, Jinyi Tang, Zhouhui Hou, Shihao Han, Mingjie Zhan, Zhiyuan Huang, Delong Liu, Jiawei Guo, Zhicheng Zhao, and Fei Su. Moom: Maintenance, organization and optimization of memory in ultra-long role-playing dialogues. *arXiv preprint arXiv:2509.11860*, 2025b.

Xilong Cheng, Yunxiao Qin, Yuting Tan, Zhengnan Li, Ye Wang, Hongjiang Xiao, and Yuan Zhang. Psymem: Fine-grained psychological alignment and explicit memory control for advanced role-playing llms. *arXiv preprint arXiv:2505.12814*, 2025.

Hang Ding, Qiming Feng, Dongqi Liu, Qi Zhao, Tao Yao, Shuo Wang, Dongsheng Chen, Jian Li, Zhenye Gan, Jiangning Zhang, et al. Rolermbench & rolerm: Towards reward modeling for profile-based role play in dialogue systems. *arXiv preprint arXiv:2512.10575*, 2025.

Feiteng Fang, Ting-En Lin, Yuchuan Wu, Xiong Liu, Xiang Huang, Dingwei Chen, Jing Ye, Haonan Zhang, Liang Zhu, Hamid Alinejad-Rokny, et al. Charm: Character-based act-adaptive reward modeling for advanced role-playing language agents. *arXiv e-prints*, pp. arXiv–2505, 2025.

Alexander Gurung and Mirella Lapata. Learning to reason for long-form story generation. *CoRR*, abs/2503.22828, 2025. doi: 10.48550/ARXIV.2503.22828. URL https://doi.org/10.48550/arXiv.2503.22828.

Kai He, Yucheng Huang, Wenqing Wang, Delong Ran, Dongming Sheng, Junxuan Huang, Qika Lin, Jiaxing Xu, Wenqiang Liu, and Mengling Feng. Crab: A novel configurable role-playing llm with assessing benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15030–15052, 2025.

Le Huang, Hengzhi Lan, Zijun Sun, Chuan Shi, and Ting Bai. Emotional rag: Enhancing role-playing agents through emotional retrieval. In *2024 IEEE International Conference on Knowledge Graph (ICKG)*, pp. 120–127. IEEE, 2024.

Ke Ji, Yixin Lian, Linxu Li, Jingsheng Gao, Weiyuan Li, and Bin Dai. Enhancing persona consistency for llms' role-playing using persona-aware contrastive learning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 26221–26238, 2025.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan O Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. In *Second Conference on Language Modeling*, 2025.

Shilong Li, Yancheng He, Hangyu Guo, Xingyuan Bu, Ge Bai, Jie Liu, Jiaheng Liu, Xingwei Qu, Yangguang Li, Wanli Ouyang, et al. Graphreader: Building graph-based agent to enhance long-context abilities of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 12758–12786, 2024.

Cheng Liu, Yifei Lu, Fanghua Ye, Jian Li, Xingyu Chen, Feiliang Ren, Zhaopeng Tu, and Xiaolong Li. Cogdual: Enhancing dual cognition of llms via reinforcement learning with implicit rule-based rewards. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 27295–27324, 2025.

Zeyang Liu, Xinrui Yang, Shiguang Sun, Long Qian, Lipeng Wan, Xingyu Chen, and Xuguang Lan. Grounded answers for multi-agent decision-making problem through generative world model. *Advances in Neural Information Processing Systems*, 37:46622–46652, 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.

Junru Lu, Jiazheng Li, Guodong Shen, Lin Gui, Siyu An, Yulan He, Di Yin, and Xing Sun. Rolemrc: A fine-grained composite benchmark for role-playing and instruction-following. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 21008–21030, 2025.

Keming Lu, Bowen Yu, Chang Zhou, and Jingren Zhou. Large language models are superpositions of all characters: Attaining arbitrary role-play via self-alignment. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7828–7840, 2024.

Zekun Moore Wang, Zhongyuan Peng, Haoran Que, Jiaheng Liu, Wangchunshu Zhou, Yuhan Wu, Hongcheng Guo, Ruitong Gan, Zehao Ni, Jian Yang, et al. Rolellm: benchmarking, eliciting, and enhancing role-playing abilities of large language models. *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 14743–14777, 2024.

Letian Peng and Jingbo Shang. Codifying character logic in role-playing. In *Proceedings of the Thirty-Ninth Annual Conference on Neural Information Processing Systems (NeurIPS 2025) Poster Session*, 2025. URL https://neurips.cc/virtual/2025/loc/san-diego/poster/117989. Poster presentation, NeurIPS 2025.

Letian Peng, Kun Zhou, Longfei Yun, Yupeng Hou, and Jingbo Shang. Deriving character logic from storyline as codified decision trees. *arXiv preprint arXiv:2601.10080*, 2026.

Yiting Ran, Xintao Wang, Tian Qiu, Jiaqing Liang, Yanghua Xiao, and Deqing Yang. BOOKWORLD: from novels to interactive agent societies for story creation. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pp. 15898–15912. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.acl-long.773/.

Nafis Sadeq, Zhouhang Xie, Byungkyu Kang, Prarit Lamba, Xiang Gao, and Julian McAuley. Mitigating hallucination in fictional character role-play. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 14467–14479, 2024.

Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. Character-llm: A trainable agent for role-playing. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 13153–13187, 2023.

Tianhao Shen, Sun Li, Quan Tu, and Deyi Xiong. Roleeval: A bilingual role evaluation benchmark for large language models. *arXiv preprint arXiv:2312.16132*, 2023.

Libo Sun, Siyuan Wang, Xuanjing Huang, and Zhongyu Wei. Identity-driven hierarchical role-playing agents. *arXiv preprint arXiv:2407.19412*, 2024.

Yihong Tang, Jiao Ou, Che Liu, Fuzheng Zhang, Di Zhang, and Kun Gai. Erabal: Enhancing role-playing agents through boundary-aware learning. *arXiv preprint arXiv:2409.14710*, 2024.

Yihong Tang, Kehai Chen, Muyun Yang, Zhengyu Niu, Jing Li, Tiejun Zhao, and Min Zhang. Thinking in character: Advancing role-playing agents with role-aware reasoning. *arXiv preprint arXiv:2506.01748*, 2025.

Yu-Min Tseng, Yu-Chao Huang, Teng-Yun Hsiao, Wei-Lin Chen, Chao-Wei Huang, Yu Meng, and Yun-Nung Chen. Two tales of persona in llms: A survey of role-playing and personalization. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 16612–16631, 2024.

Quan Tu, Shilong Fan, Zihang Tian, Tianhao Shen, Shuo Shang, Xin Gao, and Rui Yan. Charactereval: A chinese benchmark for role-playing conversational agent evaluation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11836–11850, 2024.

Kai Wang, Haoyang You, Yang Zhang, and Zhongjie Wang. Memory-driven role-playing: Evaluation and enhancement of persona knowledge utilization in llms. *arXiv preprint arXiv:2603.19313*, 2026.

Liang Wang, Nan Yang, and Furu Wei. Learning to retrieve in-context examples for large language models. In Yvette Graham and Matthew Purver (eds.), *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1752–1767, St. Julian's, Malta, March 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.eacl-long.105. URL https://aclanthology.org/2024.eacl-long.105/.

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, 2025a.

Xiaoyang Wang, Hongming Zhang, Tao Ge, Wenhao Yu, Dian Yu, and Dong Yu. Opencharacter: Training customizable role-playing llms with large-scale synthetic personas. *arXiv preprint arXiv:2501.15427*, 2025b.

Xintao Wang, Yunze Xiao, Jen-tse Huang, Siyu Yuan, Rui Xu, Haoran Guo, Quan Tu, Yaying Fei, Ziang Leng, Wei Wang, Jiangjie Chen, Cheng Li, and Yanghua Xiao. Incharacter: Evaluating personality fidelity in role-playing agents through psychological interviews. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 1840–1873. Association for Computational Linguistics, 2024b. doi: 10.18653/V1/2024.ACL-LONG.102. URL https://doi.org/10.18653/v1/2024.acl-long.102.

Xintao Wang, Heng Wang, Yifei Zhang, Xinfeng Yuan, Rui Xu, Jen-tse Huang, Siyu Yuan, Haoran Guo, Jiangjie Chen, Wei Wang, Yanghua Xiao, and Shuchang Zhou. Coser: Coordinating llm-based persona simulation of established roles. *CoRR*, abs/2502.09082, 2025c. doi: 10.48550/ARXIV.2502.09082. URL https://doi.org/10.48550/arXiv.2502.09082.

Yongjie Wang, Jonathan Leung, and Zhiqi Shen. Rolerag: Enhancing llm role-playing via graph guided retrieval. *arXiv preprint arXiv:2505.18541*, 2025d.

Zongsheng Wang, Kaili Sun, Bowen Wu, Qun Yu, Ying Li, and Baoxun Wang. Raiden-r1: Improving role-awareness of llms via grpo with verifiable reward. *arXiv preprint arXiv:2505.10218*, 2025e.

Haotian Xia, Hao Peng, Yunjia Qi, Bin Xu, Juanzi Li, Hou Lei, and Xiaozhi Wang. Storywriter: A multi-agent framework for long story generation. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pp. 6559–6563, 2025.

Ming Yan, Ruihao Li, Hao Zhang, Hao Wang, Zhilan Yang, and Ji Yan. Larp: Language-agent role play for open-world games. *arXiv preprint arXiv:2312.17653*, 2023.

Bohao Yang, Dong Liu, Chenghao Xiao, Kun Zhao, Chen Tang, Chao Li, Lin Yuan, Yang Guang, and Chenghua Lin. Crafting customisable characters with LLMs: A persona-driven role-playing agent framework. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 20216–20240, Suzhou, China, November 2025a. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.1100. URL https://aclanthology.org/2025.findings-emnlp.1100/.

Tao Yang, Yuhua Zhu, Xiaojun Quan, Cong Liu, and Qifan Wang. Psyplay: Personality-infused role-playing conversational agents. *arXiv preprint arXiv:2502.03821*, 2025b.

Jing Ye, Lu Xiang, Yaping Zhang, and Chengqing Zong. Sweetiechat: A strategy-enhanced role-playing framework for diverse scenarios handling emotional support agent. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert (eds.), *Proceedings of the 31st International Conference on Computational Linguistics, COLING 2025, Abu Dhabi, UAE, January 19-24, 2025*, pp. 4646–4669. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.coling-main.312/.

Qiang Yi, Yangfan He, Jianhui Wang, Xinyuan Song, Shiyao Qian, Xinhang Yuan, Yi Xin, Yijin Wang, Jingqun Tang, Yuchen Li, et al. Score: Story coherence and retrieval enhancement for ai narratives. *arXiv preprint arXiv:2503.23512*, 2025.

Takuma Yoneda, Jiading Fang, Peng Li, Huanyu Zhang, Tianchong Jiang, Shengjie Lin, Ben Picker, David Yunis, Hongyuan Mei, and Matthew R Walter. Statler: State-maintaining language models for embodied reasoning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 15083–15091. IEEE, 2024.

Pengfei Yu, Dongming Shen, Silin Meng, Jaewon Lee, Weisu Yin, Andrea Yaoyun Cui, Zhenlin Xu, Yi Zhu, Xingjian Shi, Mu Li, and Alex Smola. RPGBENCH: evaluating large language models as role-playing game engines. *CoRR*, abs/2502.00595, 2025. doi: 10.48550/ARXIV.2502.00595. URL https://doi.org/10.48550/arXiv.2502.00595.

Xiaoyan Yu, Tongxu Luo, Yifan Wei, Fangyu Lei, Yiming Huang, Hao Peng, and Liehuang Zhu. Neeko: Leveraging dynamic lora for efficient multi-character role-playing agent. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 12540–12557, 2024.

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.

Lingfeng Zhou, Jialing Zhang, Jin Gao, Mohan Jiang, and Dequan Wang. Personaeval: Are llm evaluators human enough to judge role-play? In *Second Conference on Language Modeling*, 2025.

Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, et al. Sotopia: Interactive evaluation for social intelligence in language agents. In *The Twelfth International Conference on Learning Representations*, 2024.

A LIMITATIONS

This work focuses on establishing BOOKMARKS as a general search-based memory framework for long-horizon role-playing, while leaving several extensions for future exploration. First, the current framework mainly maintains storyline-level states, behaviors, and concepts, and can be further extended to finer-grained recognition management, such as explicitly tracking which character knows which information at each story point. Second, BOOKMARKS is currently used as a grounding module before generation, while future work may integrate it with self-refinement so that agents can revise generated actions when bookmark evidence reveals inconsistencies. Finally, although we design type-specific update procedures for concept, state, and behavioral bookmarks, more customized update policies could be developed for different query types, characters, and narrative structures.

B PROMPTS

Tables 3–7 present the prompts for bookmark construction and updating, next-action prediction, and evaluation. Braced fields denote runtime inputs. Each evaluation dimension receives an independent yes/no judgment, scored as $100/0$.

Table 3: Bookmark proposal, matching, and concept-update prompts.

Section proposal

Scene
{scene}

It is now {target_character}'s turn to make the next utterance. Before deciding it, propose up to {max_k} bookmarks that may bring useful information from the preceding story into the current context.

A concept bookmark names a concrete concept or entity and searches that name directly. A state bookmark keeps a transition trace of natural-language state nodes for a character, relationship, object, place, plan, or shared environment.

Bookmarks already visible in the scene
{visible_catalog}

Other bookmark history
{nonvisible_history}

Characters that may own a bookmark
{character_options}

Use short noun-phrase section names. Write every "section" value in the same language as the input scene. A bookmark may concern another character when that helps with {target_character}'s continuation; use "env" only for shared or environmental state. Avoid repeating information already supplied by the visible scene or its bookmarks. Return fewer than {max_k}, or an empty list, when there are fewer useful searches.

Return only a JSON array. Each item must contain "section", "tag", and "character", and the tag must be "concept", "state", or "behavioral".
Concept example: {"section": "Project Polaris", "tag": "concept", "character": "{target_character}"}
State example: {"section": "Band members' current location", "tag": "state", "character": "env"}
Behavioral example: {"section": "{target_character}'s response pattern when being forcefully invited", "tag": "behavioral", "character": "{target_character}"}

Bookmark matching

You are choosing whether one existing bookmark can be reused for a new bookmark.

New bookmark section:
{new_section}

Candidate existing bookmark sections:
{candidate_catalog}

Choose one candidate only when its section is semantically equivalent to the new section and both sections maintain the same bookmark slot over time. A section that is merely related, partially overlapping, broader, narrower, or otherwise useful must not be reused.

Output only the exact bookmark number of the single best reuse candidate, or none if no candidate has the same section meaning.

Concept update

The bookmark section is "{section}". Here are the places where that name appears:
{mention_context}

The current answer is:
{answer}

Write a short, self-contained memory note that names "{section}" and explains it using the information above, keeping useful earlier details when they still apply. Return only the note.

Table 4: State-transition check and state-node writing prompts.

State-transition check

The state bookmark "{section}" belongs to {state_character}.
Recent state-transition trace
{recent_trace}
{empty_trace_note}
{background_note}
Current story chunk — the only transition evidence
{chunk}
{direct_action_note}
Each state node is one natural-language state note. Its applicability conditions—such as supported time, place, event phase, relationship phase, plan phase, or other situational context—are written directly inside that same note. There is no separate precondition field.
Choose exactly one operation:
none
- The current chunk gives no new relevant evidence, merely repeats the current state, or only the context-only background is relevant.
modify_current
- The current chunk refines, corrects, narrows, or adds information to the latest state under the same applicability conditions already written in that state.
- Rewrite the current node rather than adding a transition.
- This operation is invalid when the trace is empty.
append_state
- The current chunk establishes a successor state whose meaningful applicability conditions differ from the current node—for example a new time, place, event phase, relationship phase, plan phase, or situation.
- Use this for the first established state when the trace is empty.
- Do not append merely because story time passed or the location changed; the chunk must establish or change this bookmark's state under those new conditions.
For a character-owned bookmark, choose none unless the current chunk contains information explicitly about that character or unambiguously referring to them; never transfer another character's state. Here "env" means shared or environmental state.
Return exactly one token: none, modify_current, or append_state.

State-node writing

The state bookmark "{section}" belongs to {state_character}.
Recent state-transition trace
{recent_trace}
Applied operation
{operation}
{background_note}
Current story chunk — the only transition evidence
{chunk}
{direct_action_note}
{operation_instruction}
The returned state must be short and self-contained. Write its applicability conditions directly inside the state sentence itself: naturally state when, where, and under what event, interaction, investigation, relationship, plan, or other situational phase it applies whenever those conditions are supported. Conditions must be part of the state text, not separate fields, labels, JSON, or metadata. Omit unsupported time or place rather than inventing them, but always make the relevant situation or phase clear enough for a later model to decide whether to modify this node or append a successor.
Use the context-only background only to resolve references. Do not copy facts from it as new state evidence. Here "env" means shared or environmental state.
Return only the natural-language state note.

Instruction for modify_current

Rewrite the latest state node as one complete natural-language state note. Preserve earlier information that remains valid, and keep it within the same applicability conditions. Do not move it into a distinct time, place, phase, or situation; that would require append_state.

Instruction for append_state

Write one new natural-language state note that succeeds the existing trace. Do not rewrite an earlier node. If the previous state ended but no replacement is known, explicitly state that under the new conditions instead of inventing a value.

Table 5: Behavioral annotation and filtering prompts.

Scene–action annotation

Scene
{scene}
Action
{action}
In one short sentence, explain what in the preceding scene caused or prompted this action. Use any concept or state bookmark annotations only when needed to make that causal link clear. This explanation is independent of every behavioral bookmark section and target, so do not judge whether the action belongs to any behavioral pattern.
Return only that sentence in the same language as the action.

Behavioral filtering

The behavioral bookmark "{section}" collects {character}'s actions that fit that section.
{candidate_text}
Each action is followed by a short parenthetical ※ annotation explaining what in the preceding scene caused or prompted it. The annotation is independent of this behavioral bookmark. For each candidate, answer "yes" if the annotated action belongs in this bookmark and "no" otherwise. This step is only collecting matching actions; one action does not need to prove the overall pattern by itself.
Return only a JSON list of {n_candidates} "yes" or "no" strings in candidate order.

Table 6: Next-action prediction prompts, shown for spoken utterances.

Single-action prediction

Background Summary
{grounding}
– –
Current Scene
{scene}
– –
Predict the next utterance spoken by {character} in the following scene. The "action" field must contain a concise spoken line by {character}, formatted as "{character}: ...", not narration, a nonverbal action, or another character's line.
Output in the following format:
```python
{
"reasoning": "...",
"action": "...",
}
```

Multiple-candidate prediction

{grounding}
Based on the features, infer {n_candidate} diverse potential next actions (A concise sentence, {character}: ...) of {character} in the following scene:
{scene}
Output in the following format (same language as the given scene):
Reasoning
...
Actions
1. ...
2. ...
...

Table 7: NAP evaluation prompt and dimension-specific criteria.

Common evaluation template

Scene
{scene}
Your Response: {prediction}
Ground Truth: {action}
{dimension_instruction}
Output only one lowercase word: yes or no.

Emotion

Decide whether the emotion expressed by {character} in "Your Response" matches the emotion expressed by {character} in "Ground Truth".

Judge only the character's dominant expressed emotion, such as fear, anger, surprise, joy, sadness, relief, embarrassment, or calmness. Minor differences in intensity or wording do not matter. Ignore intent, causality, position, and semantic content.

If neither text expresses a clear emotion, treat them as matching. If only one text expresses a clear emotion, treat them as not matching.

Intent

Decide whether the immediate intent of {character} in "Your Response" matches the immediate intent of {character} in "Ground Truth".

Intent means what the character is trying to accomplish through the utterance or action, such as obtaining information, informing someone, refusing, persuading, comforting, threatening, escaping, investigating, or ending an interaction. Judge the immediate purpose rather than the exact wording or information conveyed.

Ignore emotion, causality, position, and semantic content.

Causality

Decide whether the character logic behind {character}'s utterance or action in "Your Response" matches the character logic behind it in "Ground Truth".

Character logic means the underlying reasoning that connects the character's understanding of the situation, beliefs, priorities, and disposition to the utterance or action. The exact immediate trigger does not need to be identical. Treat the two texts as matching when they follow the same underlying character reasoning, and as not matching when they rely on different or incompatible character reasoning.

Do not require emotion, intent, position, semantic content, or wording to match by themselves. Judge only whether the character logic is the same.

Position

Decide whether the position (立場) expressed by {character} in "Your Response" matches the position (立場) expressed by {character} in "Ground Truth".

Position (立場) means the character's own position in the situation: where the character stands and how the character aligns themself with what is happening, such as supporting or opposing, accepting or resisting, committed or hesitant, proactive or passive, serious or playful, or neutral. A position does not require a specific target or object. Judge the character's own 立場, not merely an attitude toward another person.

Ignore emotion, intent, causality, and semantic content. If neither text expresses a marked position, treat both as neutral and matching.

Content

Decide whether the main content of {character}'s utterance or action in "Your Response" matches the main content in "Ground Truth".

Judge only the single most central proposition, question, answer, decision, or concrete action in each text. If both texts communicate or perform the same main point, treat them as matching even when secondary content, supporting details, modifiers, examples, or extra sub-actions differ. If multiple contents appear, compare the most central one in each text.

Ignore emotion, intent, causality, position, and wording style. Treat them as not matching only when the main content itself differs.

C CASE STUDIES

In the Manosaba cases in Table 8, active grounding recovers earlier observations that are decisive for the current question. The proposed sections bring back the spray-paint smell in one case and distinguish an oral wound from fingertip bloodstains in the other, supporting correct answers where Vanilla fails. This illustrates the value of organizing storyline memory around the current information need and retaining the distinctions required for grounded prediction.

Table 8: MCQ cases from Manosaba. Earlier story provides reader-only background; Current scene and options are excerpts of the shared local input. Three actual state-bookmark proposals and their synchronized contents are shown per case, with the primary supporting entry in bold. Text is translated and abridged.

Context & Question	Bookmarks	Answers
Act 1, Chapter 1 *Earlier story.* Noa was found dead in her cell, and Ema investigated its bloodstained, painted floor and nearby objects. Coco and Milia had passed the cell after a livestream without noticing a body; Coco therefore argued that Noa was still alive then. *Current scene (trial).* *Coco:* "With that much blood, wouldn't we have noticed the smell?" *Ema:* "What could tell us whether there was a smell at the time?" A. Butterfly painted on the floor; B. Photo of fruit; C. Spray paint; D. Disassembled parts.	*Proposed sections:* Ema's recalled scene observations; **Material used for the butterfly drawing**; Origin and use of the disassembled parts. *Ema's recalled scene observations:* During the trial, Ema noted scratches on the floor but no white paint on the arrow. She argued that another object had made the scratches, then continued examining when Noa died and whether someone could enter the cell. **Material used for the butterfly drawing: During the trial, Ema recalled the clues found at Noa's crime scene: a white-painted floor, a crossbow arrow, a butterfly drawn in blood, a spray-paint smell, scratches, and a bucket of mechanical parts.** *Origin and use of the disassembled parts:* Ema and Meruru had found dismantled crossbow parts in a corridor bucket the previous evening. During the trial, Ema argued that the crossbow was already dismantled while Noa was alive and was difficult to reassemble.	Baseline: B (incorrect) Bookmarks: C (correct)
Act 1, Chapter 5 *Earlier story.* Nanoka's body was discovered as the execution platform rose from underground. It appeared partly frozen. Ema examined the body before the trial, but the cause of death remained unknown. *Current scene (trial).* *Margo:* "Nanoka's body had no wounds." *Ema:* "Wait, there was a wound. It was ..." A. Fingertips; B. Heart; C. Inside her mouth.	*Proposed sections:* **Injury locations Ema observed on Nanoka's body**; Known information on Nanoka's cause of death; Bloodstains on Nanoka's fingertips and hands. **Injury locations Ema observed on Nanoka's body: On examining the body, Ema found a wound inside Nanoka's mouth and faint bloodstains on her fingertips. No other conspicuous external injuries were observed.** *Known information on Nanoka's cause of death:* The cause of death remained undetermined. Recorded clues included faint fingertip bloodstains, a note marked 169231 in her pocket, and her gun alongside bullet marks in ice in the freezer room. *Bloodstains on Nanoka's fingertips and hands:* At the time of the trial, the record described faint bloodstains adhering to Nanoka's fingertips, but no wounds on the fingertips themselves.	Baseline: B (incorrect) Bookmarks: C (correct)

D EVALUATION GENERALITY VALIDATION

Open-domain Test We evaluate BOOKMARKS on 200 new starting scenes involving interactions between characters from the same artifact, with 10 turns generated per scene. We compare BOOKMARKS against 3-way ETA using both `gpt-4.1` and human judges. Figure 6 reports the win, tie, and loss rates. Both judges prefer BOOKMARKS more often than 3-way ETA, supporting its generalization to unseen interactive scenarios.

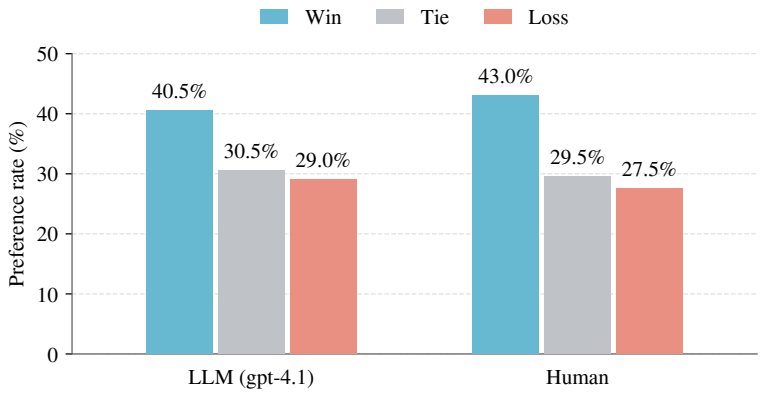

Figure 6: Open-domain role-playing evaluation against 3-way ETA on 200 scenes. Win and loss denote preference for BOOKMARKS and 3-way ETA, respectively.

E Artifact and Character Background

Tables 9 and 10 summarize the six evaluated artifacts and their characters.

Table 9: Background of the six evaluated artifacts.

Artifact	**Background**
The Melancholy of Haruhi Suzumiya (Haruhi)	A school comedy and mystery centered on Haruhi, whose impulses unknowingly reshape reality. Kyon and the SOS Brigade navigate ordinary club activities alongside supernatural incidents.
JoJo's Bizarre Adventure (JOJO)	In Part 3, Jotaro and his companions journey across the world to confront DIO. Their encounters combine tactical Stand battles with flamboyant personalities and tense alliances.
Fullmetal Alchemist (FMA)	Two brothers seek to restore their bodies after a forbidden alchemical ritual. Their journey explores sacrifice, political conflict, and the human cost of power.
A Game of Thrones (AGOT)	Noble houses compete for power through alliances, betrayal, and war. Family loyalties and personal ambitions shape their choices amid growing political conflict.
Poppin'Party (PoPiPa)	Five high school friends form a band and pursue their musical ambitions. Band Story 0 and Event 321 explore their relationships through practice, performance, and everyday life.
Magical Girl Witch Trials (Manosaba)	Thirteen girls imprisoned as suspected witches face murders and trials. Their search for the truth depends on testimony, accumulated clues, and each girl's magical abilities.

Table 10: Background of the evaluated characters.

Artifact	Character	Background
Haruhi	Haruhi	An impulsive and hyperactive high school girl whose unusual worldview sets off many of the story's strange events.
	Kyon	A sarcastic and level-headed student who narrates the story while serving as Haruhi Suzumiya's reluctant stabilizing partner.
	Nagato	A quiet and unreadable SOS Brigade member with extraordinary intelligence and mysterious otherworldly origins.
	Koizumi	A smiling transfer student and esper who supports the Brigade while carefully concealing important secrets.
	Asahina	A timid and gentle upperclassman drawn into the SOS Brigade as its cute and enigmatic mascot, often swept into its chaos.
JOJO	Jotaro	A stoic high schooler and leader of the Stardust Crusaders, known for Star Platinum and unshakable resolve.
	Polnareff	A bold French swordsman whose flamboyant personality and Silver Chariot make him a vivid ally.
	Joseph	A quick-thinking Joestar whose tricks, exaggeration, and improvisation often turn battles in his favor.
	DIO	A charismatic and ruthless vampire whose ambition and overwhelming presence define him as a legendary enemy.
	Kakyoin	A calm and analytical Crusader who fights with Hierophant Green and its precise emerald attacks.
	Avdol	A wise and loyal Egyptian Stand user whose Magician's Red controls powerful flames.
	Iggy	A stubborn Boston Terrier Stand user who dislikes cooperation but proves unexpectedly important.
FMA	Edward	A talented and stubborn young alchemist traveling to restore the bodies he and his brother lost.
	Alphonse	A kind boy whose soul is bound to a suit of armor, journeying with his brother to recover what was taken.
	Winry	A gifted automail mechanic and childhood friend of the Elrics, known for technical skill and steady compassion.
	Roy	A charismatic State Alchemist and flame user who seeks to reform the military from within.
	Ling	A determined prince from Xing who pursues immortality while carrying the burden of national duty.
AGOT	Tyrion	The sharp-witted youngest Lannister, navigating politics with intelligence, courage, and dark humor despite lifelong scorn.
	Daenerys	An exiled Targaryen princess who grows from a hesitant political pawn into a determined claimant to power.
	Cersei	An ambitious queen whose beauty masks ruthless devotion to family, status, and control.
	Jaime	The infamous Kingslayer, charming and deadly yet deeply conflicted by duty, loyalty, and reputation.
	Robb	The dutiful heir of Winterfell, forced into leadership and war before he is ready.
	Eddard	The honorable Lord of Winterfell, stern in duty and deeply committed to justice and family.
	Arya	An independent Stark girl who rejects courtly expectations in favor of freedom, survival, and the blade.
	Catelyn	The practical Lady of Winterfell, driven by fierce maternal loyalty and political urgency.
	Sansa	The elder Stark daughter, raised with courtly dreams that are tested by harsh political reality.
	Jon	Eddard's brooding illegitimate son, shaped by questions of identity, duty, and belonging.
	Bran	A curious young Stark whose fall places him on an unexpected and consequential path.

Continued on next page

Table 10 (continued)

Artifact	Character	Background
PoPiPa	Kasumi	An upbeat vocalist and guitarist whose starry enthusiasm pulls others forward and begins the band's journey.
	Tae	A free-spirited lead guitarist with strong technique and quirky instincts, often moving at her own pace while enriching the band's sound.
	Rimi	A shy and gentle bassist who gains confidence through performance and supports the group with warmth.
	Saaya	A caring and dependable drummer whose family-minded steadiness anchors the band in practice and daily life.
	Arisa	A sharp-tongued but reliable keyboardist whose practicality and quick judgment keep the band organized and grounded.
Manosaba	Gokucho	The mysterious owl warden who oversees the prison, its inmates, and the witch trials.
	Ema	A warm, outgoing girl who longs for friendship and fears rejection.
	Hiro	A diligent, principled student whose strong sense of justice leaves little room for compromise.
	An-An	A withdrawn girl who rarely speaks and communicates through a sketchbook.
	Noah	An aloof street artist whose celebrated work conceals its creator's identity.
	Leia	A popular actress who feels responsible for leading and protecting others.
	Miria	A gentle, timid girl whose flashy appearance contrasts with her hesitant manner.
	Margo	An affable con artist who keeps others at a distance through concealed distrust.
	Nanoka	A guarded loner who knows the prison well and resents its authority.
	Alisa	A confrontational runaway who dislikes others and, most of all, herself.
	Sherry	A cheerful girl who calls herself a detective and has no moral scruples.
	Hanna	A proud girl who poses as an aristocrat despite growing up poor.
	Coco	A streamer whose cheerful persona conceals a sharp tongue and volatile moods.
	Meruru	An anxious, sensitive girl who worries about others and doubts herself.