We sincerely thank all reviewers for their valuable feedback and encouraging comments.

Reviewer1: "Commendable contribution", "Interesting framework", "Potential practical applicability", "Strong results".

Reviewer2: "Adequately presented and well explained", "Clear", "Sound and plausible", "Promising empirical results and online tests".

Reviewer3: "Novel", "Valuable contribution", "Consistent improvements", "Clear insights", "Clearly written", "Reasonable".

Below, we address the reviewers' questions and clarify misunderstandings.

For Review1:

Showstopper1: Thank you for considering our empirical results strong. A convergence proof is included in the anonymous code repository, and a formal version will be added to the final paper.

Showstopper2: Figure-7 shows that even a single iteration yields substantial gains, making it suitable for online systems. As discussed in Section 4.3.1, inference is performed over all items but only for active users, balancing performance and cost with minimal overhead in large-scale deployments. Additional performance and time metrics (including offline training and inference) will be added in final version.

Showstopper3: While LLM-generated data may include hallucinations, prior work (KAR) showsthat their open-world knowledge brings substantial benefits to recommender. Furthermore, our iterative reflection and refinement mechanism is explicitly designed to detect and correct such issues, mitigating potential biases and hallucinations.

Showstopper4: Our goal is to enhance knowledge extraction from LLMs via System-2 thinking, so we focus on LLM-as-knowledge-enhancer methods like KAR. P5 is not LLM-based, and TallRec uses LLM-as-ranker, which makes them less comparable to our setting.

Showstopper5: We agree R^4ec relies on historical data and is not aimed at pure zero-shot. However, its strong performance in long-tail scenarios (4.1% revenue gain) demonstrates its effectiveness under sparse-data conditions. We leave pure cold-start settings for future work.

For Review2:

Showstopper1: Table-2 shows that R^4ec outperforms single-model fine-tuning (R^2ec) with a 0.60% relative AUC gain, and exceeds the prompting-based method (KAR) by 1.36%, indicating meaningful improvements. We initially used a single model for reasoning, reflection, and refinement, but found that generation and critique interfered with each other, yielding only marginal improvement over R^2ec. This motivated our two-model design, which will be elaborated in the final version. Specific failure cases are provided in anonymous GitHub.

Showstopper2: Although iterative self-correction and multi-step reasoning resemble aspects of System-2 thinking, they are typically implemented implicitly in prior RL-based methods (e.g., R1). In contrast, R^4ec explicitly separates reasoning and reflection into two coordinated models: an actor model for reasoning and refinement, and a reflection model for error detection and critique. This design aligns with the deliberate and error-aware nature of System-2 thinking. To our knowledge, this is the first application of System-2 thinking in recommendation.

Showstopper3: While the reflection model may occasionally introduce errors, Figure-7 shows that performance consistently improves with more iterations, suggesting that R^4ec is not overly sensitive to individual reflection flaws. Overall, the reflection process helps correct reasoning errors and enhances final recommendation performance.

Showstopper4: Our primary focus is to explore System-2 thinking in recommendation and demonstrate its benefits. Therefore, we compare with System-1-style knowledge extraction methods such as KAR and R^2ec. Broader comparison with recent LLM-based methods focusing on knowledge quality will be considered in future work.

Beyond the showstopper, we address other concerns as follows: 1.Originality: To mitigate hallucination in direct reasoning, we propose a novel application of the reflection and refine mechanism to introduce System-2 thinking into LLM-based recommender, an idea adapted from recent success in general LLM tasks, and to our knowledge, the first such attempt in this domain. As an initial and effective exploration, we anticipate our work can inspire further applications of System-2 thinking in recommendation. 2.Presentation: Prompt templates for item factual knowledge were provided in our anonymous GitHub for clarity. 3. Related Literature: We will expand the discussion and cite more closely related works in the final version. 4.Technical Soundness: As shown in Showstopper1 and Showstopper3, the reflection mechanism yields consistent improvements, and the two-model setup does not introduce prohibitive cost. 5.Reproducibility: All prompt templates were included in the anonymous GitHub, and datasets will be released upon acceptance.

For Review3:

Showstopper1: We fine-tune Qwen-2.5-7B to extract knowledge, avoiding repeated GPT calls and lowering cost. As shown in Figure-7, one iteration already yields strong performance, sufficient for deployment. Since online serving is not real-time inference, and reasoning is done offline for all items and active users, latency and cost remain acceptable.

Showstopper2: We agree that CTR prediction is a specific subtask of recommendation. We focus on it because it directly impacts revenue in our online advertising platform and serves as a representative task. While our experiments are centered on CTR, our method is general and extendable to other subtasks (e.g., reranking), which we plan to explore in future work.

Showstopper3: Thank you for the suggestion. We will cite these works and include comparisons using open-source implementations when available.