ACTOR: ACTION-CENTERED TEMPORAL ORGANIZATION FOR LITERARY ROLE-PLAYING AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language agents for literary role-playing are typically modeled around dialogue generation, leaving characters' non-dialogue behaviors and their dynamic evolution over story time underrepresented. In this paper, we propose **ACTOR** (**A**ction-**C**entered **T**emporal **O**rganization for Literary **R**ole-Playing Agents), an automated framework for constructing such behavior trajectories from raw novel texts. ACTOR extracts dialogue and persona-relevant non-dialogue behaviors, represents each behavior using a Thought–Action–Observation structure, and organizes behavior trajectories according to the internal temporal order of the story. We build a behavior dataset from 143 diverse literary works. For each source behavior, we generate counterfactual behaviors that are contextually plausible yet inconsistent with the character's persona. We find that existing models struggle to reliably infer character-consistent actions from historical behaviors and evolving character states. Using these behavior pairs, we train models with Regularized Preference Optimization (RPO), which augments DPO with a negative log-likelihood loss on the source behavior to jointly optimize behavior preference and generation. We further design a four-way next-behavior selection task to evaluate the model's ability to faithfully mimic character actions. Across 17,560 evaluation instances, the model trained on ACTOR's preference pairs achieves substantially improved accuracy. This result demonstrates that structured, temporally organized supervision enables models to leverage character profiles and prior behaviors to infer subsequent actions, greatly advancing literary role-playing beyond mere dialogue imitation. Code and data are available at *anonymity*.

1 INTRODUCTION

As research on role-playing language agents advances, researchers have constructed richer role representations and high-quality training datasets (Wang et al., 2026). Early studies draw on diverse textual sources such as scripts, biographies, and novels to build faithful role representations. These sources support retrieval, fine-tuning, and generation constraints, yielding text-grounded character profiles and literary worlds (Shao et al., 2023; Ran et al., 2025). Recent works further dynamically update character profiles and memories to simulate evolving character states across different story stages, effectively improving the faithfulness of role-playing (Xiao et al., 2026; Peng et al., 2026b).

However, fine-grained timestamp-aware behavior modeling cannot be adequately supported by these representations. Some prior works treat actions as embedded components of dialogue messages and thus mainly serve dialogue generation (Wang et al., 2025). Other existing lines of research predict characters' next actions based on plot summaries or pre-constructed coarse-grained action sequences (Xu et al., 2025; Song et al., 2026). Consequently, literary-oriented RPLA still lacks character-centric atomic-level behavior trajectories. Meanwhile, existing methods generally organize character experiences following the narrative order of source texts, with little explicit distinction between the actual chronological order of events and their disclosure order in original literary works.

To address the above practical challenges, we propose a pipeline framework that automatically constructs character-behavior trajectories from full-length novels. Behaviors extracted by this framework include both verbal dialogues and non-dialogue actions, and the trajectories are organized strictly according to the actual chronological order of unfolding events, providing a unified data foundation for investigating character behaviors and their gradual evolution throughout the story pro-

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.

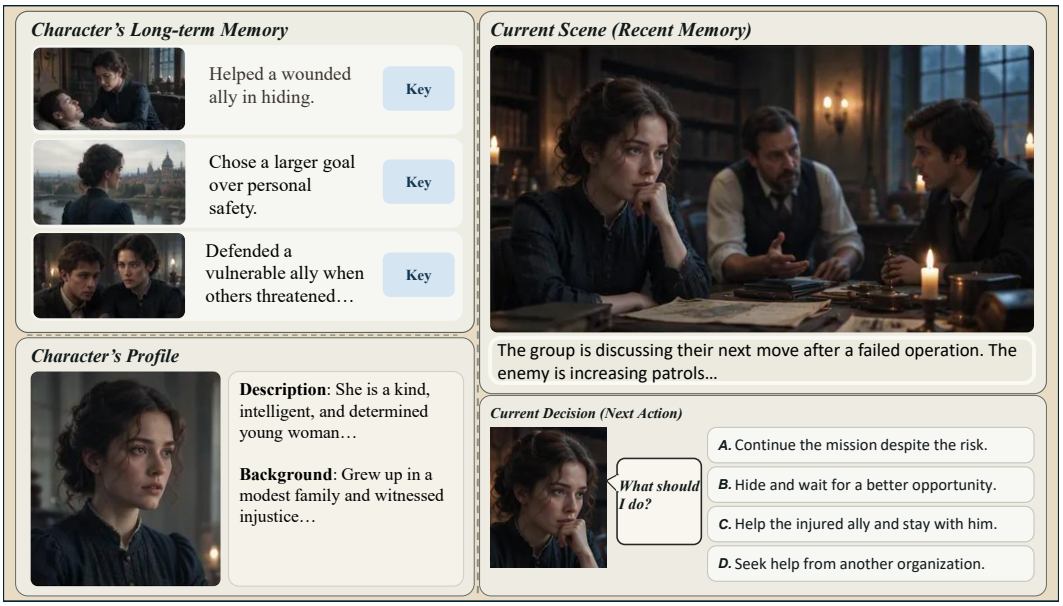


Figure 1: Overview of our behavior-modeling evaluation. Given character-related information, the model needs to select the most plausible next action from four options.

gression. During dataset construction, we concurrently generate contextually plausible yet persona-mismatched counterfactual actions for each ground-truth behavior to support subsequent model training and evaluation. Finally, we build our curated dataset extracted from 143 literary works.

For model training, we select 115 works from the curated corpus and use the first 80% of each character's behavior trajectory for training. As each source behavior is paired with a contextually plausible but persona-inconsistent counterfactual behavior, we use Regularized Preference Optimization (RPO), a joint preference-and-generation objective combining Direct Preference Optimization (DPO) with a negative log-likelihood loss on the source behavior. We treat the source behavior as chosen and one counterfactual behavior as rejected. These signals train the model to infer and generate actions consistent with the character's current state and behavioral prior history in context.

To evaluate the model's behavior-modeling capacity, we take the subsequent 20% of behaviors from each character trajectory within the aforementioned 115 works as temporal evaluation data. The remaining 28 unseen works excluded from training are entirely reserved for cross-book evaluation. For each evaluation instance, given the character profile, historical memory and current scene, the model is required to select the most plausible next action taken by the character among one ground-truth behavior and three distinct counterfactual behaviors. The former setup examines whether the model can continue observed character experiences, whereas the latter tests the model's ability to predict behaviors based on structured character information without access to the source literary works.

The main contributions of this paper are summarized as follows:

- We propose ACTOR, an automated framework that extracts character-centered atomic behaviors from raw literary texts and organizes behavior trajectories according to their actual occurrence time in the story. ACTOR addresses the lack of character behavior data in existing resources.
- We formulate literary role-playing as an action-centered temporal modeling task: at a specified point in the story, the model predicts a character's next behavior based on the character's state and previous behavior history. This task extends literary role-playing beyond dialogue response generation to the unified modeling of dialogue and non-dialogue behaviors.
- We construct a next-behavior selection benchmark to objectively evaluate a model's ability to imitate behavior based on character information, and use the behavior data generated by ACTOR to train models. Experiments on this benchmark show that training with these data improves a model's ability to imitate the behavior of a specified character.

2 Related Work

2.1 Literary Character Role-Playing Grounded in Textual Sources

Early role-playing systems primarily ground language models in character descriptions or historical dialogue (Chen et al., 2024; Chen et al., 2025). Li et al. (2023) retrieve character-specific dialogue from scripts to better support response generation. Shao et al. (2023) reconstruct character experiences from biographical descriptions and train a dedicated agent for each character. Wang et al. (2024) extend role-conditioned instruction tuning to a broader range of diverse roles. Lu et al. (2024) generate and refine role-playing supervision through self-alignment.

Later studies construct richer character information from literary sources (Park et al., 2025; Kang et al., 2025). Wang et al. (2025) extract novel-sourced authentic conversations alongside scene context and character experiences, augmenting dialogue turns with associated actions and thoughts. Cheng et al. (2026) leverage fine-grained psychological attributes to organize character memory for improved long-term consistency. Du et al. (2026) reverse-synthesize character reasoning from literary dialogues and apply reinforcement learning to enhance character motivation modeling. Tang et al. (2026) structure book-derived memory from the character's egocentric perspective to prevent god-view information leakage. These works enrich role-playing information, yet they are optimized for dialogue response generation rather than sequential behavior prediction.

2.2 Temporal Character Modeling via Action Trajectories

Another line of research examines characters at specific points in a narrative (Ahn et al., 2024; Xu et al., 2025). Ahn et al. (2024) require models to respect chapter-specific knowledge boundaries and avoid referring to future events that remain unrevealed within the current narrative segment. Xu et al. (2025) reconstruct key decisions from a character's prior experiences. Yuan et al. (2024) summarize character development from complete fictional works and evaluate whether the resulting profiles support character understanding. These studies show that character behavior depends not only on general personality, but also on experiences accumulated before the current story point.

Recent studies model character evolution based on plot progression and personal experiences (Yang et al., 2025b; Peng & Shang, 2025). Xiao et al. (2026) build temporally-ordered causal event graphs over novels for point-specific character profiles, while Zong et al. (2026) co-update character states and fictional worlds from dialogue and narrative text. Song et al. (2026) assess agent behavior across psychological phases of character development. Peng et al. (2026b) learn context-dependent behavioral rules from Fandom-sourced character-labeled action sequences and scene–action pairs, and Peng et al. (2026a) encode storylines as continuous actions with task-relevant history for next-action prediction. However, neither line of work extracts complete character action trajectories from novels nor reconstructs the genuine chronology of character experiences.

3 Introduction of ACTOR

3.1 Dataset Construction

T-A-O behavior extraction We define character behavior as contextual utterances and distinctive non-dialogue actions in literary scenarios. Deliberate silence and avoidance are treated as valid behaviors, whereas routine actions that reflect no unique personal traits or active choices are filtered out. Each behavior follows a Thought–Action–Observation (T-A-O) structure: Thought describes the internal mental state and decision rationale before an action, Action corresponds to the verbal utterance or physical conduct, and Observation captures subsequent interpersonal reactions and environmental changes perceptible to the character. To efficiently manage long-text contexts, the extraction pipeline adopts a hierarchical granularity fallback, processing full plots first, followed by individual scenes and character-level contexts as a last resort.

Counterfactual alternatives For each behavior, the system generates alternative actions that are plausible in the current situation but inconsistent with the target character. So selecting the correct behavior requires genuine understanding of the character. These alternatives are generated together with the extracted behaviors and are refined after constructing character trajectories and profiles.

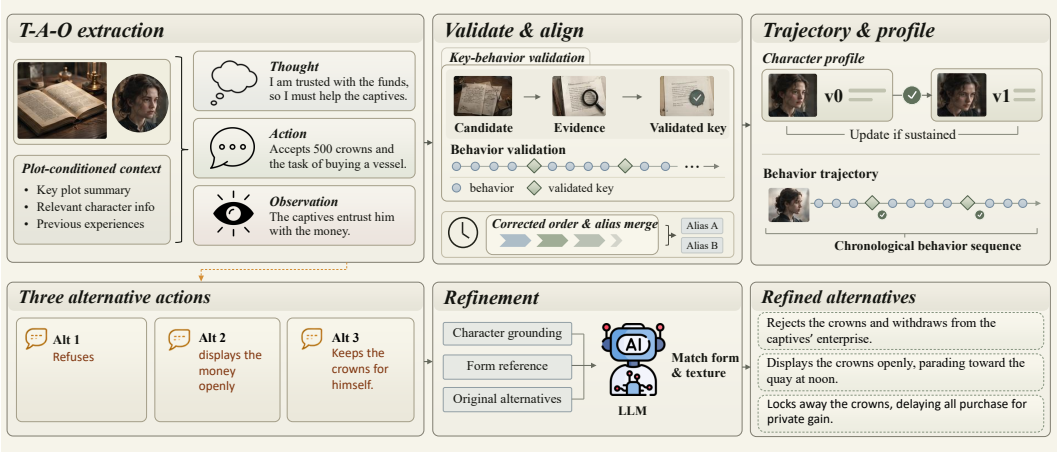


Figure 2: ACTOR dataset construction pipeline. Plot-conditioned context is transformed into Thought–Action–Observation (T-A-O) units, followed by key-behavior validation, temporal and entity alignment, and dynamic profile updates for sustained changes. Each behavior is paired with three situation-plausible but target-character-inconsistent alternatives, which are refined to match the real action's form and texture while leaving the real action unchanged.

Key-behavior validation As shown in the validation module of Figure 2, we clearly distinguish ordinary and key behaviors. A key behavior must satisfy two core criteria: the character has reasonable alternative options at the moment, and the selected action alters persistent states or future situational possibilities. To reduce annotation noise from single-pass generation, the system first produces candidate key behaviors, identifies scenes in subsequent plots that reflect long-term behavioral impacts, and verifies the candidates through an independent evidence-checking procedure. Only evidence-supported candidates are retained as valid key behaviors.

Temporal alignment After behavior verification, we re-annotate the precise occurrence time of each scene using the original plot text. The textual reading order is used as the default timeline, while ordering anomalies potentially caused by flashbacks or reversed narration are detected and converted into preliminary adjustment proposals. Each proposal is independently checked against the source text, and only scene reorderings supported by sufficient textual evidence are applied. Character aliases and scattered mentions are then normalized to construct complete character-level behavioral trajectories in the corrected in-story chronological order.

Character data The resulting behavior trajectory of a character c is represented as:

$$\mathcal{T}_c = (b_1, \ldots, b_{N_c}), \qquad b_i = (s_i, h_i, a_i, o_i, r_i, t_i, \kappa_i), \quad t_1 \leq \cdots \leq t_{N_c}, \tag{1}$$

where s_i denotes the scene context, (h_i, a_i, o_i) is the Thought–Action–Observation representation, r_i is the textual position, t_i is the reconstructed story-time position, and $\kappa_i \in \{0, 1\}$ indicates whether the behavior is a valid key behavior. The trajectory is ordered by story time, while the textual position is retained for source tracing.

Based on each trajectory, initial character profiles are derived from descriptions before the character's first appearance. Each valid key behavior triggers a profile-update check over all experiences accumulated after the latest profile version. A new profile version is saved only when these experiences lead to sustained character changes.

Alternative refinement Finally, the initially generated counterfactuals are refined to match the length and structural form of real behaviors while remaining situationally reasonable and inconsistent with the target character. We then anonymize all datasets. Book titles and character names are withheld from the model during both training and evaluation to prevent leakage from the base model's prior knowledge and improve generalization in character-action modeling.

3.2 Action Preference Training

Based on the character behavioral trajectories, we formulate character action prediction as a conditional preference learning task. The input context $\mathbf{x}$ consists of four types of inputs before each target action: a dynamic profile of the current character state, long-term memory of validated key behaviors, short-term memory of recent scene interactions, and situational context containing background, temporal cues, and participant information. All inputs are strictly sampled from pre-action story content, ensuring the model only uses information accessible to the character at that moment.

In terms of training paradigm, our dataset is naturally suited to preference optimization. For the same prediction context, we possess both the character's genuine behavior $\mathbf{y}^+$ extracted from the source text and counterfactual behaviors with matched length and format but inconsistent character profiles, which form an explicit relative preference relation: the genuine behavior should be preferred over actions the character would not take. Among the three counterfactual behaviors, the one closest in length to the genuine behavior is selected as $\mathbf{y}^-$ to minimize confounding effects from output-length differences. The source behavior and selected counterfactual are stored as chosen and rejected, respectively, defining the preference relation $\mathbf{y}^+ \succ \mathbf{y}^-$.

Direct Preference Optimization (DPO) learns this relation by optimizing the relative likelihood of the two behaviors against a fixed reference policy (Rafailov et al., 2023):

$$\mathcal{L}_{\mathrm{DPO}}(\theta) = -\mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}}\left[\log\sigma\left(\beta\left(\log\frac{\pi_\theta(y^+ \mid x)}{\pi_{\mathrm{ref}}(y^+ \mid x)} - \log\frac{\pi_\theta(y^- \mid x)}{\pi_{\mathrm{ref}}(y^- \mid x)}\right)\right)\right], \tag{2}$$

where π_θ is the policy being optimized, π_{ref} is the fixed reference policy, and β controls the scaling of the preference objective.

However, this relative objective does not explicitly require the likelihood of the source behavior itself to increase. Consequently, it may only widen the preference margin during action selection, without increasing the probability of directly generating utterances or actions consistent with the target character. Following Liu et al. (2024), we therefore adopt RPO, which augments the DPO objective with a token-level negative log-likelihood loss on the source behavior:

$$\mathcal{L}_{\mathrm{NLL}}(\theta) = -\mathbb{E}_{(x,y^+)\sim\mathcal{D}}\left[\frac{1}{|y^+|}\sum_{t=1}^{|y^+|}\log\pi_\theta\left(y_t^+ \mid x, y_{<t}^+\right)\right]. \tag{3}$$

The final training objective is:

$$\mathcal{L}_{\mathrm{RPO}}(\theta) = \mathcal{L}_{\mathrm{DPO}}(\theta) + \alpha\mathcal{L}_{\mathrm{NLL}}(\theta), \tag{4}$$

where α controls the contribution of direct supervision on the source behavior. We set $\beta = 0.1$ for all models and use $\alpha = 1.0$ for the Qwen-2.5 models and $\alpha = 0.25$ for Llama-3.1-8B. The resulting objective trains the model both to distinguish character-consistent behaviors from counterfactuals and to increase the likelihood of the source behavior itself.

4 Evaluation

4.1 Evaluation Setup

Evaluated Models We evaluate six pretrained models as baselines: Qwen-2.5-Instruct 7B and 14B (Yang et al., 2025c), Llama-3.1-Instruct 8B (Grattafiori et al., 2024), Qwen-3-32B (Yang et al., 2025a), CoSER-8B (Wang et al., 2025), and HER-32B model (Du et al., 2026). These baselines cover general-purpose instruction-tuned large language models and existing specialized role-playing models for literary character simulation. We further evaluate three ACTOR models trained with RPO: ACTOR-Qwen-2.5-7B, ACTOR-Qwen-2.5-14B, and ACTOR-Llama-3.1-8B.

Evaluation Data The evaluation data are directly extracted from character behavior trajectories constructed from 143 literary works. For 115 works that contribute to training, the first 80% of each character trajectory is used to construct the training data, while the remaining 20% is reserved for temporal evaluation. The other 28 works are excluded from training and are used entirely for cross-work evaluation. The final evaluation set contains 17,560 test instances, including 11,770 temporal-evaluation instances and 5,790 cross-work instances.

Table 1: Main results on the complete 17,560-instance evaluation set under the anonymous setting. All values are selection accuracy (%). Bold and underlined values denote the best and second-best results, respectively.

Model	Overall	Temporal	Cross-work	Key	Non-key
Qwen-2.5-Instruct 7B	71.34	71.29	71.43	71.10	71.55
Qwen-2.5-Instruct 14B	76.96	77.04	76.79	76.85	77.05
Llama-3.1-Instruct 8B	69.15	68.45	70.59	69.97	68.43
Qwen-3-32B	78.30	77.97	78.96	79.14	77.56
CoSER-8B	57.41	56.89	58.46	60.56	54.63
HER-32B	65.65	65.02	66.93	67.24	64.25
ACTOR-Qwen-2.5-7B	<u>80.20</u>	<u>80.23</u>	<u>80.14</u>	80.21	<u>80.19</u>
ACTOR-Qwen-2.5-14B	**86.00**	**86.16**	**85.68**	**86.40**	**85.65**
ACTOR-Llama-3.1-8B	78.04	78.24	77.63	<u>79.37</u>	76.87

Task and Metrics We formulate evaluation as a four-way next-behavior selection task. Given the character profile, behavioral memories, and current scene before the target behavior, the model must select the candidate behavior most likely to be performed by the target character. Each instance contains one source-text behavior and three counterfactual candidates. Selection accuracy is the primary metric. We additionally report accuracy on temporal and cross-work subsets, as well as on key and non-key behavior subsets, to examine model performance across different settings.

Open-ended Generation Evaluation Beyond next-behavior selection, we adopt the CoSER role-playing evaluation framework (Wang et al., 2025) to examine open-ended character interaction. CoSER evaluates models through multi-turn scenarios and combines environment simulation, next-state prediction, and language-model-based judging. It provides scores along four core dimensions: plot consistency, human-likeness, character fidelity, and plot quality. We use this evaluation to test whether ACTOR-derived training transfers from selecting character-consistent behaviors to generating coherent actions in dynamic interactive story settings.

4.2 Main Results

Table 1 reports selection accuracy on the evaluation set under the anonymous setting.

All three ACTOR models outperform their corresponding backbones, with overall gains ranging from 8.86 to 9.04 percentage points. ACTOR-Qwen-2.5-14B achieves the highest overall accuracy of 86.00%, while ACTOR-Qwen-2.5-7B reaches 80.20% and surpasses the larger Qwen3-32B baseline. **These results clearly show that ACTOR improves character-consistent behavior selection across diverse model families and sizes.** Since all models are evaluated under the same strictly anonymous setting, the gains cannot be attributed to access to book titles or original character names. The comparison with Qwen3-32B further suggests that model scale alone does not substitute for targeted behavioral supervision tailored for literary character modeling.

The cross-work subset contains literary works excluded from training, making it a test of whether the learned behavior patterns transfer beyond familiar stories. **ACTOR retains substantial gains on these unseen works:** all three models improve over their backbones by 7.04–8.89 percentage points. The two Qwen-based models show similar gains on the temporal and cross-work subsets, whereas the Llama-based model has a smaller cross-work gain. These improvements are unlikely to arise solely from continuing trajectories seen during training. Instead, the learned behavioral supervision generalizes effectively to unseen characters and narrative scenarios.

The improvements extend to both key and non-key behaviors. For example, ACTOR-Qwen-2.5-7B gains 9.11 percentage points on key behaviors and 8.64 points on non-key behaviors; the other two ACTOR models show similarly balanced gains. **ACTOR therefore improves the selection of ordinary persona-relevant behaviors as well as salient plot events.** This matters for role-playing, where character consistency must also be maintained in less consequential, everyday interactions.

CoSER-8B and HER-32B obtain lower scores on this selection benchmark despite being developed for role-playing. **Their results suggest that training for open-ended role-playing does not neces-**

Table 2: Representative cases from the evaluation results. Only decision-relevant evidence is shown.

Case	Relevant pre-action evidence	Qwen-2.5-7B-Instruct	ACTOR-Qwen-2.5-7B / Source behavior
State transformation	Formerly cold and dismissive of poverty, the character later resolves to change and repeatedly behaves with kindness and generosity. He only appears severe when the clerk arrives late.	Punishes the clerk and refuses further support for his family.	Raises the clerk's salary, supports his family, and asks him to add coal to the fire.
Dialogue continuity	The profile lists only the character's occupation. In the conversation, she alternates between teasing disclosures and assurances about another person's visits.	Promises to disclose the information: "I'll tell."	Reassures the interlocutor that she will not disclose it: "Certainly not."
Action adaptation	After declaring his willingness to kill an opponent and drawing his sword, the character is challenged by another character who intervenes.	Abandons the attack and shields the original target.	Redirects the attack toward the intervening character.

sarily yield strong fine-grained action selection. The difference may reflect the objectives of these models and should not be taken as evidence of weaker role-playing ability overall. We therefore examine open-ended interaction separately using the CoSER framework.

4.3 Qualitative Case Study

To complement the quantitative results, we present three illustrative instances in which ACTOR-Qwen-2.5-7B selects the behavior consistent with the source text, whereas its backbone selects a counterfactual behavior. As shown in Table 2, these cases require different forms of behavioral reasoning: tracking a character's long-term personality transformation, maintaining local dialogue coherence, and adapting an ongoing action to changes in the current scene. For readability, the contexts shown in the main text are manually summarized; more detailed versions of the three instances are provided in Tables 10–12 in Appendix C.

The first case concerns a character whose behavior has changed over time. The backbone selects a punitive action, which is superficially plausible given the character's earlier reputation for coldness. However, the behavioral history shows he has recognized the consequences of his former conduct and repeatedly acted with kindness and generosity. His severe manner at the beginning of the current interaction is therefore performative rather than evidence of a return to his former state. ACTOR instead selects the source-consistent action of raising the clerk's salary and supporting his family. Thus, ACTOR can **prioritize a temporally updated character state over an earlier salient trait**.

In the second case, the character profile provides little useful personality information beyond the character's occupation. The decision must therefore be inferred primarily from the ongoing conversation, which alternates between direct answers, teasing disclosures, and assurances. The backbone predicts an eager promise to disclose the information, whereas ACTOR selects the brief assurance appearing in the source text. This example suggests that ACTOR can **use the sequential organization of recent behaviors when profile cues are insufficient to determine the next action**.

The third case highlights ACTOR's ability to maintain action continuity after a change in the scene. The character has already drawn his sword against an opponent when another character intervenes. ACTOR redirects the attack toward the intervening character, preserving the character's existing intent while responding to the new situation. In contrast, the backbone incorrectly protects the original target, which conflicts with both the preceding attack and the character's stated objective.

Table 3: Open-ended generation results under the CoSER evaluation framework.

Model	Plot Consistency	Human-likeness	Character Fidelity	Plot Quality	Average
CoSER-8B	16.07	6.96	30.83	25.31	19.79
Qwen-2.5-Instruct 7B	11.70	1.17	21.87	6.97	10.43
ACTOR-Qwen-2.5-7B	**22.57**	**3.88**	**41.12**	**20.40**	**21.99**
Qwen-2.5-Instruct 14B	25.30	3.90	**42.95**	26.60	24.69
ACTOR-Qwen-2.5-14B	**25.30**	**15.00**	38.50	**34.75**	**28.39**
Llama-3.1-Instruct 8B	6.35	2.80	16.40	3.85	7.35
ACTOR-Llama-3.1-8B	**16.75**	**7.65**	**21.10**	**23.80**	**17.33**

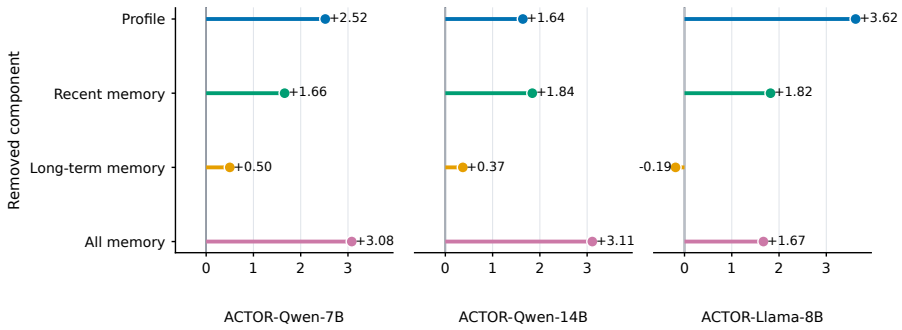


Figure 3: Accuracy changes after removing different context components. Each panel corresponds to one ACTOR model, and the vertical reference line denotes the full-context performance. Positive values indicate performance drops after removing the component.

The contrast indicates that effective prediction requires **preserving the character's initial intent while updating the immediate action in response to the intervention**.

4.4 Open-ended Generation Results

To verify whether ACTOR can generalize beyond candidate behavior selection, we test the models on open-ended multi-turn character interaction using the CoSER evaluation framework. GPT-5.6-Luna serves as both helper and judge model across all experiments. Since our experimental setup differs from the original CoSER work, we compare models under a unified evaluation protocol instead of directly matching our absolute scores against those reported in the original paper.

CoSER is constructed from multi-character dialogues extracted from authentic literary works, where each message integrates speech, physical actions, and inner thoughts. It formulates character-specific supervised training by taking scene context and other characters' utterances as input to predict the target character's responses. Additionally, it optimizes environment modeling and next-speaker prediction to facilitate multi-agent simulation. Such data structures and training objectives fundamentally differ from ours, yet our models still achieve performance gains.

As shown in Table 3, all three ACTOR models achieve higher overall scores than their corresponding backbone models, with particularly clear improvements for ACTOR-Qwen-2.5-7B and ACTOR-Llama-3.1-8B. The improvements are not uniform across individual dimensions because ACTOR is not trained on CoSER-specific interaction procedures or output formats. Nevertheless, the consistent improvement in overall performance suggests that, even without explicit training on CoSER data, **preference learning over temporally structured character trajectories can enable models to exhibit features associated with character dialogue style in open-ended interaction.** These results also demonstrate ACTOR's broader application potential. Details are shown in appendix D.

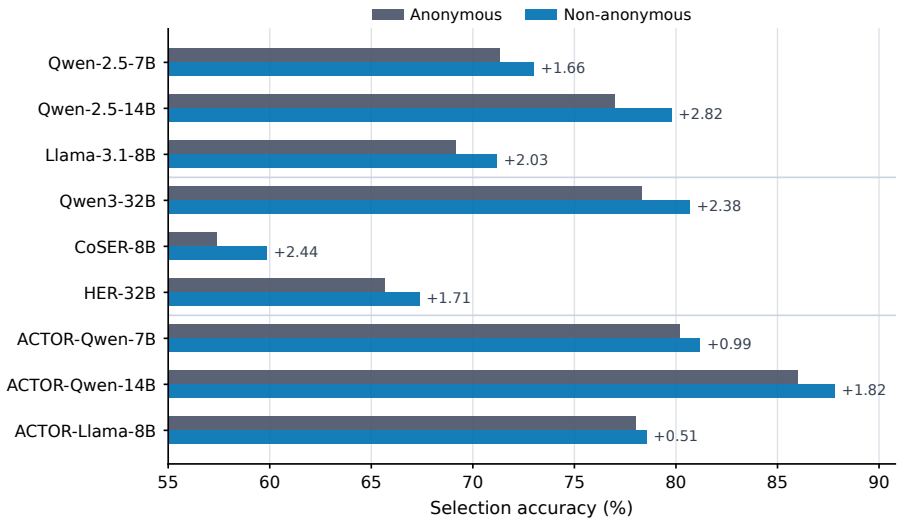


Figure 4: Selection accuracy under the anonymous and identity-provided settings.

4.5 Ablation Study

Figure 3 reports the context ablation under the anonymous setting. We keep the Current Scene unchanged and separately remove the Character Profile, Recent Memory, or Long-term Memory from the input context. We find that **removing either the Character Profile or Recent Memory consistently reduces the performance of all three ACTOR models.** This indicates stable character traits and recent behavioral context provide complementary signals for global personality reasoning and local behavioral adaptation. In contrast, removing Long-term Memory changes the accuracy of the two Qwen models by no more than 0.5 percentage points and slightly improves the Llama model. This difference may reflect the partial overlap between Long-term Memory and the Character Profile, making the marginal contribution of long-term information dependent on the backbone.

Figure 4 compares model performance under anonymous and identity-provided settings. Supplying book titles and character names boosts accuracy for all models, as explicit identities help retrieve prior knowledge of familiar characters. ACTOR's gains (0.51–1.82 pp) are smaller than those of baselines. **ACTOR's advantage therefore does not primarily depend on explicit character identities.** Its strong anonymous-setting performance shows the model leverages provided character information and scene context, rather than memorized knowledge of known books or characters. This benefit is valuable for role-playing original or unseen characters.

5 Conclusion

We presented **ACTOR**, an automated framework for extracting and chronologically organizing dialogue and persona-relevant non-dialogue behaviors from literary texts. By pairing source behaviors with counterfactual alternatives, ACTOR formulates literary role-playing as an action-centered temporal prediction task and trains models with an RPO objective that combines preference learning with direct supervision on source behaviors. Experiments show that ACTOR consistently improves next-behavior prediction across different settings, with gains on unseen works indicating transfer beyond memorizing training trajectories. Ablation and qualitative analyses further highlight the importance of updated character profiles and recent behavioral memory. Supplementary evaluation with the CoSER evaluation framework suggests that these benefits extend to open-ended character interaction, although the framework may inherit noise from automatic data construction and currently relies on separate selection and generation protocols. Future work will improve construction reliability, broaden language and genre coverage, and develop unified evaluations.

AI USE STATEMENT

We used generative AI tools to assist with extracting and cleaning character behavior data from literary texts, including behavior structuring, candidate key-behavior identification, temporal information organization, and counterfactual behavior generation. Generative AI was also used to assist with drafting and polishing the manuscript and with generating and revising figures. All AI-generated data were subject to programmatic validation, source-based evidence verification, and manual inspection. The authors also reviewed all AI-assisted text and figures. Generative AI was not used to replace the authors in formulating the core research questions, drawing theoretical conclusions, or interpreting experimental results. The authors take full responsibility for the final content, research claims, and all artifacts produced with the assistance of generative AI.

ETHICS STATEMENT

This work involves no human participants, private personal data, or direct interaction with real individuals, so institutional review board approval is unnecessary. We construct character behavior trajectories from literary texts. To comply with copyright and licensing constraints, our released resource contains only derived anonymized records. Full original novels are not redistributed.

Automatically generated behaviors and counterfactuals may carry biases inherited from source literature. Biases can also be introduced by language models. Such biases may produce stereotyped portrayals. These portrayals touch on gender, culture, and social roles. For this reason, the resulting models cannot serve as reliable depictions of real people. They must not be deployed for high-stakes decision-making. We mitigate annotation errors through source-based validation alongside structural auditing. Manual inspection further supports this process. We take responsibility for ethical usage and thorough documentation of all released materials.

REPRODUCIBILITY STATEMENT

The ACTOR data-construction pipeline is described in Section 3.1. The RPO training objective and its model-specific settings are provided in Section 3.2. The data splits, evaluated models, task formulation, and open-ended evaluation protocol are specified in Section 4.1. The appendix further provides dataset statistics and structural audits (Appendix A), the complete dataset schema (Appendix B) and the prompts used (Appendix E). Code and data are available at *anonymity*.

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 Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 3291–3325, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.197. URL https://aclanthology.org/2024.findings-acl.197/.

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. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=xrO70E8UIZ. Survey Certification.

Nuo Chen, Yan Wang, Yang Deng, and Jia Li. The oscars of AI theater: A survey on role-playing with language models, 2025. URL https://arxiv.org/abs/2407.11484.

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. *Transactions of the Association for Computational Linguistics*, 14:510–529, 2026. doi: 10.1162/tacl.a.645. URL https://aclanthology.org/2026.tacl-1.24/.

Chengyu Du, Xintao Wang, Aili Chen, Weiyuan Li, Rui Xu, Junteng Liu, Zishan Huang, Rong Tian, Zijun Sun, Yuhao Li, Liheng Feng, Deming Ding, Pengyu Zhao, and Yanghua Xiao. HER: Human-like reasoning and reinforcement learning for LLM role-playing. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 25725–25762, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.1283. URL https://aclanthology.org/2026.findings-acl.1283/.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Sua Kang, Chaelim Lee, Subin Jung, and Minsu Lee. Fic2bot: A scalable framework for persona-driven chatbot generation from fiction. *Electronics*, 14(19), 2025. ISSN 2079-9292. doi: 10.3390/electronics14193859. URL https://www.mdpi.com/2079-9292/14/19/3859.

Cheng Li, Ziang Leng, Chenxi Yan, Junyi Shen, Hao Wang, Weishi MI, Yaying Fei, Xiaoyang Feng, Song Yan, HaoSheng Wang, Linkang Zhan, Yaokai Jia, Pingyu Wu, and Haozhen Sun. Chatharuhi: Reviving anime character in reality via large language model, 2023. URL https://arxiv.org/abs/2308.09597.

Zhihan Liu, Miao Lu, Shenao Zhang, Boyi Liu, Hongyi Guo, Yingxiang Yang, Jose Blanchet, and Zhaoran Wang. Provably mitigating overoptimization in RLHF: Your SFT loss is implicitly an adversarial regularizer. *Advances in Neural Information Processing Systems*, 37:138663–138697, 2024.

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 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)*, pp. 7828–7840, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.423. URL https://aclanthology.org/2024.acl-long.423/.

Jeiyoon Park, Chanjun Park, and Heuiseok Lim. CharacterGPT: A persona reconstruction framework for role-playing agents. In Weizhu Chen, Yi Yang, Mohammad Kachuee, and Xue-Yong Fu (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 3: Industry Track)*, pp. 287–303, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-194-0. doi: 10.18653/v1/2025.naacl-industry.24. URL https://aclanthology.org/2025.naacl-industry.24/.

Letian Peng and Jingbo Shang. Codifying character logic in role-playing. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=RoVS9vmpd2.

Letian Peng, Ziche Liu, Yiming Huang, Longfei Yun, Kun Zhou, Yupeng Hou, and Jingbo Shang. Bookmarks: Efficient active storyline memory for role-playing, 2026a. URL https://arxiv.org/abs/2605.14169.

Letian Peng, Kun Zhou, Longfei Yun, Yupeng Hou, and Jingbo Shang. Deriving character logic from storyline as codified decision trees. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12480–12509, San Diego, California, United States, July 2026b. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.568. URL https://aclanthology.org/2026.acl-long.568/.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2023.

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)*, pp. 15898–15912, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.773. URL https://aclanthology.org/2025.acl-long.773/.

Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. Character-LLM: A trainable agent for role-playing. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 13153–13187, Singapore, December 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.emnlp-main.814/.

Woojung Song, Nalim Kim, Sangjun Song, Chaewon Heo, Jongwon Lim, and Yohan Jo. Arcane: Do role-playing language agents stay in character at the right time?, 2026. URL https://arxiv.org/abs/2606.05553.

Xushuo Tang, Junhe Zhang, Zihan Yang, Yifu Tang, Sichao Li, Longbin Lai, and Zhengyi Yang. Staying in character: Perspective-bounded memory for book-based role-playing agents, 2026. URL https://arxiv.org/abs/2606.25632.

Noah Wang, Z.Y. Peng, Haoran Que, Jiaheng Liu, Wangchunshu Zhou, Yuhan Wu, Hongcheng Guo, Ruitong Gan, Zehao Ni, Jian Yang, Man Zhang, Zhaoxiang Zhang, Wanli Ouyang, Ke Xu, Wenhao Huang, Jie Fu, and Junran Peng. RoleLLM: Benchmarking, eliciting, and enhancing role-playing abilities of large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 14743–14777, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.878. URL https://aclanthology.org/2024.findings-acl.878/.

Xintao Wang, Heng Wang, Yifei Zhang, Xinfeng Yuan, Rui Xu, Jen tse Huang, Siyu Yuan, Haoran Guo, Jiangjie Chen, Shuchang Zhou, Wei Wang, and Yanghua Xiao. CoSER: Coordinating LLM-based persona simulation of established roles. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=BOrR7YqKUt.

Ye Wang, Jiaxing Chen, and Hongjiang Xiao. Role-playing agents driven by large language models: Current status, challenges, and future trends, 2026. URL https://arxiv.org/abs/2601.10122.

Zhihao Xiao, Mengting Li, Xintao Wang, Linfeng Li, Limin Shui, Mengqi Ji, and Borui Cai. Dream: LLM-based Dynamic Role-playing via Event-Aware Memory Graph. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, KDD '26, pp. 5686–5697, New York, NY, USA, 2026. Association for Computing Machinery. ISBN 9798400722592. doi: 10.1145/3770855.3818027. URL https://doi.org/10.1145/3770855.3818027.

Rui Xu, Xintao Wang, Jiangjie Chen, Siyu Yuan, Xinfeng Yuan, Jiaqing Liang, Zulong Chen, Xiaoqingdong, and Yanghua Xiao. Character is destiny: Can persona-assigned language models make personal choices? In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 15038–15059, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.813. URL https://aclanthology.org/2025.findings-emnlp.813/.

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

Bohao Yang, Hainiu Xu, Jinhua Du, Ze Li, Yulan He, and Chenghua Lin. Evolvtrip: Enhancing literary character understanding with temporal theory-of-mind graphs, 2025b. URL https://arxiv.org/abs/2506.13641.

Table 4: Summary of the dataset.

Split	Works	Records	Characters	Scenes	Key	Non-key
Training	115	270,856	3,939	30,818	26,374	244,482
Evaluation	143	17,560	2,437	11,201	8,221	9,339
Temporal / cross-work	115 / 28	17,560	–	–	–	–

Qwen: An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025c. URL https://arxiv.org/abs/2412.15115.

Xinfeng Yuan, Siyu Yuan, Yuhan Cui, Tianhe Lin, Xintao Wang, Rui Xu, Jiangjie Chen, and Deqing Yang. Evaluating character understanding of large language models via character profiling from fictional works. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8015–8036, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.456. URL https://aclanthology.org/2024.emnlp-main.456/.

Qing Zong, Yue Guo, Mengxin Yang, Yiwen Guo, and Yangqiu Song. Evolvingworld: An open-schema framework for co-evolving role-play agents and world model in interactive literary world, 2026. URL https://arxiv.org/abs/2607.17250.

A Dataset Statistics

The final anonymized corpus contains behavior records from 143 non-empty literary works. The training partition contains 270,856 source preference records from 115 works. The evaluation partition contains 17,560 four-way next-behavior questions from 143 works: 11,770 temporal-evaluation instances from the 115 training works and 5,790 cross-work instances from 28 additional works. The evaluation set contains 8,221 key-behavior instances and 9,339 non-key-behavior instances. Table 4 summarizes the corpus composition.

The context supplied to each instance is compact but structured. In the evaluation set, the median character trajectory contains 6 recorded behaviors, the median profile contains 120 whitespace-delimited words, and the median recent and long-term memories contain 8 and 7 entries, respectively. Training records have longer trajectories because they include the complete training prefix; their median trajectory length is 15 behaviors. Profile freshness is reported per record and indicates whether the profile was immediately updated, moderately distant, or stale relative to the target behavior. Trajectory-length distributions are shown in Figures 5 and 6.

We also audit the structural validity of the released records. The rebuilding validation retained 288,416 records and excluded 279 records before release. All 17,560 evaluation instances contain exactly four options and a valid answer, while all 270,856 training records contain non-empty chosen and rejected behaviors. No duplicate record identifiers or malformed placeholder strings were detected. A small number of participant-description keys are composite or relationship-style expressions rather than single "Character n" placeholders; they are reported separately for manual review and are not counted as known-alias or malformed-placeholder failures. The complete audit is given in Table 5.

Finally, source chosen and rejected behaviors are closely matched in surface length: their medians are both 13 whitespace-delimited words, with means of 21.69 and 21.30 words, respectively. The corresponding ranges and percentiles are reported in Table 6; the distributions are visualized in Figures 7 and 8.

Table 5: Evaluation-question and preference-pair audit.

Audit	Training	Evaluation	Result
Records with required answer fields	270,856	17,560	valid
Invalid preference/options records	0	0	none
Duplicate record IDs	0	0	none
Malformed placeholders	0	0	none
Non-placeholder participant-key occurrences	764	65	manual review
Four-option questions	–	17,560	all evaluation records

Table 6: Length statistics for source preference pairs, measured in whitespace-delimited words.

Response	Count	Mean	Median	P10	P90	Min–Max
Chosen	270,856	21.6877	13	6	42	1–3635
Rejected	270,856	21.3041	13	5	41	2–3297

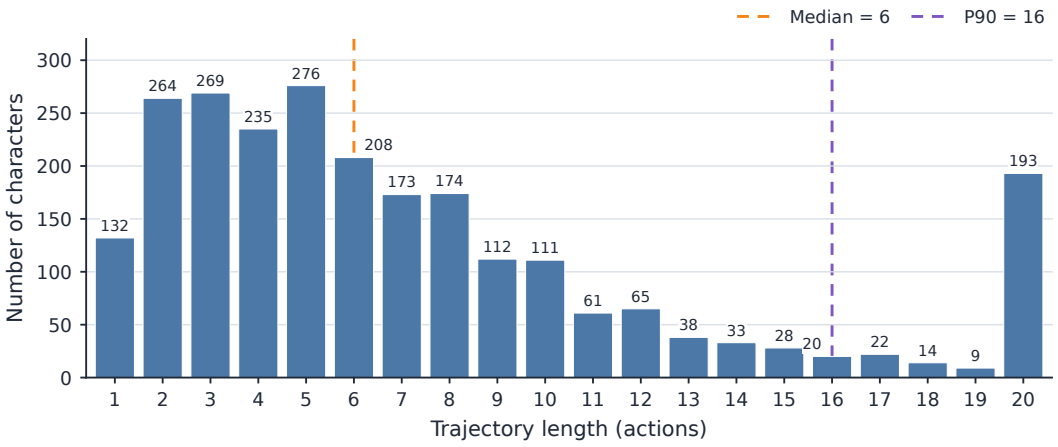


Figure 5: Evaluation trajectory-length distribution.

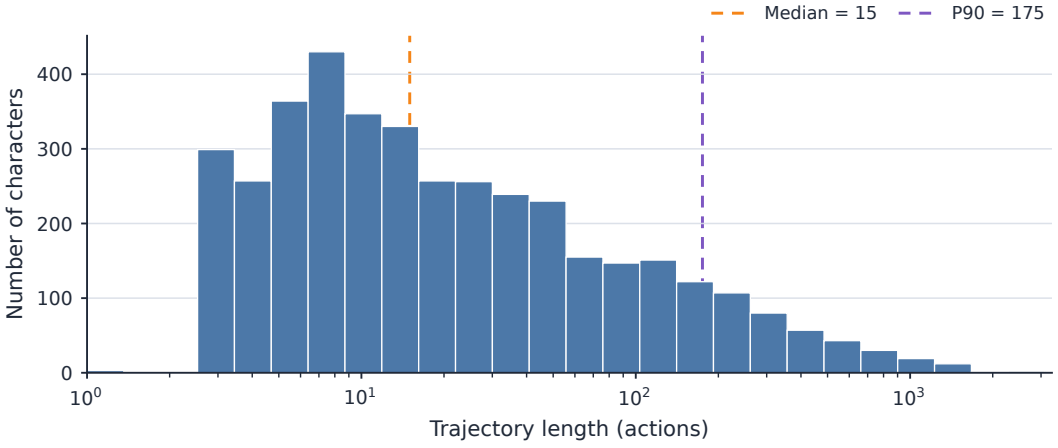


Figure 6: Training trajectory-length distribution.

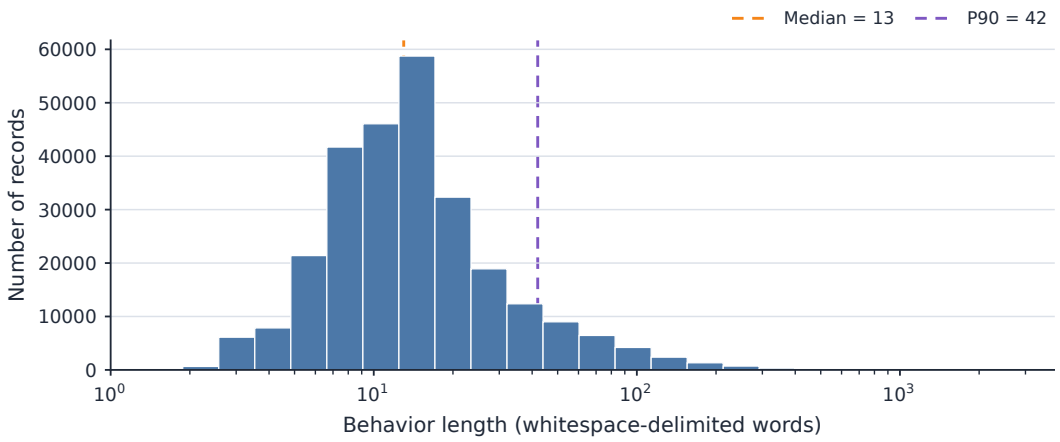


Figure 7: Length distribution of chosen behaviors.

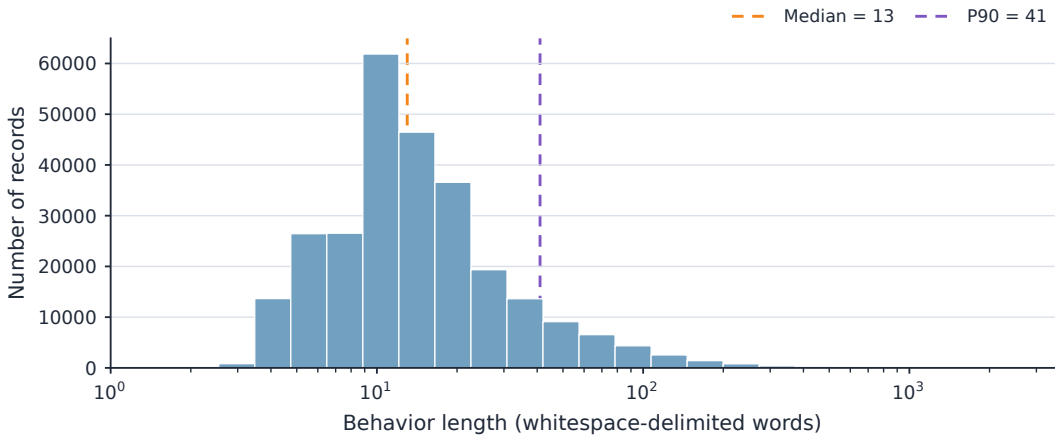


Figure 8: Length distribution of rejected behaviors.

B Dataset Schema

Each ACTOR instance is centered on a target character and a behavior occurring at a particular point in the story. All instances contain the character profile, behavioral memories, and current-scene context available before the target behavior. Training instances additionally contain a preferred source behavior and a rejected counterfactual, whereas evaluation instances contain four candidate behaviors and the corresponding answer label. All textual fields are anonymized before being provided to the model.

Table 9 presents an actual anonymized evaluation instance from the released dataset. The textual fields and candidate options are preserved as stored in the dataset; only internal bookkeeping fields are omitted from the table.

For training instances, the same contextual fields are paired with *chosen* and *rejected* behaviors instead of four candidate options. This shared schema enables the extracted trajectories to support both preference-based training and objective next-behavior evaluation.

Table 7: Context fields shared by ACTOR training and evaluation instances.

Field	Type	Description
character	Anonymized label	Anonymized name of the target character shown to the model.
profile	Structured object	The latest character-profile version available before the target behavior. It summarizes persistent traits, relationships, and established states.
long_term_memory	List of T-A-O records	Previously verified key behaviors that provide long-range behavioral history.
recent_memory	List of T-A-O records	Behaviors from the most recent preceding scenes, representing the local behavioral context.
scene	Structured object	The current-scene description, temporal information, present participants, and anonymized participant descriptions. It excludes the target behavior and all subsequent events.
is_critical	Boolean	Indicates whether the target behavior is a verified key behavior.
action_uid, *scene_uid*	Identifiers	Internal identifiers for the target behavior and its containing scene. They are used for traceability and are not required as model inputs.

Table 8: Split-specific fields in ACTOR training and evaluation instances.

Field	Split	Description
chosen	Training	The behavior performed by the target character in the source text. It is treated as the preferred response during RPO training.
rejected	Training	A contextually plausible but character-inconsistent counterfactual behavior paired with the source behavior.
options	Evaluation	Four candidate next behaviors, consisting of one source behavior and three counterfactual behaviors.
correct_answer	Evaluation	The identifier of the candidate corresponding to the source behavior.
character_action_order	Training / Evaluation	The position of the target behavior in the character-level chronological trajectory.
split	Training	Partition label indicating that the instance belongs to the training prefix of a character trajectory.
sample_id, *question_id*	Training / Evaluation	Unique identifiers used to trace each instance during construction, validation, and evaluation.

C Detailed Case-Study Instances

Tables 10–12 provide additional details for the three cases discussed in Section 4.3. To maintain readability, we retain the evidence most relevant to each behavioral decision and use ellipses to indicate omitted profile, memory, or scene content.

Table 9: An actual anonymized next-behavior selection instance from the ACTOR evaluation set.

Component	Dataset content
Target character	Character 1
Character profile	Character 1 is a young law student and the heir presumptive to Character 2, positioned by expectation as the future baronet. Beyond this social and familial standing, no stable temperament, values, interaction style, personal goals, or relationship orientation are evidenced before his first recorded action.
Recent memory	**Thought:** I do not feel eager to be drawn into the family's notice. **Action:** Receives Character 2 overtures without warmth. **Observation:** Character 2 persists, making allowance for what he treats as youthful modesty.
Long-term memory	**Thought:** I do not feel eager to be drawn into the family's notice. **Action:** Receives Character 2 overtures without warmth. **Observation:** Character 2 persists, making allowance for what he treats as youthful modesty.
Current scene	**Time:** in one of their spring excursions to London, when Character 3 was in her first bloom. **Present:** Character 3, Character 2, Character 1. The scene has just begun. Character 3 is the eldest daughter of Character 2 and Character 4; Character 2 is a baronet and the father of Character 3, Character 5, and Character 6; Character 1 is the heir presumptive to Character 2 and a young law student expected to become the future baronet.
Candidate A	Behaves agreeably during the introduction.
Candidate B	Regards Character 3 with open disdain throughout.
Candidate C	Defers slavishly to Character 2 guidance.
Candidate D	Presses Character 3 hand with ardent presumption.
Correct answer	A

D CoSER Interaction Examples

Tables 13 and 14 present two qualitative comparisons from the CoSER evaluation. Each model is evaluated in a separate simulation of the same initial scene. Context and character descriptions are summarized from the evaluation inputs. We select excerpts from the first eight character messages in each simulation, omitting next-speaker predictions and environment messages. Turn numbers preserve their original order; gaps indicate omitted turns, and ellipses mark omissions within a turn. Speaker attributions follow the recorded role fields, including inconsistencies; names are shortened in the first table and redundant name prefixes are omitted. Scores are scene-level averages of the four CoSER dimensions, computed from the full simulations rather than these excerpts. These selected cases illustrate interaction patterns and limitations.

In Table 13, the Qwen-2.5-Instruct 7B backbone initially attends to the dome model, but Tommaso's next turn addresses himself by name. The interaction then settles into declarations of shared purpose, and both characters repeat almost the same handshake description instead of discussing the design. CoSER-8B avoids that repetition and raises a question about the dome's shape, although the exchange soon turns to general reassurance. ACTOR-Qwen-2.5-7B sustains the subject more clearly: Tommaso asks about a flying buttress, Michelangelo responds in terms of support and visibility, and later turns discuss what the dome symbolizes. **The useful contrast is whether the conversation develops beyond the initial scene cue.** The backbone recognizes that cue but does not maintain a distinct or progressing exchange; ACTOR does so in these selected turns. This reading concerns conversational progression, not the architectural or historical correctness of the design claims.

Table 10: Detailed instance for Case 1, which requires tracking a long-term character transformation. Ellipses indicate omitted context.

Field	Content
Profile	Character 1 was formerly cold, solitary, suspicious, anti-Christmas, and dismissive of poverty. After witnessing the consequences of his former life, he becomes grateful, humble, playful, openly generous, and eager to repair his past hardness. . . .
Long-term memory	**Thought:** I now see that this neglected and plundered dead man's case may be my own if my present life continues in this direction. **Action:** He recognizes that his own life is tending toward the same unloved death. **Observation:** . . . **Thought:** I must bind myself to a changed life and preserve the lessons I have learned. **Action:** He vows to honor Christmas throughout the year and to live by the lessons of the past, present, and future. **Observation:** . . .
Recent memory	He goes to church, walks through the streets with pleasure, treats children kindly, and questions beggars rather than ignoring them. He then approaches his nephew's house, overcomes his anxiety, and joins the family gathering. . . .
Current scene	The character waits for his clerk, who arrives eighteen and a half minutes late. He initially challenges the clerk in a severe tone and pretends that he will no longer tolerate such conduct. His severity is described as feigned and suddenly playful, although the clerk remains alarmed.
Candidates	**A:** Punishes the clerk for arriving late and refuses to provide additional warmth or assistance. **B:** Offers only conditional assistance after scrutinizing the clerk's family and conduct. **C:** Wishes the clerk a merry Christmas, raises his salary, promises to assist his family, and asks him to add coal to the fire. **D:** Offers help primarily as a public demonstration of his own generosity.
Predictions	Source behavior: **C**; Qwen-2.5-Instruct 7B: **A**; ACTOR-Qwen-2.5-7B: **C**.

Table 11: Detailed instance for Case 2, which requires maintaining local dialogue continuity. Ellipses indicate omitted context.

Field	Content
Profile	Character 1 is identified only as a chambermaid responsible for room care. No personal traits, social relationships, motives, history, or attitudes are available in the profile.
Long-term memory	The chambermaid has previously revealed that Character 3 visited nearly every day and that the visits were made to Character 4. She further reveals that Character 4 and Character 3 sometimes went out together. . . .
Recent memory	When asked about Character 4, the chambermaid answers directly that Character 4 went out only a few minutes earlier. Character 2 finds the news strange and hurriedly searches through his belongings.
Current scene	During an extended conversation, the chambermaid alternates between direct answers, teasing remarks, and evasive responses. She discusses Character 3's visits, tells Character 2 that the visits occurred nearly every day, and reveals that Character 3 and Character 4 sometimes went out together. Character 2 becomes visibly disturbed and questions her about the details. . . .
Candidates	**A:** promises: "Certainly not." **B:** greedily hints: "Pay me." **C:** haughtily says: "Never, sir." **D:** eagerly says: "I'll tell."
Predictions	Source behavior: **A**; Qwen-2.5-Instruct 7B: **D**; ACTOR-Qwen-2.5-7B: **A**.

Table 14 shows a different failure in the Qwen-2.5-Instruct 7B backbone. It introduces Ellen's concern and the tray of food, but the turn labelled Heathcliff adopts the caregiver's perspective and then repeats Ellen's appeal to eat almost verbatim. Later turns dwell on the wind and the room rather than developing a response to his refusal. CoSER-8B maintains a reciprocal dialogue, yet

Table 12: Detailed instance for Case 3, which requires adapting an ongoing action to a change in the current scene. Ellipses indicate omitted context.

Field	Content
Profile	Character 1 is a steward and trusted envoy whose loyalty lies primarily with his mistress and her political faction. Although normally self-protective under direct threat, he is willing to use violence against Character 5 to demonstrate his factional loyalty. . . .
Long-term memory	**Thought:** If I meet Character 5, killing him will prove my allegiance to the party I serve. **Action:** He declares that, if he encounters Character 5, he will show which party he follows. **Observation:** . . . **Thought:** Here is the proclaimed traitor I hoped to meet; killing him can raise my fortunes. **Action:** He draws his sword and threatens to kill Character 5. **Observation:** Character 10 intervenes between them.
Recent memory	The character repeatedly follows his mistress's instructions, refuses to disclose her confidential letter, and prioritizes her orders despite warnings that the journey is dangerous. . . .
Current scene	Character 1 prepares to attack Character 5. Character 10 steps between them and refuses to move. Character 1 threatens Character 10 and orders him to stand aside, but Character 10 remains defiant and prepares to resist the attack.
Candidates	**A:** Kneels before Character 10. **B:** Casts away his sword. **C:** Fights Character 10. **D:** Shields Character 5, whom he had intended to kill.
Predictions	Source behavior: **C**; Qwen-2.5-Instruct 7B: **D**; ACTOR-Qwen-2.5-7B: **C**.

shifts from the supplied conflict toward a reassuring exchange. ACTOR-Qwen-2.5-7B keeps the offered food, refusal to eat, and withdrawal in view, but turns 3 and 5 are labelled Ellen Dean while voicing Heathcliff's perspective. **Stronger engagement with the scene therefore does not resolve speaker attribution.** Although ACTOR has the highest average score here, its character-fidelity score is lower than the backbone's (42.50 versus 62.50); the scene average should not be interpreted as improvement on every dimension.

Together, the examples suggest that ACTOR's behavioral supervision can support more sustained exchanges and closer engagement with a scene's conflict in open-ended interaction. They also expose a remaining failure mode: maintaining the correct speaker and perspective across turns. The examples complement the aggregate CoSER results without establishing how often either pattern occurs.

E PROMPT

To facilitate reproducibility, this section lists the prompts used throughout the ACTOR model. Tables 15–17 present the prompts adopted in the action trajectory extraction stage. Tables 18–19 are for dynamic character-profile construction and counterfactual rewriting. Tables 20–21 provide the shared context and output prompts used for RPO training and four-way next-behavior evaluation.

Table 13: Conversational progression in *The Agony and the Ecstasy* (scene 0006). The scene context is summarized and model outputs are excerpted from the recorded simulations.

Scene. Michelangelo and Tommaso examine a wooden model of St. Peter's dome. After years of design work, Michelangelo is conscious of his declining health and wants his vision to survive him.

Characters and motivations. Michelangelo is independent and devoted to his craft; he wants Tommaso to understand the design. Tommaso admires his work, worries about his health, and wants to support him.

Qwen-2.5-Instruct 7B **Scene score: 26.75**

1. **Michelangelo:** … I stand before the model of the dome, my hands tracing the lines with a steady hand. …

2. **Tommaso:** … (Tommaso, you're my inspiration. Let's do this together.)

3. **Michelangelo:** … Together, we shall see this through. (We shake hands, a silent pact between us.)

4. **Tommaso:** (Our hands meet, a bond of friendship and shared purpose.) …

5. **Michelangelo:** (Our hands meet, a bond of friendship and shared purpose.) (I return his smile, a flicker of hope lighting my eyes.) …

6. **Tommaso:** (Our hands meet, a bond of friendship and shared purpose.) (I return his smile, a flicker of hope lighting my eyes.) …

CoSER-8B **Scene score: 19.50**

3. **Michelangelo:** … What do you think of making it more concave?

4. **Tommaso:** … What inspired this change?

5. **Michelangelo:** … I just feel that it needs more... movement. But I fear I may be losing my perspective.

6. **Tommaso:** … Your vision is unparalleled. Trust in your instincts. …

8. **Tommaso:** … Your work will outlive us all. How can I assist you in bringing this vision to life?

ACTOR-Qwen-2.5-7B **Scene score: 46.00**

1. **Michelangelo:** … "Tommaso, I have a new idea for the dome," …

2. **Tommaso:** … "But why not use a flying buttress?" …

3. **Michelangelo:** … "It would support the weight, but it would also be visible. I want the dome to be a perfect sphere, hidden from view."

5. **Michelangelo:** … "It's not just a building, Tommaso, it's a symbol."

6. **Tommaso:** … "A symbol of what, Michelangelo?"

7. **Michelangelo:** … "Of God's creation, of man's ingenuity, of our faith and our dreams."

Table 14: Scene alignment and speaker attribution in *Wuthering Heights* (scene 0189). The scene context is summarized and model outputs are excerpted from the recorded simulations.

Scene. During Heathcliff's final days, Ellen Dean visits his chamber. He has withdrawn from company and refuses food, while she worries about his deteriorating condition.

Characters and motivations. Ellen Dean is practical and compassionate, with a strong sense of duty toward those in her care. Heathcliff is preoccupied with rejoining Catherine and resists interruption. Her concern conflicts with his withdrawal.

Qwen-2.5-Instruct 7B **Scene score: 23.75**

1. **Ellen Dean:** ... (I place the tray on the table, my hand trembling slightly as I watch Heathcliff's shallow breathing.)

2. **Heathcliff:** [Heathcliff's fate is sealed, yet I must fulfill my duty.] (I approach the bed, my steps hesitant, and gently place the tray beside him, hoping he might find some solace in the food.)

3. **Ellen Dean:** ... "Eat, Mr. Heathcliff. It's the least I can do."

4. **Heathcliff:** ... "Eat, Mr. Heathcliff. It's the least I can do."

7. **Heathcliff:** ... The wind howls mournfully, as if echoing his fate. ...

CoSER-8B **Scene score: 26.50**

2. **Heathcliff:** ... I am... better. Thank you.

3. **Ellen Dean:** ... I've brought you some food, sir. Shall I open the windows?

6. **Heathcliff:** ... Do you... do you miss me, Nelly?

7. **Ellen Dean:** ... I... I do, Mr. Heathcliff. You've been kind to me in the past.

8. **Heathcliff:** ... Thank you, Nelly. That's all I need to know.

ACTOR-Qwen-2.5-7B **Scene score: 47.50**

1. **Ellen Dean:** ... Heathcliff, you've not eaten. I brought you some broth.

2. **Heathcliff:** [Heathcliff's eyes flicker with a mix of anger and despair as he looks at the food.]

3. **Ellen Dean:** (Heathcliff looks up, his eyes hollow and distant) Nelly, I've no appetite. (His voice is a hoarse whisper) I've no need for food.

5. **Ellen Dean:** ... I've seen enough of this world. (His voice is a mere shadow of its former self) I've tasted enough of its bitterness.

8. **Ellen Dean:** ... Heathcliff, you're not yourself. (She tries to keep her voice steady) You've been ill for days.

Table 15: Prompts for plot and character-context curation.

Prompt	Template and requirements
Plot Segmentation	*(intro)* Partition the supplied book chunk into consecutive, non-overlapping plot segments for downstream character-action extraction. *(requirements)* Return the exact first and last sentences of every segment. Preserve source-text order and continue truncated segments from the previous chunk. Maintain the canonical character registry and report newly observed aliases. Do not create overlapping segments or include a broad parent plot together with a contained sub-plot. Return strict JSON only.
Overlap Repair	*(intro)* Repartition a source-text region that has been assigned to overlapping plot segments. *(requirements)* Cover the supplied region exactly once, without gaps or duplicated passages. Segments must be mutually exclusive and remain in source order. Start a new segment only when there is a meaningful change in the immediate goal, setting, interaction, or outcome. Return a parsimonious set of segments with exact first-sentence anchors.
Character Enrichment	*(intro)* Generate compact character details for each supplied character in a finalized plot. *(requirements)* Use only the plot summary and the supplied plot text. For every character, generate a short description of identity, social position, or relevant relationships, together with a concise account of the character's actions, decisions, intentions, or experiences in this plot. Do not use outside knowledge or information from later plots. Preserve the supplied canonical names and return strict JSON.
Alias Grouping	*(intro)* Group character names that refer to the same person and select a canonical name. *(requirements)* Merge formal names, honorifics, nicknames, and diminutives only when the identity is supported. Do not merge family members merely because they share a surname or title. If a name is genuinely ambiguous, place it in the unresolved list rather than guessing. Every input name must appear exactly once in either a group or the unresolved list.

Table 16: Prompts for scene and T-A-O behavior extraction.

Prompt	Template and requirements
Scene Splitting	*(intro)* Split a finalized plot into consecutive scenes representing continuous interactions in one setting. *(requirements)* Scenes must jointly cover the entire plot text without gaps or overlaps. Start a new scene when the setting changes or the interaction shifts to a different focus. For each scene, provide a description, canonical participants, relationship context, and the exact first sentence used to locate the scene in the source text.
T-A-O Extraction	*(intro)* Extract character behavior trajectories as Thought–Action–Observation loops from the supplied plot text. *(requirements)* An action is a decision-bearing behavior for which the character has plausible alternatives. Include consequential decisions, decisional speech such as refusal, persuasion, promise, deception, or threat, and deliberate silence or non-intervention. Exclude passive information exchange, routine movements, decorative gestures, and actions with no meaningful character choice. Each step must contain exactly one atomic decision. Thought is written from the character's first-person perspective and must be grounded in the source text. Action describes the performed behavior or preserves a direct quotation when applicable. Observation records only reactions or environmental changes perceptible to the character; omniscient narration and later events are prohibited. For every behavior, generate exactly three situationally plausible but character-inconsistent alternatives. Mark a behavior as critical only when the character has a reasonable alternative and the selected behavior changes a persistent state or a future situational possibility. Record the scene's actual time and supporting evidence, rather than dates mentioned only in plans, memories, or documents. Return strict JSON only.

Table 17: Prompts for criticality verification and timeline reconstruction.

Prompt	Template and requirements
Criticality Locator	*(intro)* Locate later plots and scenes that may contain evidence of the long-term consequences of a candidate critical behavior. *(requirements)* Use only the supplied event ledger and later scenes. Do not select the candidate's own scene, an earlier scene, or a scene that merely repeats its wording or emotional tone. Return at most three supporting scene identifiers and a concise explanation of the possible downstream consequence. Return an empty list when no later evidence can be identified.
Criticality Verifier	*(intro)* Independently verify whether a candidate behavior should retain its critical-behavior label. *(requirements)* Keep a candidate only when the supplied later evidence contains a concrete event, relationship change, decision, or consequence that could plausibly differ under another reasonable choice. Every retained case must cite at least one supporting scene and action identifier. Vague plot importance, current-scene mood, or repeated wording does not constitute evidence. Return `keep` or `downgrade` in strict JSON.
Scene-Time Relabeling	*(intro)* Re-annotate the time at which the actions and dialogue in each scene actually occur. *(requirements)* Use only explicit or directly grounded evidence from the original plot text. A future appointment, a remembered event, or a date on a document is not automatically the time of the current scene. If the scene time is not established, return null for both the time label and its evidence.
Timeline Anomaly Detection	*(intro)* Detect confident cases in which the story-time order of scenes differs from their textual presentation order. *(requirements)* Treat textual order as the default timeline. Report only scenes or contiguous scene blocks whose primary actions and dialogue clearly occur earlier in story time. Do not move a scene merely because it contains a recollection, discussion of a past event, or a future plan. Return proposed moves and uncertain cases separately.
Timeline Review	*(intro)* Independently review each proposed timeline adjustment against the original plot text. *(requirements)* Check both the moved block and its insertion neighborhood. Return `accept` only when the boundaries and insertion point are supported, `reject` when the textual order should be preserved or evidence is insufficient, and `modify` when a different move is required. Provide source-grounded reasoning for every decision.
Entity Resolution	*(intro)* Resolve source-level character names into persistent character entities. *(requirements)* Group names only when the supplied time-ordered appearances establish that they refer to the same person. Do not merge people based only on shared surnames, titles, gender, or proximity. Every source name must appear exactly once in an entity group or in the unresolved list. Prefer conservative unresolved assignments to unsupported merges.

Table 18: Prompts for dynamic character-profile construction.

Prompt	Template and requirements
Profile Initialization	*(intro)* Construct the initial dynamic profile of a character immediately before the character's first recorded behavior. *(requirements)* Use only the pre-action character description. Do not use the first action, later events, future relationships, or outside knowledge of the novel. Summarize the character's supported identity, social position, temperament, interaction style, current relationships, and relevant lived experience in a compact profile. Return an evidence assessment together with the profile.
Profile Update	*(intro)* Update a character profile after a valid critical behavior and the accumulated actions since the previous profile version. *(requirements)* Set `changed` to true only when the action window causes a durable change in identity, social position, stable attitudes, important relationships, goals, self-understanding, or relevant lived experience. Repeated or temporary behavior does not require a new version. When `changed=false`, return a null profile. When `changed=true`, rewrite the complete current profile rather than producing a chronological action log. Do not use future events or counterfactual alternatives.

Table 19: Prompts for counterfactual rewriting and record-level anonymization.

Prompt	Template and requirements
Counterfactual Rewriting	*(intro)* Rewrite the initially generated counterfactual behaviors for a source behavior. *(requirements)* The three alternatives must remain plausible responses to the same scene while violating the target character's personality, values, goals, or behavioral style. Match the real behavior in length, structural form, speech format, register, and degree of formality. Preserve a quotation for dialogue behaviors and avoid quotations for non-dialogue behaviors. The alternatives must not be simple paraphrases and should differ along distinct behavioral dimensions. Return exactly three independent strings in strict JSON.
Record Anonymization	*(intro)* Replace person names in a dataset record with consistent anonymous character identifiers. *(requirements)* Identify every person-name form appearing in the record, including full names, nicknames, abbreviations, possessives, and title–name combinations. Map forms referring to the same person to the same `Character N`, ordered by first appearance. Preserve place names, animal names, relational terms, structural identifiers, and labels required for evaluation. Return a strict JSON name map.

Table 20: Shared character-context prompt used for RPO training.

Prompt	Template and requirements
Character Context	*(intro)* You are role-playing as the target character. Decide what the character will do next based on the supplied character profile, memories, and current scene. *(requirements)* Treat the provided Character Profile, Long-term Memory, Recent Memory, and Current Scene as the complete and authoritative history for the task. Do not rely on prior knowledge of the original novel, reconstruct omitted events, or infer future developments. The profile describes the current character state; long-term memory contains validated key behaviors; recent memory provides local behavioral context; and the current scene contains the situation immediately before the target behavior. During training, the source behavior is provided as `chosen` and one counterfactual behavior is provided as `rejected`. The model is trained to prefer the source behavior while also increasing its conditional likelihood under the auxiliary generation loss.

Table 21: Prompt used for four-way next-behavior selection and context ablations.

Prompt	Template and requirements
Next-Behavior Selection	*(intro)* Choose the action that the target character would most likely take next. *(requirements)* Use only the supplied Character Profile, Long-term Memory, Recent Memory, and Current Scene. Do not use prior knowledge of the original novel or any event that occurs after the target behavior. Four candidate behaviors are provided, including one source behavior and three counterfactual alternatives. Output exactly one letter: `A`, `B`, `C`, or `D`. Do not provide an explanation or any additional text.
Context Ablations	*(intro)* Evaluate the effect of individual character-information sources on next-behavior selection. *(requirements)* Use the same selection prompt while selectively removing the book identity, character profile, long-term memory, recent memory, or both profile and memory. All other scene and option information remains unchanged. The model must follow the same single-letter output constraint in every condition.