

TRACE: A Long-Term Conversational Recommendation Dataset with Evolving User Personas

Jihyung Ahn^{1,†} Seonah Han^{1,†} Chanho Park^{1,†}
Hyeongwoo IM² Daeyeong Kim² Subin Kim² Buru Chang^{1,*}
¹Korea University ²KT Corporation
{gina0520, sunahan, cksgh0984, buru_chang}@korea.ac.kr
{im.hyeongwoo, daeyeong.kim, subin.k}@kt.com

Abstract

Conversational Recommender Systems (CRS) have improved personalized recommendation through dialogue, but most studies and datasets remain limited to single-session settings. Such single-session settings make it difficult to capture long-term preference evolution and the interplay between latent behavioral patterns and explicit conversational styles, partly because real-world conversations aligned with users' long-term behavioral histories are scarce. To address this gap, we introduce *TRACE*, a long-term CRS dataset built from behavioral histories that reflect persistent user preferences. Our analysis shows that *TRACE* provides reliable and diverse behavioral personas, as well as higher-quality CRS interactions than existing CRS datasets. Building on *TRACE*, we introduce *EvoCRS*, a persona-based CRS framework that models accumulated and evolving preferences through updated behavioral and dialogue personas. Experiments show that both persona types improve recommendation performance, and that persona updates are important for capturing long-term preference evolution. Our code and datasets are available online.¹

1 Introduction

Conversational Recommender Systems (CRS) provide personalized recommendations through natural language interactions (Jannach et al., 2021). While they can capture users' immediate needs from dialogue, real-world preferences are shaped by long-term behavioral histories and repeated interactions. Thus, realistic CRS requires modeling users as temporally evolving entities whose preferences persist, change, and reappear across sessions.

However, existing CRS datasets provide limited support for long-term preference modeling. Most

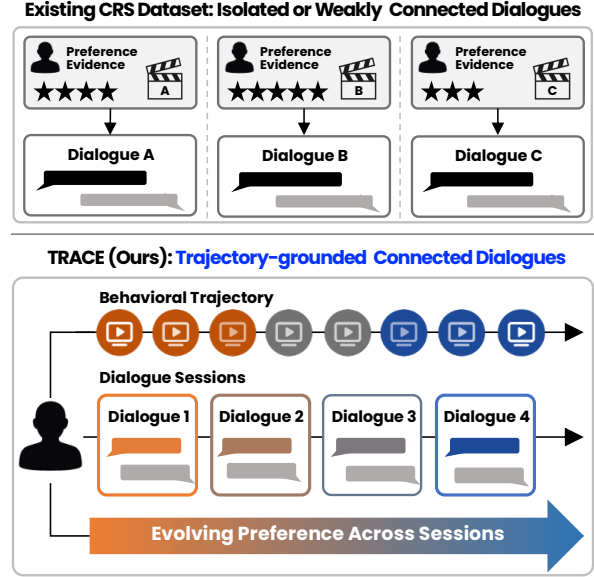


Figure 1: Comparison between existing CRS datasets and *TRACE*.

datasets are designed for single-session recommendation, where each dialogue is treated as an isolated interaction (Li et al., 2018; Kang et al., 2019; Hayati et al., 2020; He et al., 2023). Although some datasets introduce multi-session structures or historical information, their sessions are often weakly connected or organized as relatively independent recommendation episodes (Zhou et al., 2020; Liang et al., 2024). Recent LLM-generated datasets improve scalability, but they typically generate standalone conversations from reviews, ratings, or user-item evidence rather than continuous user-level behavioral trajectories (Kim et al., 2024; Liang et al., 2024; Martina et al., 2025). Consequently, existing datasets remain insufficient for studying long-term multi-session CRS, where conversations should be grounded in temporally ordered user histories and connected through evolving preference states.

To address this gap, we introduce *TRACE* (Temporally-grounded Recommendation with Aligned Conversations and Evolving personas), a

[†]Equal contribution

^{*}Corresponding author

¹<https://github.com/ahnji0520/TRACE>

long-term multi-session CRS dataset built from temporally ordered viewing histories. Unlike isolated reviews, viewing histories provide frequent chronological signals recorded at the time of consumption (Hu et al., 2008), making them suitable for observing how user preferences form and evolve. We organize each user’s viewing history into behavioral sequences and sessions, allowing dialogues to follow the temporal flow of user interactions. As shown in Figure 1, *TRACE* differs from prior datasets by grounding conversations in long-term behavioral trajectories rather than isolated or weakly connected dialogue episodes.

A key challenge is that behavioral histories and conversations reveal different aspects of user preferences. Behavioral histories capture persistent but implicit patterns, such as genre tendencies, consumption routines, and viewing motivations. We represent these patterns as *behavioral personas*, constructed from user behavior logs based on Uses and Gratifications Theory (Katz et al., 1973a). Since behavior logs alone cannot fully express conversational intent, situational context, or explicit requirements, we also construct *dialogue personas* from generated conversations. In *TRACE*, behavioral personas guide the generation of history-aligned dialogue sessions, dialogue personas summarize preferences revealed in those conversations, and both persona types are iteratively updated across sessions. This enables *TRACE* to capture long-term preference evolution from both behavioral and conversational perspectives.

Using this pipeline, *TRACE* contains more than 8k temporally connected recommendation dialogues from 1,136 users, with an average of 7.19 sessions per user. Although the dialogues are generated by LLMs, they are not produced independently; each session is grounded in the user’s chronological viewing history and evolving personas. Our analysis shows that *TRACE* provides reliable and diverse behavioral personas, and human evaluations show that its conversations are preferred over existing CRS datasets in naturalness, coherence, engagingness, and explainability. These results suggest that *TRACE* offers a high-quality testbed for long-term preference modeling in CRS.

To validate the utility of *TRACE*, we develop *EvoCRS*, a persona-based CRS framework that uses evolving behavioral and dialogue personas for recommendation. Rather than positioning *EvoCRS* solely as a new recommendation model, we use it to examine whether the multi-view signals in

TRACE improve long-term user modeling. Experiments show that behavioral and dialogue personas provide complementary signals and that persona updates help capture long-term preference changes. These findings demonstrate the value of *TRACE* for developing CRS models that exploit long-term, multi-view user information.

Our contributions are as follows:

- Introduce *TRACE*, the first long-term multi-session CRS dataset that grounds LLM-generated dialogues in temporally ordered viewing histories and evolving personas.
- Propose and validate a persona-evolving pipeline for generating history-aligned dialogues and updating behavioral and dialogue personas across sessions.
- Validate *TRACE* with *EvoCRS*, showing that both persona types improve recommendation and that persona updates are crucial for long-term preference modeling.

2 Related Work

2.1 Conversational Recommender Systems

Many existing CRS studies focus on single-session recommendation, inferring users’ immediate preferences from the current dialogue (Kostric et al., 2021; Kook et al., 2025). However, such methods mainly capture short-term conversational signals and overlook accumulated preferences across historical interactions, making recommendation unreliable when the current dialogue is sparse or ambiguous. Some studies incorporate historical interactions. *UCCR* (Li et al., 2022) uses historical dialogue sessions and look-alike users, while *Mem-oCRS* (Xi et al., 2024) stores and retrieves past preferences through memory. However, these methods treat past interactions mainly as auxiliary evidence for the current session, rather than tracking evolving preference states across sessions.

In contrast, *EvoCRS*, built on *TRACE*, represents preferences as evolving personas updated across sessions and integrated with the current dialogue context. This motivates datasets that support temporally connected CRS and long-term user modeling.

2.2 Datasets for Conversational Recommendation

Existing CRS datasets provide limited support for long-term preference modeling. Early human-written datasets (Li et al., 2018; Hayati et al., 2020)

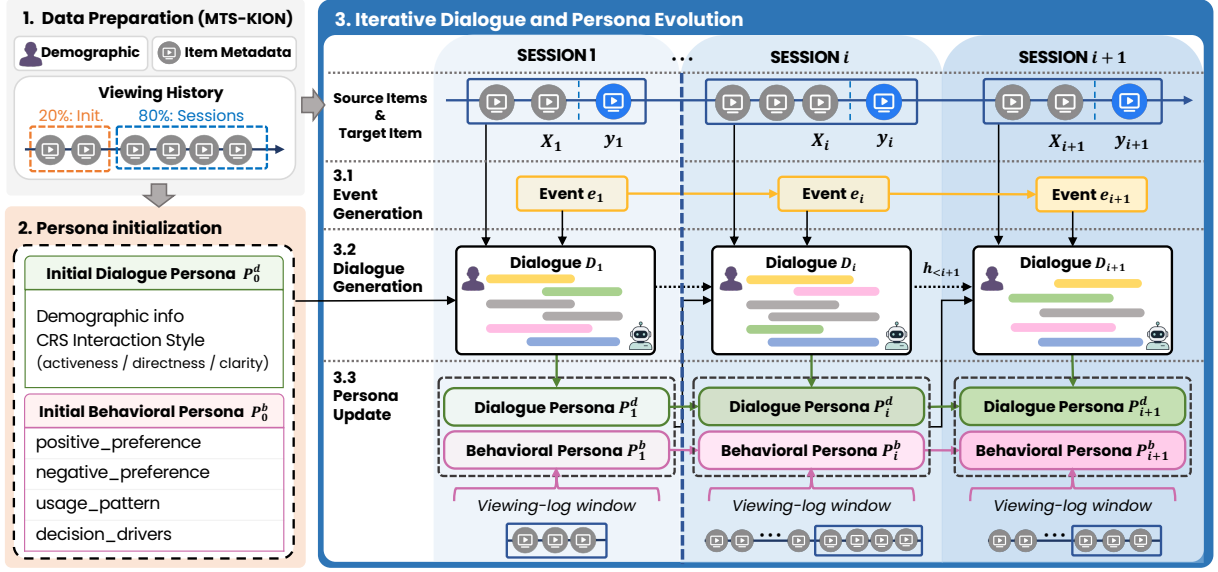


Figure 2: Construction pipeline of *TRACE*. The pipeline initializes user personas from early viewing logs and demographic information, segments long-term user histories into sessions, and iteratively generates events, persona-conditioned dialogues, and updated personas across sessions.

are single-session, while multi-session datasets such as *TG-ReDial* (Zhou et al., 2020) and *DuRecDial* (Liu et al., 2020) use updated user profiles, watching records, or topic-guided conversations. However, these datasets only partially support preference evolution, as their sessions are either weakly connected or lack grounding in continuous user-level behavioral trajectories. Recent LLM-generated datasets improve scalability by constructing dialogues from reviews, ratings, or user-item evidence (Kim et al., 2024; Liang et al., 2024; Martina et al., 2025), but typically produce standalone dialogues rather than conversations organized along chronological user histories.

In contrast, *TRACE* builds LLM-generated recommendation dialogues from temporally ordered viewing histories into user-level multi-session trajectories. By grounding sessions in chronological behavioral histories and evolving personas, *TRACE* supports studying how preferences accumulate, persist, and evolve in CRS.

3 Dataset Construction

In this section, we describe the construction of *TRACE*, which links chronological viewing histories with long-term, multi-session recommendation dialogues. Figure 2 illustrates the overall dataset construction pipeline. Further preprocessing details are provided in Appendix A.

3.1 Data Preparation

Data Source. We build *TRACE* on MTS-KION (Petrov et al., 2022), a media viewing dataset containing user-level viewing histories, item metadata, and demographic information. We use this dataset because viewing histories provide chronological behavioral signals recorded at the time of consumption, making them more suitable for modeling long-term preference evolution than isolated reviews or ratings. The availability of item metadata also enables event and dialogue generation grounded in the consumed content, while demographic attributes provide auxiliary cues for initializing user interactions.

Trajectory Construction. For each user, we sort viewing logs chronologically to construct a user-level behavioral trajectory. We split each trajectory into two parts. The earliest 20% of viewing logs are used to initialize the user’s behavioral persona, providing an initial estimate of long-term consumption tendencies. The remaining 80% are used to construct temporally ordered recommendation sessions, allowing later dialogues to follow the user’s evolving behavioral history.

Session Construction. We construct each session as a CRS scenario in which the system elicits preferences and recommends an item the user is likely to accept. Items with a completion rate of at least 80% are treated as positive interactions and used as candidate targets. For each candidate session, the last item is set as the target y_t , while earlier items

Metric	TRACE (Ours)	PEARL (Kim et al., 2024)
Credibility	4.23	3.89
Consistency	4.66	4.94
Completeness	4.94	4.63
Clarity	4.87	3.98
Immersion	4.81	4.01

Table 1: PPS metric (Salminen et al., 2020) results of behavioral personas on a 5-point scale.

Method	k-NN dist.	Cosine dist.	Sem-Ent entropy
TRACE (Ours)	0.1513	0.2541	0.8522
PEARL (Kim et al., 2024)	0.1173	0.2363	0.7958

Table 2: Diversity metrics for behavioral personas.

serve as source items X_t for preference elicitation. Adjacent candidate sessions are merged when necessary to ensure at least two viewing records per session and avoid overlapping target dates. This yields temporally ordered sessions that maintain behavioral continuity across interactions.

3.2 Persona Design and Initialization

In *TRACE*, each user is represented by two complementary personas: a behavioral persona and a dialogue persona. The former captures long-term consumption patterns from viewing histories, while the latter captures conversational traits and explicitly expressed preferences from CRS interactions.

Behavioral Persona. To ground generated dialogues in accumulated behavior, we define a behavioral persona p^b that summarizes each user’s long-term consumption patterns. The schema covers positive and negative preferences over content materials and production contexts, such as genre, theme, cast, and crew. It captures usage patterns and decision drivers, such as temporal regularity, consumption intensity, selection priorities, and latent viewing motivations. Grounded in Uses and Gratifications Theory (Katz et al., 1973a; Cha and Chan-Olmsted, 2024), this schema abstracts noisy viewing logs into interpretable user characteristics. The full schema is provided in Appendix A.2.

From the initialization split, we construct p_0^b as the user’s initial preference state for later history-aligned dialogue generation.

Dialogue Persona. While the behavioral persona captures implicit long-term tendencies, it does not fully represent how users express preferences in conversation. We therefore define a dialogue persona p^d that summarizes conversational traits, expressed interests, feedback patterns, and explicit recommendation conditions revealed during dia-

logue. Since no prior dialogue is available before the first session, we initialize p_0^d using demographic information from MTS-KION and predefined CRS interaction styles. The interaction styles vary along three axes: activeness, feedback directness, and preference clarity. As sessions progress, p^d is updated from accumulated dialogue history to reflect evolving conversational behavior and explicitly stated preferences.

3.3 Iterative Dialogue and Persona Evolution

After persona initialization and session construction, *TRACE* generates multi-session dialogues iteratively. At each session, it creates an event, synthesizes a dialogue, and updates both personas with behavioral and conversational signals.

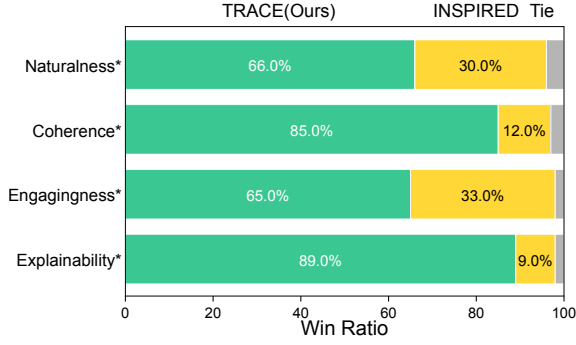
Event Generation. For each session s_t , we generate an event e_t as session-specific context that motivates the user’s CRS interaction. The event is derived from item metadata and demographic information and conditioned on the previous event e_{t-1} to maintain situational coherence across sessions.

Dialogue Generation. Given the event e_t , we synthesize a dialogue using GPT-4o (OpenAI, 2024) as $\text{LLM}(\cdot)$, conditioned on the previous behavioral persona p_{t-1}^b , dialogue persona p_{t-1}^d , source items X_t , target item y_t , and compressed cross-session history $h_{<t}$:

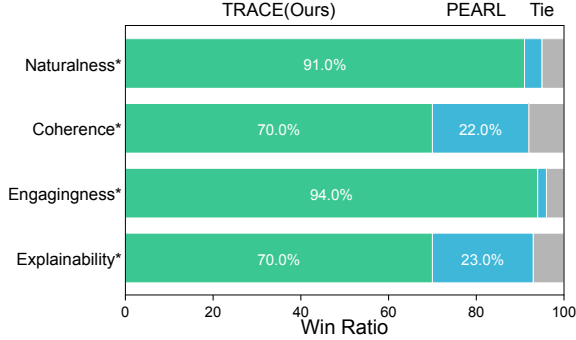
$$D_t = \text{LLM}(p_{t-1}^b, p_{t-1}^d, e_t, X_t, y_t, h_{<t}),$$

where $h_{<t}$ summarizes previous events and recommendation outcomes. Each dialogue begins with event-grounded chitchat, proceeds to preference exploration using source items, and culminates in recommending the target item. Thus, LLM-generated dialogues remain grounded in temporally ordered behavior and evolving personas rather than independent conversations.

Persona Update. After each session, both personas are updated to model gradual user evolution. For p_t^b , we extract preference and consumption-pattern signals from a sliding window over the current and recent k sessions. For p_t^d , we extract conversational signals from the current dialogue, including interaction style, expressed interests, feedback patterns, and recommendation conditions. These signals are incorporated into the previous personas, allowing *TRACE* to represent users as evolving preference states across sessions.



(a) *TRACE* (Ours) vs. *INSPIRED* (Hayati et al., 2020).



(b) *TRACE* (Ours) vs. *PEARL* (Kim et al., 2024).

Figure 3: Human evaluation results comparing conversations from *TRACE* with those from existing CRS datasets, *INSPIRED* and *PEARL*. (*: $p < 0.05$)

3.4 Dataset Analysis

Dataset Statistics. Table 7 summarizes the basic statistics of *TRACE*. *TRACE* contains 1,136 users and 8,168 dialogue sessions, with 7.19 sessions per user and 12.97 utterances per session on average. This provides temporally connected trajectories for evaluating cross-session preference modeling and within-session adaptation.

Persona Quality Analysis. We evaluate whether the constructed behavioral personas are both high-quality and diverse. For persona quality, we adapt and reorganize criteria from the Persona Perception Scale (PPS) (Salminen et al., 2020) into five LLM-as-a-judge evaluation dimensions: credibility, consistency, completeness, clarity, and immersion. These dimensions measure whether a persona is realistic, internally coherent, sufficiently informative, easy to interpret, and vivid enough to support a concrete understanding of the user. Since personas are generated using GPT-4o, we conduct PPS evaluation with Gemini-2.5 Pro (Gemini Team, 2025) to enable cross-model assessment and reduce evaluator-model bias.

As shown in Table 1, our schema-based personas achieve higher overall PPS scores than a *PEARL*-

style like/dislike baseline (Kim et al., 2024), indicating more informative representations for personalization. Although *TRACE* has a slightly lower consistency score, it remains high in absolute terms, suggesting that evolving personas add complexity without substantially reducing persona quality.

We also evaluate persona diversity using k-NN distance, cosine distance, and *Sem-Ent* entropy (Han et al., 2022), with embeddings computed by Qwen3-Embedding-0.6B (Zhang et al., 2025). These metrics respectively capture local diversity, average semantic dissimilarity, and the semantic spread of persona contents.

As shown in Table 2, *TRACE* achieves higher diversity, indicating that our schema-based personas capture richer user variation for personalization.

Dialogue Quality Evaluation. We evaluate dialogue quality through human judgments on 100 sampled dialogue pairs for each baseline comparison. Annotators compare dialogues along four dimensions: naturalness, coherence, engagingness, and explainability. Details on the human evaluation setup are provided in Appendix B.

As shown in Figure 3, *TRACE* outperforms existing CRS datasets across all evaluation dimensions. These results suggest that *TRACE* provides high-quality dialogues for learning and evaluating long-term multi-session CRS models.

4 Proposed Framework

To demonstrate the utility of *TRACE* for long-term conversational recommendation, we develop *EvoCRS*, a persona-based CRS framework that uses evolving behavioral and dialogue personas across sessions. While many prior CRS methods use user histories mainly as static or retrieved evidence, *EvoCRS* explicitly maintains persona states that accumulate and change over time. We use this framework to examine whether the evolving multi-view signals in *TRACE* improve long-term user modeling and recommendation.

As illustrated in Figure 4, *EvoCRS* operates in two steps for each session. First, it performs persona-aware recommendation using the personas updated through the previous session. Then, after the session, it updates both personas using behavioral and conversational signals from the session. Prompt details are provided in Appendix F.

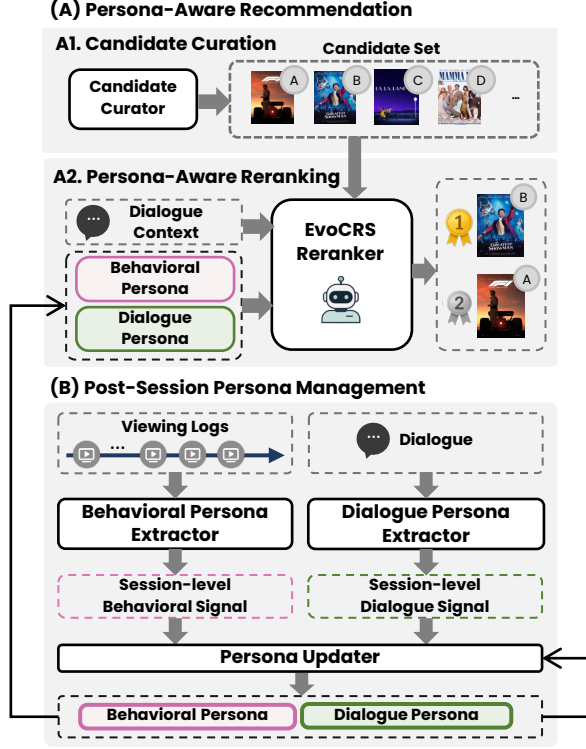


Figure 4: Architecture of *EvoCRS*, which performs persona-aware recommendation and post-session persona updates using behavioral and dialogue signals.

4.1 Persona-Aware Recommendation

At session t , *EvoCRS* recommends an item using the current dialogue context and the personas updated through the previous session, p_{t-1}^b and p_{t-1}^d . To handle large item spaces, recommendation is performed through candidate curation followed by persona-aware reranking.

Candidate Curation. Directly applying an LLM-based ranker to the full item space is costly and constrained by context length. Following prior work that applies LLMs at the ranking stage over retrieved candidates, we use a candidate curator to retrieve a compact item set before reranking (Hou et al., 2024). The curator can be instantiated with any efficient recommendation or retrieval model.

Persona-Aware Reranking. Given the curated candidate set, *EvoCRS* reranks candidates by jointly considering the current dialogue context, behavioral persona, and dialogue persona. The ranker is trained to identify the target item from the candidate set, allowing recommendations to reflect both accumulated long-term preferences and immediate conversational intent.

Inspired by FIRST, which combines single-token scoring with a pairwise learning-to-rank objective (Reddy et al., 2024), the persona-aware ranker

is trained with a cross-entropy objective for target prediction and a pairwise ranking objective that encourages the target item to score higher than other candidates:

$$\mathcal{L}_{\text{ranker}} = \mathcal{L}_{\text{CE}} + \lambda_{\text{rank}} \cdot \mathcal{L}_{\text{rank}}, \quad (1)$$

where $\mathcal{L}_{\text{CE}}$ and $\mathcal{L}_{\text{rank}}$ denote the target prediction and pairwise ranking losses, respectively, and λ_{rank} controls the ranking loss weight.

4.2 Post-Session Persona Management

After each recommendation session, *EvoCRS* updates the user’s personas to track long-term preference evolution. It extracts two types of session-level signals: behavioral signals from the session’s interaction records and dialogue signals from the current conversation. Behavioral signals summarize implicit preferences and consumption patterns, while dialogue signals capture explicitly stated needs, constraints, feedback, and interests.

The LLM-based persona updater integrates these signals into the previous personas. Consistent evidence reinforces existing preferences, new non-conflicting evidence is added, and conflicting evidence is resolved by prioritizing more recent observations. Through this update process, *EvoCRS* maintains evolving behavioral and dialogue personas for later sessions.

5 Experiments

In this section, we evaluate whether *TRACE* supports long-term user modeling in CRS through the *EvoCRS* framework. We conduct experiments to answer three research questions: **(RQ1)** whether *EvoCRS* improves long-term CRS performance over existing methods; **(RQ2)** how each persona type contributes to recommendation performance; and **(RQ3)** whether persona updates across sessions improve long-term recommendation.

5.1 Experimental Setup

Baselines. We compare *EvoCRS* with traditional CRS baselines, *ReDial* (Li et al., 2018) and *KBRD* (Chen et al., 2019), as well as *MemoCRS* (Xi et al., 2024), an LLM-based sequential CRS baseline with memory over previous conversational evidence. For a controlled comparison, we reimplement *MemoCRS* using the same reranker backbone and candidate setting as *EvoCRS*. We also report *Candidate Curation* to evaluate performance before persona-aware reranking.

Model	Recall@1	Recall@5	Recall@10	MRR	NDCG@1	NDCG@5	NDCG@10
<i>Traditional CRS baselines</i>							
<i>ReDial</i> (Li et al., 2018)	0.0030	0.0135	0.0195	0.0097	0.0030	0.0082	0.0100
<i>KBRD</i> (Chen et al., 2019)	0.0075	0.0225	0.0375	0.0175	0.0075	0.0156	0.0204
<i>Reranker Backbone: Llama-3.1-8B-Instruct</i>							
<i>Candidate Curation</i>	0.0616	0.1652	0.2117	0.1146	0.0616	0.1154	0.1305
<i>MemoCRS</i> (Xi et al., 2024)	0.1171	0.2492	0.3033	0.1776	0.1171	0.1854	0.2026
<i>EvoCRS (Ours)</i>	0.1366	0.2598	0.3108	0.1938	0.1366	0.2002	0.2165
<i>Reranker Backbone: Qwen2.5-7B-Instruct</i>							
<i>Candidate Curation</i>	0.0616	0.1652	0.2117	0.1146	0.0616	0.1154	0.1305
<i>MemoCRS</i> (Xi et al., 2024)	0.0991	0.2312	0.3198	0.1619	0.0991	0.1653	0.1938
<i>EvoCRS (Ours)</i>	0.1066	0.2508	0.3093	0.1738	0.1066	0.1816	0.2006

Table 3: Overall recommendation performance comparison across different backbone models.

Model	Recall@1	Recall@5	Recall@10	MRR	NDCG@1	NDCG@5	NDCG@10
<i>Reranker Backbone: Llama-3.1-8B-Instruct</i>							
<i>EvoCRS</i>	0.1366	0.2598	0.3108	0.1938	0.1366	0.2002	0.2165
<i>w/o Behavioral Persona</i>	<u>0.1186</u>	<u>0.2553</u>	0.3048	<u>0.1858</u>	<u>0.1186</u>	<u>0.1932</u>	<u>0.2091</u>
<i>w/o Dialogue Persona</i>	<u>0.1186</u>	0.2523	<u>0.3093</u>	0.1817	<u>0.1186</u>	0.1882	0.2068
<i>w/o Both Personas</i>	0.1141	0.2372	0.2973	0.1741	0.1141	0.1777	0.1973
<i>Reranker Backbone: Qwen2.5-7B-Instruct</i>							
<i>EvoCRS</i>	0.1066	0.2508	<u>0.3093</u>	0.1738	0.1066	0.1816	0.2006
<i>w/o Behavioral Persona</i>	0.1006	<u>0.2447</u>	0.3168	<u>0.1653</u>	0.1006	<u>0.1722</u>	<u>0.1956</u>
<i>w/o Dialogue Persona</i>	<u>0.1021</u>	0.2237	0.3033	0.1645	<u>0.1021</u>	0.1660	0.1911
<i>w/o Both Personas</i>	0.0991	0.2222	0.3048	0.1630	0.0991	0.1640	0.1908

Table 4: Ablation study on the effect of behavioral and dialogue personas across different backbone models.

Evaluation Metrics. We evaluate recommendation performance using Mean Reciprocal Rank (MRR), Recall@K, and Normalized Discounted Cumulative Gain (NDCG@K), where $K \in \{1, 5, 10\}$.

Evaluation Protocol. Throughout evaluation, we use only the dialogue prefix preceding the first mention of the target item. The final target recommendation, the user’s acceptance, and all subsequent turns are excluded from the input. For reranking-based methods, candidate retrieval produces a top-50 candidate set for subsequent ranking. All test sessions are retained regardless of retrieval success; failures to retrieve the target within the top-50 candidates are counted as incorrect to ensure pipeline-level evaluation.

Implementation Details. We implement the *EvoCRS* ranker using instruction-tuned LLM backbones, namely Llama-3.1-8B-Instruct (Llama Team, 2024) and Qwen2.5-7B-Instruct (Yang et al., 2025). In our implementation, we instantiate the candidate curator with Qwen3-Embedding-8B (Zhang et al., 2025). Following Yang and Chen (2024), we train the curator with a contrastive retrieval objective that scores the ground-truth item higher than in-batch negatives given the

conversation context. We split *TRACE* into training, validation, and test sets with an 8:1:1 ratio at the user level. All models are fine-tuned on the *TRACE* training split and evaluated on the same test split under an identical candidate setting.

5.2 Experimental Results

Answer to RQ1: *EvoCRS* improves long-term conversational recommendation over baselines.

Table 3 reports the recommendation performance. *EvoCRS* achieves the strongest overall performance with both backbones, although *MemoCRS* obtains a higher Recall@10 with the Qwen backbone. Both *MemoCRS* and *EvoCRS* outperform the curation-only setting, showing that reranking adds value beyond candidate retrieval. In particular, *EvoCRS*’s gains over *MemoCRS*, the long-term CRS baseline, suggest that maintaining evolving personas is more effective than using retrieved memory summaries alone. Overall, these results show that *EvoCRS* effectively uses the persona signals provided by *TRACE* for conversational recommendation.

Answer to RQ2: Behavioral and dialogue personas provide complementary signals. We conduct an ablation study by removing the behavioral persona, the dialogue persona, or both from

Model	Session 2	Session 3	Session 4	Session 5	Session 6	Session 7	Session 8
Reranker Backbone: Llama-3.1-8B-Instruct							
<i>MemoCRS</i> (Xi et al., 2024)	0.1461	0.1404	0.1915	0.1870	0.1837	<u>0.2011</u>	0.2397
<i>EvoCRS</i> w/ <i>Static Persona</i>	0.1772	0.1343	0.1687	<u>0.1942</u>	<u>0.1844</u>	0.1875	<u>0.2459</u>
<i>EvoCRS</i> w/ <i>Evolving Persona</i>	0.1772	<u>0.1403</u>	<u>0.1890</u>	0.2209	0.1994	0.2090	0.2707
Reranker Backbone: Qwen2.5-7B-Instruct							
<i>MemoCRS</i> (Xi et al., 2024)	0.1370	0.1325	0.1549	0.2209	<u>0.1560</u>	0.1414	0.2001
<i>EvoCRS</i> w/ <i>Static Persona</i>	0.1470	<u>0.1276</u>	<u>0.1685</u>	<u>0.2096</u>	0.1761	<u>0.1559</u>	<u>0.2227</u>
<i>EvoCRS</i> w/ <i>Evolving Persona</i>	0.1470	0.1240	0.1724	<u>0.2155</u>	0.1545	0.1919	0.2404

Table 5: Session-wise MRR comparison of static and evolving personas across backbone models.

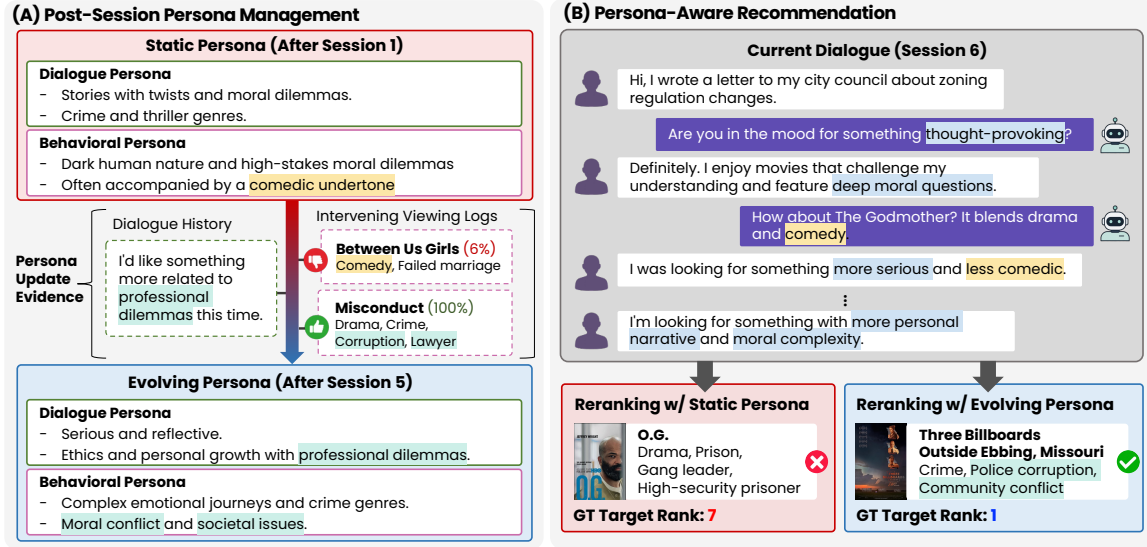


Figure 5: Case study of persona management and persona-aware reranking. Evolving personas incorporate prior dialogue and intervening viewing logs, allowing *EvoCRS* to identify the user’s preference trajectory and promote the ground-truth target under an ambiguous request.

EvoCRS; the variant with both personas removed serves as a dialogue-only control. As shown in Table 4, removing either persona generally degrades performance, indicating that both provide useful signals. Behavioral personas capture implicit preferences from viewing behaviors, whereas dialogue personas capture explicitly expressed needs from conversations. The larger degradation without both personas suggests that the two views provide complementary information for long-term user modeling beyond the current dialogue alone.

Answer to RQ3: Cross-session persona updates improve long-term preference modeling. We evaluate long-term multi-session recommendation by tracking MRR from the 2nd to the 8th session of each user trajectory. Each model predicts the session target using previous-session information and the dialogue context. We compare *MemoCRS* with two *EvoCRS* variants: *Static Persona* uses the first-session persona without updates, while *Evolving Persona* updates personas after each session.

As shown in Table 5, *MemoCRS* remains competitive in some early sessions, suggesting that memory summaries capture recent preference evidence. However, *Evolving Persona* generally performs better in later sessions, especially Sessions 7 and 8, where updated personas better reflect long-term preference changes. This indicates that *TRACE* models meaningful cross-session preference evolution and that explicit persona updates capture changing user states beyond memory summaries.

5.3 Case Study

Figure 5 shows how evolving personas resolve ambiguity in long-term recommendation. In session 6, the user requests a serious, thought-provoking film with moral complexity, but this request remains underspecified. Using the current dialogue and static persona, *EvoCRS* ranks the ground-truth target, “Three Billboards Outside Ebbing, Missouri,” in the seventh place. With evolving personas incorporating the prior-session dialogue and intervening

viewing logs, *EvoCRS* captures more specific interests in moral conflict and societal issues, promoting the target to the top rank. This case suggests that persona evolution helps more accurately identify the specific preference trajectory behind an ambiguous recommendation need.

6 Conclusion

We introduced *TRACE*, a long-term multi-session CRS dataset that grounds LLM-generated dialogues in temporally ordered viewing histories and evolving personas. We developed *EvoCRS*, a persona-based framework for validating *TRACE* in long-term conversational recommendation. Our analyses show that *TRACE* provides reliable personas and high-quality CRS dialogues, while experiments show that persona updates improve recommendation performance. These findings highlight the importance of structurally tracking preference evolution across interactions, rather than relying on past dialogues as retrieved evidence.

Limitations

TRACE is grounded in real temporally ordered viewing histories, but the associated dialogues and persona updates are synthetically constructed with an LLM. Thus, the evolving personas in *TRACE* should be understood as structured representations of behavior-grounded conversational trajectories, rather than direct observations of how real users' preferences and conversational behavior evolve over time. Although our quality analyses and downstream experiments demonstrate the utility of these trajectories for long-term CRS research, future work should validate whether similar benefits hold in real longitudinal user interactions.

Ethical Considerations

TRACE is constructed from viewing histories and demographic attributes in the source dataset, and the generated personas and dialogues may inherit biases present in these signals. The personas are intended as research representations for studying long-term conversational recommendation, rather than factual psychological profiles of individual users. Accordingly, *TRACE* and *EvoCRS* should not be used for deployment-level profiling or for decisions affecting users without further privacy and fairness assessment. Future work should examine these issues when applying evolving persona modeling in real user-facing systems.

Acknowledgement

This work was the result of project supported by the KT (Korea Telecom)–Korea University AICT R&D Center. This work was also partly supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (No. IITP-2026RS-2020-II201819, the ICT Creative Consilience Program, 20%; No. RS-2026-25585074, Development of Agentic AI Technology for Enterprise-wide Business Innovation Based on Goal-Oriented Long-term Context Reasoning, 50%), and by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. RS-2025-24533089, 20%).

References

- Jiyoung Cha and Sylvia Chan-Olmsted. 2024. [Binge-watching behaviors: Conceptualization, motives, demographics, and program genre preferences](#). *International Journal on Media Management*, 26(3-4):117–143.
- Qibin Chen, Junyang Lin, Yichang Zhang, Ming Ding, Yukuo Cen, Hongxia Yang, and Jie Tang. 2019. [Towards knowledge-based recommender dialog system](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1803–1813.
- Gemini Team. 2025. [Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities](#). *arXiv preprint arXiv:2507.06261*.
- Seungju Han, Beomsu Kim, and Buru Chang. 2022. [Measuring and improving semantic diversity of dialogue generation](#). In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 934–950.
- Shirley Anugrah Hayati, Dongyeop Kang, Qingxi-aoyang Zhu, Weiyan Shi, and Zhou Yu. 2020. [INSPIRED: Toward sociable recommendation dialog systems](#). In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 8142–8152.
- Zhankui He, Zhouhang Xie, Rahul Jha, Harald Steck, Dawen Liang, Yesu Feng, Bodhisattwa Prasad Majumder, Nathan Kallus, and Julian McAuley. 2023. [Large language models as zero-shot conversational recommenders](#). In *Proceedings of the 32nd ACM international conference on information and knowledge management*, pages 720–730.
- Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao.

2024. [Large language models are zero-shot rankers for recommender systems](#). In *Advances in Information Retrieval: 46th European Conference on Information Retrieval, ECIR 2024*, pages 364–381.
- Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. [Collaborative filtering for implicit feedback datasets](#). In *2008 Eighth IEEE international conference on data mining*, pages 263–272. Ieee.
- Dietmar Jannach, Ahtsham Manzoor, Wanling Cai, and Li Chen. 2021. [A survey on conversational recommender systems](#). *ACM Computing Surveys (CSUR)*, 54(5):1–36.
- Dongyeop Kang, Anusha Balakrishnan, Pararth Shah, Paul Crook, Y-Lan Boureau, and Jason Weston. 2019. [Recommendation as a communication game: Self-supervised bot-play for goal-oriented dialogue](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1951–1961.
- Elihu Katz, Jay G Blumler, and Michael Gurevitch. 1973a. [Uses and gratifications research](#). *The public opinion quarterly*, 37(4):509–523.
- Elihu Katz, Hadassah Haas, and Michael Gurevitch. 1973b. [On the use of the mass media for important things](#). *American sociological review*, 38(2):164–181.
- Minjin Kim, Minju Kim, Hana Kim, Beong-woo Kwak, SeongKu Kang, Youngjae Yu, Jinyoung Yeo, and Dongha Lee. 2024. [PEARL: A review-driven persona-knowledge grounded conversational recommendation dataset](#). *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1105–1120.
- Heejin Kook, Junyoung Kim, Seongmin Park, and Jongwuk Lee. 2025. [Empowering retrieval-based conversational recommendation with contrasting user preferences](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7692–7707.
- Ivica Kostic, Krisztian Balog, and Filip Radlinski. 2021. [Soliciting user preferences in conversational recommender systems via usage-related questions](#). In *Proceedings of the 15th acm conference on recommender systems*, pages 724–729.
- Raymond Li, Samira Ebrahimi Kahou, Hannes Schulz, Vincent Michalski, Laurent Charlin, and Chris Pal. 2018. Towards deep conversational recommendations. *Advances in neural information processing systems*, 31.
- Shuokai Li, Ruobing Xie, Yongchun Zhu, Xiang Ao, Fuzhen Zhuang, and Qing He. 2022. [User-centric conversational recommendation with multi-aspect user modeling](#). In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 223–233.
- Tingting Liang, Chenxin Jin, Lingzhi Wang, Wenqi Fan, Congying Xia, Kai Chen, and Yuyu Yin. 2024. [LLM-REDIAL: A large-scale dataset for conversational recommender systems created from user behaviors with llms](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 8926–8939.
- Zeming Liu, Haifeng Wang, Zheng-Yu Niu, Hua Wu, Wanxiang Che, and Ting Liu. 2020. [Towards conversational recommendation over multi-type dialogs](#). In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 1036–1049.
- Llama Team. 2024. [The Llama 3 herd of models](#). *arXiv preprint arXiv:2407.21783*.
- Alessandro Francesco Maria Martina, Alessandro Petruzzelli, Cataldo Musto, Marco de Gemmis, Pasquale Lops, and Giovanni Semeraro. 2025. [Dis-tillRecDial: A knowledge-distilled dataset capturing user diversity in conversational recommendation](#). In *Proceedings of the Nineteenth ACM Conference on Recommender Systems*, pages 726–735.
- OpenAI. 2024. [GPT-4o system card](#). *arXiv preprint arXiv:2410.21276*.
- Aleksandr Petrov, Ildar Safilo, Daria Tikhonovich, and Dmitry Ignatov. 2022. [MTS KION implicit contextualised sequential dataset for movie recommendation](#). *arXiv preprint arXiv:2209.00325*.
- Revanth Gangi Reddy, JaeHyeok Doo, Yifei Xu, Md Arafat Sultan, Deevya Swain, Avirup Sil, and Heng Ji. 2024. [FIRST: Faster improved listwise reranking with single token decoding](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8642–8652.
- Joni Salminen, Joao M Santos, Haewoon Kwak, Jisun An, Soon-gyo Jung, and Bernard J Jansen. 2020. [Persona perception scale: development and exploratory validation of an instrument for evaluating individuals’ perceptions of personas](#). *International Journal of Human-Computer Studies*, 141:102437.
- Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. [Is ChatGPT good at search? investigating large language models as re-ranking agents](#). In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 14918–14937.
- Yunjia Xi, Weiwen Liu, Jianghao Lin, Bo Chen, Ruiming Tang, Weinan Zhang, and Yong Yu. 2024. [Mem-oCRS: Memory-enhanced sequential conversational recommender systems with large language models](#). In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 2585–2595.

- 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, and 23 others. 2025. [Qwen2.5 technical report](#). *Preprint*, arXiv:2412.15115.
- Ting Yang and Li Chen. 2024. [Unleashing the retrieval potential of large language models in conversational recommender systems](#). In *Proceedings of the 18th ACM Conference on Recommender Systems*, pages 43–52.
- Christian Zabel, Louis Schaffeld, and Daniel O’Brien. 2025. [Netflix and chill? the content-related and gratificational antecedents of binge-watching tendency](#). *Studies in Communication and Media*, 14(1):105–139.
- Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. [Qwen3 embedding: Advancing text embedding and reranking through foundation models](#). *arXiv preprint arXiv:2506.05176*.
- Kun Zhou, Yuanhang Zhou, Wayne Xin Zhao, Xiaoke Wang, and Ji-Rong Wen. 2020. [Towards topic-guided conversational recommender system](#). In *Proceedings of the 28th international conference on computational linguistics*, pages 4128–4139.

A Dataset Construction Details

We use GPT-4o, a strong instruction-following model (OpenAI, 2024), to support the construction of our multi-session conversational recommendation dataset. Specifically, GPT-4o is used for event generation, persona extraction, persona updating, and dialogue generation. We use a random seed of 42 for stochastic procedures where applicable. The complete *TRACE* construction cost approximately USD 900 and was executed with 50 parallel workers, requiring approximately 8 hours in total. The prompts used for dataset construction are provided in Appendix F.

A.1 Dataset Source Preprocessing

We construct our dataset from MTS-KION (Petrov et al., 2022), a real-world VOD platform log dataset collected over a five-month period from March 13 to August 22, 2021. MTS-KION records implicit user-item interactions at viewing time and provides interaction-level information such as timestamps and watched percentage. To observe long-term preference dynamics, we retain active users whose viewing histories span at least 20 days. This filtering yields 1,695 users, from which we randomly sample 1,200 users and limit each trajectory to at most 10 sessions due to generation cost. We then sort each user’s interactions chronologically and construct multi-session trajectories from the resulting sequence. For preference modeling, we use watched percentage as the implicit feedback signal, treating it as an indicator of preference strength.

A.2 Behavioral Persona Extraction

We extract behavioral personas in an attribute-wise manner to improve extraction quality while maintaining consistency across attributes. Table 6 summarizes the behavioral persona schema used in *TRACE*.

Theoretical Rationale. Prior persona-based CRS approaches primarily represent users through positive and negative preferences (Kim et al., 2024; Kook et al., 2025), whereas our schema additionally incorporates usage patterns and decision drivers. These components are motivated by Uses and Gratifications Theory (UGT) and related studies showing associations between media-consumption motivations, genre preferences, and viewing behaviors such as binge-watching (Zabel et al., 2025; Cha and Chan-Olmsted, 2024). Among these attributes, the motivation component of deci-

Category	Attribute	Description
Positive preferences	Content material	Preferred genres, themes, and topics
	Production context	Preferred cast, crew, and production-related metadata
Negative preferences	Content material	Avoided genres, themes, and topics
	Production context	Avoided cast, crew, and production-related metadata
Usage patterns	Temporal patterns	Timing and regularity of content consumption
	Immersion style	Consumption intensity, volume, and completion tendency
Decision drivers	Selection priority	Criteria that guide content selection
	Motivation	Latent viewing intent inferred from behavior

Table 6: Behavioral persona schema used in *TRACE*.

sion drivers is most directly connected to UGT, as it represents functional gratifications underlying media consumption. For example, seeking cognitively demanding content can be interpreted through the cognitive-needs dimension of UGT, which characterizes media use for intellectual stimulation and knowledge acquisition (Katz et al., 1973b).

Extraction Procedure. For content-material and production-context preferences, positive and negative descriptions are jointly extracted within each attribute. This is because they are defined over the same evidence space and should remain mutually consistent.

For usage patterns, we extract temporal patterns and immersion style separately because they rely on different behavioral signals. Temporal patterns capture when and how regularly the user consumes content, whereas immersion style captures viewing intensity, volume, and completion tendency.

For decision drivers, we extract selection priority and motivation using the previously extracted preference attributes as input. This allows the persona to reflect both the preference signals that directly guide content selection and the user’s latent viewing intent.

A.3 Dialogue Persona Initialization

We initialize the dialogue persona using demographic attributes and randomly assigned CRS interaction styles. For each user, age, gender, and parental status are taken directly from the metadata provided in MTS-KION. The CRS interaction style is randomly assigned to introduce diversity in how users express preferences, provide feedback, and interact with the recommender.

For occupation, which is not directly available in MTS-KION, we use occupation information from the 2021 wave of the Russia Longitudinal Monitoring Survey–Higher School of Economics (RLMS-

HSE).² Specifically, using age, gender, and income information available in MTS-KION, we match each user to the corresponding demographic group in the 2021 RLMS-HSE data and randomly sample an occupation from individuals in that group. This allows the initial dialogue persona to reflect plausible demographic variation while avoiding deterministic assignment from a single attribute.

These initialized attributes are used only to construct the initial conversational profile before any dialogue history is available. As sessions progress, the dialogue persona is updated using accumulated conversational evidence.

A.4 Event Generation

For each session, we generate an event as session-specific context that motivates the user’s conversational recommendation request. The event is derived from item metadata and demographic information, and, for later sessions, conditioned on the previous event to maintain situational coherence across the user’s trajectory.

We implement event generation in two stages: candidate event generation and final selection. In the candidate generation stage, we abstract high-level contextual cues from the item metadata, such as a broadly relevant mood, theme, or situation, and use them to generate multiple plausible real-life scenarios. The generated scenarios are designed to provide a natural motivation for initiating a recommendation conversation, rather than directly describing a specific item. For sessions after the first session, the preceding event is additionally provided so that newly generated scenarios remain coherent with earlier conversational contexts.

In the final selection stage, we select and refine one candidate event based on three criteria: plausibility as an everyday situation, compatibility with the user’s demographic information, and coherence with preceding events. The selected event is then used as contextual grounding for dialogue generation, enabling each session to be connected to the user’s temporally ordered trajectory while preserving a natural conversational setting.

²Source: “Russia Longitudinal Monitoring Survey, RLMS HSE,” conducted by National Research University “Higher School of Economics” and OOO “Demoscope” together with the Institute of Sociology of the Federal Center of Theoretical and Applied Sociology of the Russian Academy of Sciences. RLMS HSE website: <https://www.hse.ru/rlms>

Statistic	Value
# of users	1,136
# of dialogues	8,168
# of utterances	105,922
# of unique items	4,588
Avg. dialogues per user	7.19
Avg. utterances per dialogue	12.97
Avg. source items per dialogue	1.74
Avg. viewing logs per session	18.76

Table 7: Dataset statistics of *TRACE*. Each dialogue contains one target item, and items include both movies and series.

A.5 Manual Dialogue Filtering

After dialogue generation, we manually inspected the generated dialogues and removed cases that violated the intended dialogue format, including premature target mentions, to maintain dialogue quality and prevent target leakage in downstream evaluation. After filtering, the final dataset contains 1,136 users and 8,168 dialogue sessions.

A.6 Dataset Statistics

Table 7 presents the detailed statistics of *TRACE*. The dataset contains 1,136 user-level trajectories comprising 8,168 temporally ordered dialogue sessions and 105,922 utterances in total. Each dialogue session is associated with one target item and an average of 1.742 source items derived from the corresponding viewing history, resulting in 4,588 unique items across the dataset. On average, each user trajectory contains 7.19 dialogue sessions, and each dialogue contains 12.97 utterances. In addition, each session is grounded in an average of 18.76 viewing logs used to construct its behavioral context. These statistics show that *TRACE* provides multi-session conversational trajectories aligned with substantial temporally ordered behavioral evidence, supporting the study of long-term preference modeling in CRS.

B Human Evaluation

We provide additional details on the human evaluations. For dialogue quality evaluation, we randomly sample 100 dialogue pairs for each baseline comparison, and for response personalization evaluation, we sample 100 response pairs. Each pair is independently evaluated by three annotators. System identities are anonymized, and the two outputs are presented in randomized A/B order. Annotators may select Tie when neither output is clearly preferable. All annotators were undergraduate or

graduate students who participated voluntarily and received no monetary compensation.

The preference rates reported in the main results are computed over individual annotations. For significance testing, the three annotations for each item are first aggregated by majority vote. Items with a Tie majority or no majority are excluded from the primary test, and we apply an exact two-sided binomial test with $p = 0.5$ to the remaining decisive items. We additionally conduct a conservative all-item analysis in which Tie-majority and no-majority items are counted as non-wins for the evaluated method. This analysis uses an exact one-sided binomial test to determine whether the win probability exceeds 0.5.

We report inter-annotator agreement using pairwise agreement and unanimous agreement, treating Tie as a valid third response category. Pairwise agreement measures the proportion of agreeing annotator pairs, while unanimous agreement measures the proportion of items for which all three annotators select the same category.

Screenshots of the evaluation interfaces are shown in Figure 6.

B.1 Dialogue Quality Evaluation

For dialogue quality evaluation, annotators compare our constructed dialogues with baseline dialogues. Each dialogue pair is evaluated along four criteria: naturalness, coherence, engagingness, and explainability. The criteria are defined as follows:

- **Naturalness:** Which dialogue sounds more natural and human-like in its wording and conversational style?
- **Coherence:** Which dialogue has a more logical and consistent flow toward the recommendation?
- **Engagingness:** Which dialogue feels more interactive and enjoyable, making the user feel involved in the recommendation process?
- **Explainability:** Which dialogue better explains why the recommendation fits the user by connecting it to the user’s inferred preferences and current request?

As reported in the main paper, *TRACE* is preferred over both baselines across all four criteria. Both significance analyses—the primary item-level

Baseline	Metric	Pairwise	Unanimous
INSPIRED	Naturalness	0.483	0.240
	Coherence	0.717	0.590
	Engagingness	0.527	0.300
	Explainability	0.763	0.650
PEARL	Naturalness	0.773	0.660
	Coherence	0.633	0.470
	Engagingness	0.813	0.730
	Explainability	0.627	0.470

Table 8: Inter-annotator agreement for dialogue quality evaluation.

binomial test on decisive items and the conservative all-item binomial test—yield the same conclusion across all eight comparisons, with all results remaining statistically significant ($p < 0.05$).

We additionally report inter-annotator agreement for each comparison in Table 8.

B.2 Response Personalization Evaluation

We further evaluate whether persona information improves the personalization of recommendation responses. For this evaluation, we train two response generation models based on LLaMA-3.1-8B-Instruct. Both models are trained to generate the final recommendation utterance from the same dialogue context, but they differ in whether persona information is provided as input. The persona-aware model receives the dialogue context together with the user’s persona, whereas the non-persona model receives only the dialogue context.

For each sampled conversation, we generate two recommendation responses, one from the persona-aware model and one from the non-persona model. Annotators are asked to choose which response is more personalized. A better response should reflect the user’s dialogue persona, behavioral persona, expressed needs, constraints, mood, and preferences in a natural way.

As shown in Table 9, the persona-aware model is preferred in 61.0% of judgments, while the non-persona model is preferred in 31.0%, with 8.0% judged as Tie. These percentages are computed over individual annotator judgments. Both the primary item-level binomial test on decisive items and the conservative all-item binomial test confirm that the preference for persona-aware responses is statistically significant ($p < 0.05$). Pairwise agreement is 0.727, and unanimous agreement is 0.600.

Metric	w/ Persona	w/o Persona	Tie
Personalization	61.0%	31.0%	8.0%

Table 9: Human evaluation results on response personalization.

C Plausibility Analysis of Synthetic Persona Trajectories

To assess whether the synthetic persona trajectories remain grounded in the underlying behavioral evidence, we manually inspected five randomly sampled users, covering 38 persona updates. Across these trajectories, persona changes were generally gradual, reinforcing or refining existing preferences rather than abruptly replacing them. This pattern is consistent with our sliding-window update design, which integrates recent behavioral signals into the existing persona state.

As a representative example, one user showed very low engagement with *Dislike* (Drama, 2017; 3% watched), whose metadata included keywords such as “dysfunctional family,” “divorce,” and “missing child.” Correspondingly, the user’s negative preference was refined from “low engagement with films that explore themes of domestic life, friendship, and interpersonal conflicts” to “low engagement with stories exploring family dysfunction, marital discord, and mundane aspects of everyday life.”

In the same behavioral window, the user showed relatively high engagement with multiple action-oriented items, including *Time Trap* (Action/Science Fiction/Adventure; 49%), *Killer Elite* (Action/Thriller; 51%), and *The Triple Threat* (Action/Thriller; 100%). The corresponding positive preference was updated from “engages with content that features action-packed adventures, often set in fantastical or historical contexts” to “consistently engages with action-packed adventures that blend elements of fantasy and science fiction.” These examples illustrate that the observed persona updates were directionally consistent with the behavioral signals in the corresponding viewing windows.

D Failure Case Analysis of EvoCRS

We further analyze cases where *EvoCRS*, which uses both behavioral and dialogue personas, ranks the target item lower than the *w/o Both Personas* variant. We observe two recurring failure patterns

related to the interaction between accumulated persona information and the user’s current dialogue intent.

Conflict between persona and current intent. In one case, the current dialogue expressed interest in “magic and mythic storytelling,” while the accumulated persona contained negative preferences toward supernatural, fantastical, and mystical content. The target item, *Aladdin*, matched the current dialogue through its fantasy, magic, and adventure elements. However, *EvoCRS* ranked the target lower than the *w/o Both Personas* variant, suggesting that conflicting historical persona information can interfere with the preferences expressed in the current dialogue.

Broad persona overshadowing specific intent. In another case, the current dialogue emphasized “nostalgia” and “personal connections,” whereas the accumulated persona described broader preferences for animation, fantasy, magical adventures, and mythical elements. The target item, *Letters to Juliet*, was aligned with the more specific romance and nostalgia cues expressed in the current dialogue. Again, *EvoCRS* ranked the target lower than the *w/o Both Personas* variant, indicating that broad historical preferences can sometimes overshadow more specific current-session needs.

Overall, these cases illustrate a trade-off in long-term personalization: accumulated persona information can provide useful historical context, but may also conflict with or dilute more specific current intent. This suggests that future long-term CRS models could benefit from mechanisms that dynamically balance historical persona information against session-level preference signals.

E Implementation and Training Details

E.1 Candidate Curation

For candidate curation, we use a simplified retrieval-based approach inspired by prior work on LLM-based conversational recommendation (Yang and Chen, 2024). Specifically, we use Qwen3-Embedding-8B as the backbone encoder and train it to retrieve target items from dialogue contexts.

Each dialogue context is treated as a query, and each item is represented using its textual metadata. Given a dialogue context and its corresponding target item, we encode both with the same encoder and optimize the encoder with an in-batch negative contrastive learning objective. For each mini-batch, the paired target item is treated as the positive item,

while target items paired with other dialogues in the same batch are used as negatives.

After training, we precompute representations for all items in the candidate pool. For each dialogue context, we use its representation as a query and compute similarity scores against all item representations. The top-50 items are selected as curated candidate items for the subsequent ranking process. The candidate curator achieves a Recall@50 of 58.27% on the training split and 42.31% on the test split.

Implementation details. We fine-tune Qwen3-Embedding-8B using LoRA. The retriever is trained for 3 epochs with AdamW, using a per-device batch size of 8 on two GPUs, an initial learning rate of 2×10^{-4} , and a contrastive temperature of 0.05. We evaluate the retriever on the validation set and save checkpoints every 100 training iterations, selecting the checkpoint with the lowest validation objective value. All candidate curation runs are conducted on two NVIDIA A100 80GB GPUs.

E.2 Persona-Aware Reranker

For persona-aware reranking, we implement a listwise ranker following the single-token scoring strategy of FIRST (Reddy et al., 2024). Instead of generating a complete ranked list, we perform a single decoding step and use the output logits of all candidate identifier tokens as their ranking scores.

Sliding-window inference. The candidate curation module first retrieves the top-50 candidate items for each dialogue context. Since each ranker input represents candidates using identifiers from A to T, we rerank the retrieved candidates in windows of 20 items. Following sliding-window reranking strategies used in prior listwise LLM ranking work (Sun et al., 2023), we use a stride of 10, yielding four overlapping windows over the top-50 candidates.

Training instance construction. To construct ranker training instances, we first apply the trained candidate curator to the training sessions and retain sessions whose target item appears among the retrieved top-50 candidates. For each retained session, we form four overlapping windows using the same window size and stride as in inference. When a window does not contain the target item, we remove the last candidate in the window and insert the target item. We then randomly shuffle the can-

didates within each training window to prevent the ranker from exploiting positional patterns associated with the injected target item. At inference time, no target-item injection is performed.

Implementation details. We use Llama-3.1-8B-Instruct and Qwen2.5-7B-Instruct as the ranker backbones. Both models are fine-tuned using QLoRA with 4-bit NF4 quantization and double quantization. We use LoRA rank $r = 64$, scaling factor $\alpha = 16$, and dropout rate 0.1. We train the rankers with DeepSpeed ZeRO Stage 2 in bfloat16 precision, using AdamW with an initial learning rate of 2×10^{-5} and a linear learning-rate scheduler with 100 warm-up steps. The per-device batch size is 2, with 4 gradient accumulation steps and two GPUs per run, resulting in an effective batch size of 16. We evaluate the ranker on the validation set and save checkpoints every 100 training iterations, selecting the checkpoint with the lowest validation objective value. All ranker runs are conducted on two NVIDIA A100 80GB GPUs. We set $\lambda_{\text{rank}} = 10.0$ in all ranker experiments.

E.3 Persona Extraction and Update Models

For each *EvoCRS* backbone, we train behavioral persona extraction, dialogue persona extraction, and persona update models using the same backbone as the corresponding ranker. All persona processing models are fine-tuned using QLoRA with 4-bit NF4 quantization and double quantization. We use LoRA rank $r = 64$, scaling factor $\alpha = 16$, and dropout rate 0.1. Training is performed with DeepSpeed ZeRO Stage 2 in bfloat16 precision, using AdamW with an initial learning rate of 2×10^{-5} , weight decay of 0.01, a WarmupDecayLR scheduler with 100 warm-up steps, and gradient clipping with a maximum norm of 1.0. All models are trained on two NVIDIA A100 80GB GPUs with an effective batch size of 32, using a per-device batch size of 1 and 16 gradient accumulation steps for the behavioral persona extractor, and a per-device batch size of 2 and 8 gradient accumulation steps for the dialogue persona extractor and persona updater. We set the maximum sequence length limit to 13,507 tokens. Validation and checkpoint saving are performed every 200 training iterations for the behavioral persona extractor and every 100 iterations for the dialogue persona extractor and persona updater, and the checkpoint with the lowest validation loss is selected for inference.

E.4 Software Environment and Result Reporting

For the LLM-based experiments, we implement the candidate curator, persona-aware reranker, and persona extraction/update models in Python 3.10 with PyTorch 2.8.0. We use Transformers 4.57.1, Datasets 3.6.0, Accelerate 1.11.0, DeepSpeed 0.18.2, PEFT 0.17.1, BitsAndBytes 0.48.2, vLLM 0.11.0, and FlashAttention 2.8.3. For CUDA kernels, FlashAttention is compiled with TORCH_CUDA_ARCH_LIST=8.0 for NVIDIA A100 GPUs. All LLM-based results are reported from a single run on a fixed user-level test split. All LLM-based methods are evaluated using the same train/validation/test split and the same 50-candidate evaluation setting.

F Full Prompts

Instruction description

You generate plausible ‘day-of’ events for a user using universal seeds inspired by a movie or series they watched, preserving narrative plausibility while avoiding direct data matching.

Definitions:

- watched_date, weekday: from WATCH_LOG_ITEM

- Seed: a short noun phrase capturing the implied core emotion/message/perspective/human-scale dilemma of the movie or series. Seed must be inspired by WATCH_LOG_ITEM; specific yet broadly applicable.

- Event: one sentence describing a plausible real-world situation on watched_date before viewing, grounded in USER_PROFILE, that would make the movie/series feel personally resonant or emotionally useful. Event must be explainable purely from USER_PROFILE + real-world context, (+ X, if provided), and derived from the seed. Event must be purely diegetic (in-world) and observational: do not mention or explain themes/messages/lessons/motifs (e.g., “heroism”, “redemption”, “identity”), and do not explicitly compare the event to the seed/title.

Seed → Event Mapping Strategies:

- situational_mirroring: Generate an event that mirrors the seed’s situation/material in the user’s real life, so the user is more likely to empathize or identify with the movie. The Event itself must still describe only the real-world situation before viewing.

- mood_modulation: Generate an event that establishes the user’s emotional state before viewing, making the movie appealing as a mood tool—either to match and release the emotion, counter and lift it, or maintain/amplify it. The Event itself must still describe only the real-world situation before viewing and any mood modulation.

- wish_compensation: Generate an event that highlights a concrete constraint or lack in the user’s life before viewing, making the movie appealing as vicarious fulfillment that “fills the gap.”

- curiosity_expansion: Generate an event that sparks a specific, non-emotional curiosity before viewing, such as a topic, skill, place, era, or culture, making the movie appealing as a way to explore, learn, or broaden perspective.

Task:

1. Read USER_PROFILE and WATCH_LOG_ITEM.

2. Quality constraints:

- Grounding: use at least one USER_PROFILE field.

- Expand: add one small new detail not explicit in USER_PROFILE.

- Specificity: include at least one concrete anchor, such as time, place, action, object, or role.

- Diversity: vary angle/anchor across four strategies; do not reuse the same anchor more than twice.

3. Create exactly four pairs, one per strategy key, in the required JSON.

Output Format:

Produce a JSON object with the following structure:

```
{
  "pairs": {
    "situational_mirroring": {
      "seed": "<short noun phrase>",
      "event": "On <watched_date>, <one sentence>"
    },
    "mood_modulation": {
      "seed": "<short noun phrase>",
      "event": "On <watched_date>, <one sentence>"
    },
    "wish_compensation": {
      "seed": "<short noun phrase>",
      "event": "On <watched_date>, <one sentence>"
    },
    "curiosity_expansion": {
      "seed": "<short noun phrase>",
      "event": "On <watched_date>, <one sentence>"
    }
  }
}
```

Constraints:

- Output raw JSON only.

- Event must be third-person, using “the user”.

- One event = one day. Describe only watched_date before viewing.

- Do not reenact the title’s plot or use title-unique entities/keywords in Events.

Now generate the output using the inputs below.

Input Data:

1. USER_PROFILE:

NOTE: relationship_status meanings

- single: not in a romantic relationship

- couple: in a romantic relationship but not married

- married: married

{User_Profile}

2. WATCH_LOG_ITEM:

NOTE: Use title/genres/description/keywords to derive Seeds, but do not copy title-unique entities or keywords into Events.

{Watch_Log_Item}

Output (JSON only)

Table 10: Prompt used for generating plausible events from a user’s profile and item metadata.

Instruction description

You will (1) select the best one event from the already-generated candidate pairs, then (2) minimally refine it only if needed for data quality.

Step 1: Selection of the Best Event:

- Choose the best one of the four pairs.
- Priority order: User alignment/Grounding > Anti-overfitting > Naturalness > Plausibility.
- selected_event must be copied exactly from one of the pairs.*.event values, with no rephrasing.

Red Flags:

Avoid the following in selection; if unavoidable, refine the event.

1. Same-day confusion: The event reads like it happens on the same day as another known event.
2. Movie-locked language: The event uses overt theme/setting words, such as “heroism”, “mythology”, “Norway/Norse myths”, or “alien culture”, or uses words such as “mirrors”, “resonates”, or “reminds”.

Step 2: Minimal Refinement Rules:

- Only apply edits if the selected_event triggers any red flag.
- Edit as little as possible, while keeping the same core situation and anchors.
- Allowed edits:
 - (a) Date clarity: Make it unambiguously on watched_date before viewing, for example by adding a small time marker such as “that morning” or “after work”.
 - (b) De-theme: Remove or replace movie-locked words with neutral real-life phrasing.

Output Format:

Produce a JSON object with the following structure:

```
{
  "reasoning": "Alignment=<...>; Anchors=<...>; Anti-overfit=<...>.",
  "selected_pair_key": "<situational_mirroring|mood_modulation|wish_compensation|curiosity_expansion>",
  "selected_event": "<exact copy from pairs[selected_pair_key].event>",
  "refined_event": "<either identical to selected_event OR minimally edited version>"
}
```

Constraints:

- Output raw JSON only.
- selected_event must be copied exactly from the selected candidate pair.
- refined_event must be identical to selected_event unless a red flag requires minimal editing.
- Do not introduce new events, new anchors, or unnecessary details during refinement.

Now select using the inputs below.

Input Data:**1. USER_PROFILE:**

NOTE: relationship_status meanings

- single: not in a romantic relationship
- couple: in a romantic relationship but not married
- married: married

{User_Profile}

2. WATCH_LOG_ITEM:

NOTE: Use title/genres/description/keywords to derive Seeds, but do not copy title-unique entities or keywords into Events.

{Watch_Log_Item}

3. CANDIDATE_PAIRS:

{Candidates}

Output (JSON only)

Table 11: Prompt used for selecting and refining the best event from candidate event pairs.

Instruction description

You generate plausible ‘day-of’ sub-events (Y) as direct causal continuations of the last existing event (X), using universal seeds inspired by a movie or series they watched to preserve narrative plausibility while avoiding direct data matching. The Event must be a direct causal continuation of X: it should explicitly reference one concrete, observable detail from X, such as person, action, object, commitment, place, or time, using a causal connector such as because, after, following, or in response to, and it should be unlikely to occur if X had not happened.

Definitions:

- watched_date, weekday: from WATCH_LOG_ITEM
- Seed: a short noun phrase capturing the implied core emotion/message/perspective/human-scale dilemma of the movie or series. Seed must be inspired by WATCH_LOG_ITEM; specific yet broadly applicable.
- Event: one sentence describing a plausible real-world situation on watched_date before viewing, grounded in USER_PROFILE, that would make the movie/series feel personally resonant or emotionally useful. Event must be explainable purely from USER_PROFILE + real-world context, (+ X, if provided), and derived from the seed. Event must be purely diegetic (in-world) and observational: do not mention or explain themes/messages/lessons/motifs, such as “heroism”, “redemption”, or “identity”, and do not explicitly compare the event to the seed/title.

Seed → Event Mapping Strategies:

- situational_mirroring: Generate an event that mirrors the seed’s situation/material in the user’s real life, so the user is more likely to empathize or identify with the movie. The Event itself must still describe only the real-world situation before viewing.
- mood_modulation: Generate an event that establishes the user’s emotional state before viewing, making the movie appealing as a mood tool—either to match and release the emotion, counter and lift it, or maintain/amplify it. The Event itself must still describe only the real-world situation before viewing and any mood modulation.
- wish_compensation: Generate an event that highlights a concrete constraint or lack in the user’s life before viewing, making the movie appealing as vicarious fulfillment that “fills the gap.”
- curiosity_expansion: Generate an event that sparks a specific, non-emotional curiosity before viewing, such as a topic, skill, place, era, or culture, making the movie appealing as a way to explore, learn, or broaden perspective.

Task:

1. Read USER_PROFILE, WATCH_LOG_ITEM, and LAST_EXISTING_EVENT (X).
2. Quality constraints:
 - Grounding: use at least one USER_PROFILE field.
 - Causal continuation: explicitly reference one concrete, observable detail from X using a causal connector.
 - Dependency: make the event unlikely to occur if X had not happened.
 - Expand: add one small new detail not explicit in USER_PROFILE.
 - Specificity: include at least one concrete anchor, such as time, place, action, object, or role.
 - Diversity: vary angle/anchor across four strategies; do not reuse the same anchor more than twice.
3. Create exactly four pairs, one per strategy key, in the required JSON.

Output Format:

Produce a JSON object with the following structure:

```
{
  "pairs": {
    "situational_mirroring": {
      "seed": "<short noun phrase>",
      "event": "On <watched_date>, <one sentence>"
    },
    "mood_modulation": {
      "seed": "<short noun phrase>",
      "event": "On <watched_date>, <one sentence>"
    },
    "wish_compensation": {
      "seed": "<short noun phrase>",
      "event": "On <watched_date>, <one sentence>"
    },
    "curiosity_expansion": {
      "seed": "<short noun phrase>",
      "event": "On <watched_date>, <one sentence>"
    }
  }
}
```

Constraints:

- Output raw JSON only.
- Event must be third-person, using “the user”.
- One event = one day. Describe only watched_date before viewing.
- Event must be a direct causal continuation of LAST_EXISTING_EVENT (X).
- Do not reenact the title’s plot or use title-unique entities/keywords in Events.

Now generate the output using the inputs below.

Input Data:**1. USER_PROFILE:**

NOTE: relationship_status meanings

- single: not in a romantic relationship
- couple: in a romantic relationship but not married
- married: married

{User_Profile}

2. WATCH_LOG_ITEM:

NOTE: Use title/genres/description/keywords to derive Seeds, but do not copy title-unique entities or keywords into Events.

{Watch_Log_Item}

3. LAST_EXISTING_EVENT (X):

{Last_Event}

Output (JSON only)

Table 12: Prompt used for generating plausible sub-events as causal continuations of the last existing event.

Instruction description

You will (1) select the best one event from the already-generated candidate pairs, then (2) minimally refine it only if needed for data quality.

Step 1: Selection of the Best Event:

- Choose the best one of the four pairs.
- Priority order: Causal directness > User alignment/Grounding > Anti-overfitting > Naturalness > Plausibility.
- selected_event must be copied exactly from one of the pairs.*.event values, with no rephrasing.

Causality Requirement (SUB):

- The event must be a direct causal continuation of X: cite one concrete trigger from X, such as person, action, object, commitment, place, or time, with an explicit causal connector such as because, after, following, or in response to, and it should be unlikely if X did not happen.

Red Flags:

Avoid the following in selection; if unavoidable, refine the event.

1. Same-day confusion: The event reads like it happens on the same day as X, although it must be on watched_date only.
2. Movie-locked language: The event uses overt theme/setting words, such as “heroism”, “mythology”, “Norway/Norse myths”, or “alien culture”, or uses words such as “mirrors”, “resonates”, or “reminds”.
3. Fake causality: The event uses causal wording without a concrete trigger from X, failing the counterfactual requirement.

Step 2: Minimal Refinement Rules:

- Only apply edits if the selected_event triggers any red flag.
- Edit as little as possible, while keeping the same core situation and anchors.
- Allowed edits:
 - (a) Date clarity: Make it unambiguously on watched_date before viewing, for example by adding a small time marker such as “that morning” or “after work”.
 - (b) De-theme: Remove or replace movie-locked words with neutral real-life phrasing.
 - (c) Strengthen causality: If the causal link is weak, keep the original trigger from X, but minimally adjust the resulting action or situation in the event so it becomes a plausible consequence of that trigger.

Output Format:

Produce a JSON object with the following structure:

```
{
  "reasoning": "Causality=<...>; Alignment=<...>; Anchors=<...>; Anti-overfit=<...>",
  "selected_pair_key": "<situational_mirroring|mood_modulation|wish_compensation|curiosity_expansion>",
  "selected_event": "<exact copy from pairs[selected_pair_key].event>",
  "refined_event": "<either identical to selected_event OR minimally edited version>"
}
```

Constraints:

- Output raw JSON only.
- selected_event must be copied exactly from the selected candidate pair.
- refined_event must be identical to selected_event unless a red flag requires minimal editing.
- Do not introduce new events, new anchors, or unnecessary details during refinement.
- If causality must be strengthened, keep the original trigger from X and change only the minimum necessary part of the resulting action or situation.

Now select using the inputs below.

Input Data:**1. USER_PROFILE:**

NOTE: relationship_status meanings

- single: not in a romantic relationship
- couple: in a romantic relationship but not married
- married: married

{User_Profile}

2. WATCH_LOG_ITEM:

NOTE: Use title/genres/description/keywords to derive Seeds, but do not copy title-unique entities or keywords into Events.

{Watch_Log_Item}

3. LAST_EXISTING_EVENT (X):

{Last_Event}

4. CANDIDATE_PAIRS:

{Candidates}

Output (JSON only)

Table 13: Prompt used for selecting and refining the best causal sub-event from candidate event pairs.

Instruction description

You are an expert data annotator and creative writer specializing in Conversational Recommender Systems (CRS). Your task is to generate a natural, multi-turn dialogue between a User and a Recommender Agent based on the provided information.

Given Information:

Analyze the following categories to drive the narrative:

1. Dialogue Persona: A holistic representation of user expression formed within the conversational context. It integrates linguistic attributes, such as identity, attitude, and speaking style, with user-related information observed through dialogue, including states, preferences, and factual information.
2. Behavioral Persona: A structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—to personalize decision-making effectively.
3. Event: A specific event that happened to the user on this day.
4. Memory: A bridge between sessions that captures the user's evolving history. It consists of an Episodic Memory, a chronological record of life events, and the Recommendation Outcomes from the immediate preceding session.
5. User's Historical Completion Stats
6. Source Items: Items used to navigate tastes as either Anchors, which serve as references for similarity or exclusion, or Recommendation Candidates, which must be rejected if suggested.
7. Target Item: The optimal solution to be proposed at the conclusion as the final choice. It must be the only item accepted by the user.

Strategic Dialogue Design:

Before generating the dialogue, you must first design the flow by integrating all given information into a coherent plan. This design process ensures that the interaction remains personalized, logically consistent, and leads naturally to the recommendation.

Phase 1: Persona Initialization

Initialize the user's identity and logic by integrating explicit conversational data with implicit behavioral patterns.

1. Dialogue Persona Manifestation

Establish the user's identity, attitude, and communication style, incorporating explicit preferences and user information revealed during the dialogue.

- Adaptive Manifestation: If the current flow of the dialogue, driven by the Event and Source/Target items, deviates from the established Dialogue Persona, interpret it as a situational shift or a natural evolution of the user's explicit identity and needs.
- Communication Style Operationalization: Translate the user's communication styles into specific verbal behaviors:
 - Clarity: Use well-defined requirements for High clarity, or express vague and mood-oriented preferences for Low clarity.
 - Initiative: Proactively drive the dialogue and ask questions for High initiative, or passively respond only to the Agent's inquiries for Low initiative.
 - Feedback: Deliver blunt, unambiguous evaluations with explicit reasons for Direct feedback, or use polite, hedged, or nuanced expressions for Indirect feedback.

2. Behavioral Persona Assessment

Analyze the user's implicit information and multi-dimensional tendencies derived from observable behavior logs.

- Behavioral Alignment: If the current flow of the dialogue, driven by the Event and Source/Target items, contradicts established patterns, interpret the shift as a situational event or a natural expansion/evolution of the user's underlying implicit tastes, usage patterns, and decision drivers.
- Progressive Concretization: Ensure that the latent preferences and recurring tendencies from the Behavioral Persona are not merely used as background filters but are actively articulated and concretized through the User's verbal expressions as the dialogue unfolds.
- Narrative Embodiment: Naturally weave implicit behavioral patterns into the dialogue by translating them into the user's subjective, everyday language.

Phase 2: Continuity and Contextual Setup

This phase focuses on why the conversation is happening now and how it connects to the user's past.

1. Memory Integration Strategy

- Awareness Boundary: The Agent is strictly unaware of the new Event until disclosed by the User. Strictly distinguish the new Event from past events in the Memory.
- Natural Timing: Use relative markers rather than specific dates.
- Memory Integration Guideline: Naturally weave in memory if appropriate for the current dialogue flow to ensure conversational continuity.
 - Agent-Initiated: The Agent may actively inquire about a past Event in memory or request a review of the previously accepted Target Item. When requesting a review of a past item, use open-ended questions instead of assuming the user liked it.
 - User-Initiated: The User may proactively initiate the session by linking the new Event to a past context or offering spontaneous feedback on the item accepted in the last session before the Agent asks.

2. Event Expansion

Deeply infuse the Event into the dialogue through Event Expansion, exploring how the event has impacted the user's current emotional state or practical needs.

Phase 3: Strategic Bridging and Navigation

In this phase, you bridge the user's current situation to the item exploration process.

1. Contextual Bridge Strategy

Select one of the following strategies to connect the Event to the Target Item:

- Mood Modulation: Recommend an item to either enhance or shift the user's current mood caused by the event.
- Situation Mirroring: Select items that reflect or resonate with the user's current circumstances.
- Wish Compensation: Offer an item that provides something the user currently lacks or desires due to the event.
- Curiosity Expansion: Use the event as a catalyst to explore new genres or topics the user is now curious about.

2. Source Item Role Assignment

Assign specific roles to the Source Items to facilitate the navigation of tastes. No Source Item can be accepted as a final recommendation; if recommended, it must result in rejection, such as already seen, soft reject, or direct reject.

- Preferred Items (High watched_pct): Assign roles such as Anchor, Already Seen, or Soft Reject.
 - Disliked Items (Low watched_pct): Assign roles such as Anchor to exclude styles or Reject due to taste mismatch.
 - Discussion of Source Items can be initiated by either the Agent or the User.
-

Table 14: Dialogue generation prompt, Part 1: persona initialization, continuity setup, and strategic bridging.

Instruction description

Phase 4: Narrative Arc and Target Synthesis

The final step is to organize the sequence of the dialogue so it leads to a successful resolution.

1. Sequence Planning (Narrative Arc)

Plan the sequence of the dialogue to ensure both the Event and all Source Items are woven in organically. The conversation should move through:

- Opening: Establishing context via Memory and the Event.
- Exploration: Discussing Source Items and gathering feedback to narrow down the search.
- Transition: Using the chosen Bridge Strategy to pivot toward the final solution.

2. Target Item Synthesis and Explainability

- Design a natural transition where the Target Item emerges as the only successful solution. It must logically synthesize the immediate impact of the Event, the underlying Behavioral Persona and Dialogue Persona, and the feedback gathered from discussing Source Items.
- The final recommendation utterance must explicitly verbalize the reasoning by integrating these synthesized points into a clear and persuasive explanation.
- This synthesis leads to a final acceptance of the Target Item, concluding the Narrative Arc with a highly personalized recommendation.

Core Rules and Constraints:**1. Item Usage Logic**

- Source Items: All must be discussed as anchors or recommendations.
- Target Item: Introduced only at the end. The User may ask for metadata details to verify suitability before the final acceptance.
- The final recommendation must emerge as the logical synthesis of the Event's impact, user personas, and source item feedback, explicitly verbalizing these factors into a clear and persuasive explanation.
- No Duplicate Recommendations: Once an item has been discussed or referenced in the dialogue, it must not be re-introduced or suggested as a new recommendation in any subsequent turns. Each item's role must be established and concluded in its respective turn.
- The Agent must not suggest re-watching items.

2. Factual Consistency

- Closed-World Constraint: Do not mention any items outside the provided Source Items, Target Item, or Memory.
- No Personal Experience: The Agent must not claim to have watched or felt emotions about the items.
- Strict Event-Blindness Protocol: The Agent must remain strictly unaware of the new Event until it is disclosed by the User. Disclosure occurs either when the User proactively introduces it or when the Agent prompts the User to share their day or current mood.
- Memory Accuracy: If the Memory is empty, do not mention or imply any past conversation history. Never use phrases such as "I remember you mentioned" for non-existent memory.
- Source vs. Memory Distinction: Do not refer to Source Items as if they were mentioned in previous sessions.

3. Structure and Labeling

- The dialogue must consist of between 10 and 14 turns.
- The dialogue must strictly follow an alternating sequence between the Agent and the User, strictly prohibiting any consecutive utterances from the Agent.
- Every Agent turn must start with exactly one of these labels: [chitchat], [question], [answer], [recommendation].
 - [recommendation]: Any utterance that proposes a specific title for the user to watch as a candidate in this session.
 - [chitchat]: Use for rapport, empathy, or general social interaction.
 - [question]: Use when asking about general preferences or moods without mentioning a specific title. Asking for a review or opinion on an item recommended in a previous session must also be labeled as [question].
 - [answer]: Use only when providing factual metadata or direct answers to a User's specific question about an item.
- User turns must not have any labels.

Reflection and Refinement:

Before providing the final dialogue, perform a self-audit to ensure that the Strategic Dialogue Design and Core Rules are strictly met. Verify that all Agent labels strictly match their functional definitions.

Output Instructions:

Respond only in valid JSON format as a single JSON object.

- Dialogue Design Strategy: Briefly describe how you will map the User's persona, the Event's impact on their persona, and the integrated flow of items and the event.
- Dialogue: Generate the multi-turn conversation as a list of objects. Each object must follow this structure:
 - label: The strategic label for the turn, only for Agent turns.
 - speaker: The role of the speaker, either Agent or User.
 - text: The actual utterance text.

[Example Output]

```
{
  "Dialogue Design Strategy": "...",
  "Dialogue": [
    {
      "speaker": "User",
      "text": "Utterance text here..."
    },
    {
      "label": "[chitchat]",
      "speaker": "Agent",
      "text": "Utterance text here..."
    }
  ]
}
```

Input Data:

Omitted for brevity; input fields follow the categories described above.

Generate conversation with multiple turns based on the given information.

Table 15: Dialogue generation prompt, Part 2: narrative synthesis, constraints, and output format.

Instruction description
You are an expert data analyst constructing a Behavioral Persona for a recommendation system. Definition of Behavioral Persona: “A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.” The Behavioral Persona consists of three main components: 1. Preferences (Positive/Negative) - subdivided into Content Material and Production Context. 2. Consumption Habits 3. Decision Driver Your Current Task: Focus exclusively on the “Content Material” sub-component of Preferences. You must analyze “What” the user consumes or avoids by contrasting high-engagement and low-engagement logs. Input Data Context: 1. User’s Historical Completion Stats - The provided Historical Stats already include the New Behavior Logs. 2. Behavior Logs - Relative Preference: Interpret completion rates relatively based on User’s Historical Completion Stats; treat items with relatively high completion rates as evidence to strengthen/expand Positive Preferences, and relatively low rates for Negative Preferences. - Even if the provided history is limited, you must infer the behavioral patterns based on the available records. Do not omit any fields or state that information is insufficient; provide the most plausible analysis derived from the existing logs while ensuring all sections are fully populated. Step 1: Identify Content Material Elements Examine the behavior logs to extract three core elements for both positive and negative preferences: 1) Genres (the broad category), 2) Subject Matter (tangible objects, settings, character types), 3) Topics (core messages, ideologies, or worldviews). Focus strictly on “What” is being consumed, identifying the recurring ingredients across the logs. Step 2: Resolve Metadata Conflicts Extract Positive attributes only from high-completion logs and Negative attributes only from low-completion logs; do not reference the other set in wording (no “relative to” framing). Use contrast only to validate consistency. If an attribute appears in both sets, (a) split into non-overlapping context facets when supported by logs; otherwise (b) treat as ambivalent and drop it from both. Ensure Positive and Negative are mutually exclusive (no duplicates). Step 3: Generate Narrative Output Synthesize the findings into two distinct narrative paragraphs (one for positive, one for negative). Do not use comparative phrases like “instead of,” “rather than,” or “prefers A over B.” Avoid referencing specific titles. To keep the response concise, produce each paragraph in no more than three sentences, starting directly with the subject. [Example Output] <pre>{ "positive_content_material": "The user ...", "negative_content_material": "The user ..." }</pre> Now, based on the information provided below, extract the Preference - Content Material. Input Data: 1. User’s Historical Completion Stats: {user_completion_stats} 2. Behavior Logs: {log_data} Output (JSON only)

Table 16: Prompt used for extracting the session-level content material signal used to update the behavioral persona.

Instruction description

You are an expert data analyst constructing a Behavioral Persona for a recommendation system.

Definition of Behavioral Persona:

“A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.”

The Behavioral Persona consists of three main components:

1. Preferences (Positive/Negative) - subdivided into Content Material and Production Context.
2. Consumption Habits
3. Decision Driver

Your Current Task: Focus exclusively on the “Production Context” sub-component of Preferences. You must analyze the “External Metadata” (Who, When, Where) by contrasting high-engagement and low-engagement logs.

Input Data Context:

1. User’s Historical Completion Stats
 - The provided Historical Stats already include the New Behavior Logs.
2. Behavior Logs
 - Relative Preference: Interpret completion rates relatively based on User’s Historical Completion Stats; treat items with relatively high completion rates as evidence to strengthen/expand Positive Preferences, and relatively low rates for Negative Preferences.
 - Even if the provided history is limited, you must infer the behavioral patterns based on the available records. Do not omit any fields or state that information is insufficient; provide the most plausible analysis derived from the existing logs while ensuring all sections are fully populated.

Step 1: Identify Production Metadata Elements

Examine the behavior logs to identify recurring signals across four elements: 1) Actors (recurring names), 2) Directors (filmmakers whose works are consistently completed or abandoned), 3) Countries (preferred or avoided production origins), 4) Release Year (favored or avoided eras/time periods). Focus strictly on these metadata fields and COMPLETELY IGNORE the content material (genre, subject matter, topic).

Step 2: Resolve Metadata Conflicts

Extract Positive attributes only from relatively high-completion logs and Negative attributes only from relatively low-completion logs; do not reference the other set in wording (no “relative to” framing). Use contrast only to validate consistency. If an attribute appears in both sets, (a) split into non-overlapping context facets when supported by logs (e.g., era, country, collaborator); otherwise (b) treat as ambivalent and drop it from both. Ensure Positive and Negative are mutually exclusive (no duplicates).

Step 3: Generate Narrative Output

Synthesize the findings into two distinct narrative paragraphs (one for positive, one for negative). Do not use comparative phrases like “instead of,” “rather than,” or “prefers A over B.” Avoid referencing specific titles. To keep the response concise, produce each paragraph in no more than three sentences, starting directly with the subject.

[Example Output]

```
{
  "positive_production_context": "The user ...",
  "negative_production_context": "The user ..."
}
```

Now, based on the information provided below, extract the Positive/Negative Preference - Production Context.

Input Data:

1. User’s Historical Completion Stats: {user_completion_stats}
2. Behavior Logs: {log_data}

Output (JSON only):

Table 17: Prompt used for extracting the session-level production context signal used to update the behavioral persona.

Instruction description

You are an expert data analyst constructing a Behavioral Persona for a recommendation system.

Definition of Behavioral Persona:

“A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.”

The Behavioral Persona consists of three main components:

1. Preferences (Positive/Negative)
2. Consumption Habits (Temporal patterns, Immersion style)
3. Decision Drivers

Your Current Task: Focus exclusively on extracting the “Consumption Habits - Temporal Patterns” component. You must analyze the user’s “Life Rhythm” based on the viewing history.

Input Data Context:

1. User’s Historical Completion Stats
 - The provided Historical Stats already include the New Behavior Logs.
2. Behavior Logs
 - The input is a dictionary where Keys are Dates. Each Value is an object containing:
 - * day: The string representing the day of the week.
 - * watched_pct: A list of completion percentages.
 - Relative Activeness: Evaluate the watched_pct based on the user’s historical stats. “Active” viewing is defined by both Quantity and Quality: A specific day is considered “Active” only if the list is long (many items watched) AND the values are high (high completion rates).
 - Even if the provided history is limited, you must infer the behavioral patterns based on the available records.

Step 1: Analyze Frequency & Schedule

Using the Dates, Days, and the definition of Active viewing above, identify lifestyle patterns:

1. Peak Activity Days: Identify which specific days consistently show the most “Active” behavior (high volume + high completion), distinguishing them from days with only light or passive usage.
2. Routine Type: Determine the overall rhythm. Is it Clustered (heavy activity only on specific days like weekends), Steady (consistent, moderate viewing spread evenly across most days), or Sporadic (random/unpredictable gaps)?

Step 2: Generate Narrative Output

Synthesize these findings into a narrative paragraph. Do not list the items. Describe the user’s routine naturally across the observed dates. Avoid referencing specific titles. To keep the response concise, you must produce the answer in no more than three sentences.

[Example Output]

```
{
  "temporal_patterns": "The user ..."
```

Now, based on the information provided below, extract the Consumption Habits - Temporal Patterns.

Input Data:

1. User’s Historical Stats: {user_stats}
2. Behavior Logs: {log_data}

Output (JSON only):

Table 18: Prompt used for extracting the session-level temporal patterns signal used to update the behavioral persona.

Instruction description

You are an expert data analyst constructing a Behavioral Persona for a recommendation system.

Definition of Behavioral Persona:

“A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.”

The Behavioral Persona consists of three main components:

1. Preferences (Positive/Negative)
2. Consumption Habits (Temporal patterns, Immersion style)
3. Decision Drivers

Your Current Task: Focus exclusively on extracting the “Consumption Habits - Immersion Style” component. You must analyze the user’s day-level viewing depth based on the viewing history.

Input Data Context:

1. User’s Historical Completion Stats
 - The provided Historical Stats already include the New Behavior Logs.
2. Behavior Logs
 - The input is a dictionary where Keys are Dates. Each Value is an object containing:
 - * day: The string representing the day of the week.
 - * watched_pct: A list of completion percentages.
 - Relative Immersion: Evaluate the watched_pct based on the user’s historical stats.
 - Even if the provided history is limited, you must infer the behavioral patterns based on the available records.

Step 1: Analyze Immersion Metrics

Examine the watched_pct lists to identify viewing behaviors:

1. Commitment Level: Check if completion is typically high relative to the user’s historical completion stats (e.g., at/above their historical average or upper-range), or if there are frequent low-relative drop-offs.
2. Binging: Analyze the volume of completed items. “True binging” means a high number of titles in a day with consistently high completion rates. A high number of titles with low completion rates indicates zapping/indecisive browsing.

Step 2: Generate Narrative Output

Synthesize these findings into a narrative paragraph. Do not list the items. Describe the user’s viewing style naturally. Avoid referencing specific titles. To keep the response concise, you must produce the answer in no more than three sentences.

[Example Output]

```
{
  "immersion_style": "The user ..."
}
```

Now, based on the information provided below, extract the Consumption Habits - Immersion Style.

Input Data:

1. User’s Historical Stats: {user_stats}
2. Behavior Logs: {log_data}

Output (JSON only):

Table 19: Prompt used for extracting the session-level immersion style signal used to update the behavioral persona.

Instruction description

You are an expert data analyst constructing a Behavioral Persona for a recommendation system.

Definition of Behavioral Persona:

“A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.”

The Behavioral Persona consists of three main components:

1. Preferences (Content Material, Production Context)
2. Consumption Habits
3. Decision Drivers (Selection Priority, Behavioral Motivation)

Your Current Task: Focus exclusively on extracting the “Decision Drivers - Selection Priority” component. You must identify the Hierarchy of Choice by uncovering the single most influential factor, the “Anchor”, that governs the user’s positive engagement.

Input Data Context:

1. User’s Historical Completion Stats
 - The provided Historical Stats already include the New Behavior Logs.
2. Behavior Logs
 - Relative Preference: Interpret completion rates relatively based on User’s Historical Completion Stats; treat items with relatively high completion rates as evidence to strengthen/expand Positive Preferences, and relatively low rates for Negative Preferences.
 - Consistency Rule: Summarized Preferences are also provided as reference. Your analysis must ensure a seamless logical connection between the behavior logs and these preferences, ensuring the final output does not conflict with either data source.
 - Even if the provided history is limited, you must infer the behavioral patterns based on the available records.

Step 1: Analyze the Raw Behavior Logs

Examine the high-engagement items, defined as relatively high completion compared with the user’s historical completion stats, to identify the primary anchor that remains consistent across them across sessions.

Step 2: Cross-check with the Reference Summary

Verify the identified anchor against the confirmed Positive and Negative Preferences. Ensure the final output does not logically conflict with the provided Positive/Negative Preferences.

Step 3: Generate Narrative Output

Synthesize the findings into a narrative paragraph. Do not list the items. Describe the user’s selection priority naturally. Avoid referencing specific titles. To keep the response concise, you must produce the answer in no more than three sentences.

[Example Output]

```
{
  "selection_priority": "The user ..."
}
```

Now, based on the provided Input Data, extract the Decision Drivers - Selection Priority.

Input Data:

1. Preference: {preference}
2. User’s Historical Completion Stats: {user_completion_stats}
3. Behavior Logs: {log_data}

Output (JSON only):

Table 20: Prompt used for extracting the session-level selection priority signal used to update the behavioral persona.

Instruction description
You are an expert data analyst constructing a Behavioral Persona for a recommendation system. Definition of Behavioral Persona: “A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.” The Behavioral Persona consists of three main components: 1. Preferences 2. Consumption Habits 3. Decision Drivers (Selection Priority, Behavioral Motivation) Your Current Task: Focus exclusively on extracting the “Decision Drivers - Behavioral Motivation” component based on the viewing history. You must infer the “Functional Purpose” (The implicit goal or reward) behind the user’s viewing sessions. Input Data Context: 1. User’s Historical Completion Stats - The provided Historical Stats already include the New Behavior Logs. 2. Behavior Logs - Relative Preference: Interpret completion rates relatively based on User’s Historical Completion Stats; treat items with relatively high completion rates as evidence to strengthen/expand Positive Preferences, and relatively low rates for Negative Preferences. - Consistency Rule: Summarized Preferences are also provided as reference. Your analysis must ensure a seamless logical connection between the behavior logs and these preferences, ensuring the final output does not conflict with either data source. - Even if the provided history is limited, you must infer the behavioral patterns based on the available records. Step 1: Map Content Patterns to Functional Motivations Analyze the nature of the high-engagement content, defined as relatively high completion compared with the user’s historical completion stats, to deduce the user’s goal. DO NOT discuss “Criteria” (Selection Priority). FOCUS on “Utility” (Motivation). Use the following Inference Logic: 1. Stimulation & Adrenaline (Visceral Reward): - Signal: High engagement with Action/Thriller/Horror/Racing genres marked by chase/fight/violence/high-stakes cues and higher age_rating. - Inference: The user seeks immediate excitement, tension, and sensory stimulation. 2. Escapism & Immersion (Imaginative Reward): - Signal: High engagement with Fantasy/Sci-Fi/Adventure genres marked by world-building cues (alternate worlds, magic/aliens, quests, lore) and a relatively high watched_pct. - Inference: The user seeks to detach from reality and immerse themselves in alternative universes. 3. Mood Regulation & Comfort (Emotional Reward): - Signal: High engagement with Family/Animation/Comedy/Musical genres marked by warmth/relationship/growing-up cues and lower age_rating. - Inference: The user seeks relaxation, emotional warmth, or stress relief (healing/comfort viewing). 4. Intellectual Curiosity (Cognitive Reward): - Signal: High engagement with Mystery/Crime/Detective/Documentary/History genres marked by investigation/clue/truth-seeking cues and a relatively high watched_pct. - Inference: The user seeks mental stimulation, puzzle-solving, or knowledge acquisition. 5. Killing Time & Passive Enjoyment (Low-Effort Reward): - Signal: High engagement with light Comedy/Popcorn content marked by simple-premise descriptions, low cognitive/world-rule cues, and a relatively “good enough” watched_pct. - Inference: The user seeks to pass time with minimal mental effort and light entertainment. 6. Social Connection & Talk Value (Social Reward): - Signal: High engagement with blockbuster/franchise/trending-popular content marked by recognizable IP cues and relatively recent release_year. - Inference: The user seeks to stay culturally relevant (FOMO), share common ground with peers, or acquire topics for social conversation (“Watercooler moments”). Step 2: Cross-check with the Reference Summary Verify the identified anchor against the confirmed Positive and Negative Preferences. Ensure the final output does not logically conflict with the provided Positive/Negative Preferences. Step 3: Generate Narrative Output Synthesize the findings into a narrative paragraph. Do not list the items. Describe the user’s motivation naturally. Avoid referencing specific titles. Avoid emotional guessing. Stick to the functional utility of the media consumption. To keep the response concise, you must produce the answer in no more than three sentences. [Example Output] <pre>{ "behavioral_motivation": "The user ..." }</pre> Now, based on the provided Input Data, extract the Decision Drivers - Behavioral Motivation. Input Data: 1. Preference: {preference} 2. User’s Historical Completion Stats: {user_completion_stats} 3. Behavior Logs: {log_data} Output (JSON only):

Table 21: Prompt used for extracting the session-level motivation signal used to update the behavioral persona.

Instruction description

You are an expert data analyst synthesizing a Long-term Behavioral Persona for a recommendation system.

Definition of Behavioral Persona:

“A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.”

The Behavioral Persona consists of three main components:

1. Preferences (Positive/Negative) - subdivided into Content Material and Production Context.
2. Consumption Habits
3. Decision Driver

Your Task:

Focus exclusively on the “Content Material” sub-component of Preferences. You must update the “Long-term Persona” by integrating the “Session-level Persona” newly extracted from the latest behavior.

Input Data Context:

1. Long-term Persona (Current): The user’s established preferences accumulated from all previous sessions.
2. Session-level Persona (New): The user’s preferences extracted specifically from the current session’s logs.

Step 1: Analyze Long-term Persona

Review the established Genres, Subject Matter, and Topics in the Long-term Persona to understand the user’s historical baseline.

Step 2: Evaluate Session-level Persona

Identify the newly revealed Genre, Subject Matter, and Topic insights from the Session-level Persona.

Step 3: Synthesize and Evolve

- Reinforce (Consistency): If the Session-level insights align with the Long-term Persona, reinforce those specific traits with more detail or higher confidence.
- Integrate (Expansion): If the Session-level insights introduce new elements not present in the Long-term Persona, merge them to reflect a broadening spectrum of interest. Treat these as cumulative additions.
- Pivot (Contradiction): If the Session-level insights directly contradict the Long-term Persona, such as a previously liked topic now explicitly marked as negative in the current session, prioritize the Session-level insights based on recency to update the core profile.

Step 4: Resolve Metadata Conflicts

- Perform a mandatory check to ensure the updated Positive and Negative sets are mutually exclusive.
- If an attribute appears in both sets:
 - (a) Split into non-overlapping context facets if logically supported.
 - (b) Otherwise, prioritize the Session-level insight to assign the attribute, or drop it from both if it remains ambivalent.

Step 5: Generate Final Narrative Output

Synthesize the insights into two distinct narrative paragraphs.

- Avoid listing; use overarching concepts to describe the user’s evolution.
- Do not use comparative phrases like “instead of,” “rather than,” or “prefers A over B.”
- Avoid referencing specific movie/series titles.
- Keep each paragraph to a maximum of three sentences, starting directly with the subject.

[Example Output]

```
{
  "positive_content_material": "",
  "negative_content_material": ""
}
```

Now, update the Preference - Content Material using the inputs below.

Input Data:

1. Long-term Positive Content Material: {current_positive_content}
2. Long-term Negative Content Material: {current_negative_content}
3. Session-level Positive Content Material (New): {new_positive_content}
4. Session-level Negative Content Material (New): {new_negative_content}

Output (JSON only)

Table 22: Prompt used for updating the content material component of the behavioral persona.

Instruction description

You are an expert data analyst synthesizing a Long-term Behavioral Persona for a recommendation system.

Definition of Behavioral Persona:

“A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.”

The Behavioral Persona consists of three main components:

1. Preferences (Positive/Negative) - subdivided into Content Material and Production Context.
2. Consumption Habits
3. Decision Driver

Your Task:

Focus exclusively on the “Production Context” sub-component of Preferences. You must synthesize the “Long-term Persona” with the newly extracted “Session-level Persona” to create an evolved representation of the user’s external metadata preferences, including Who, When, and Where.

Input Data Context:

1. Long-term Persona (Current Production Context): The user’s established preferences regarding actors, directors, countries, and eras accumulated from previous sessions.
2. Session-level Persona (New Production Context): The user’s external metadata preferences extracted specifically from the current session’s behavior.

Step 1: Understand Long-term Persona

Analyze the Long-term Persona to understand the established baseline for Actors, Directors, Production Countries, and Release Eras that the user has previously favored or avoided.

Step 2: Evaluate Session-level Persona

Identify the newly revealed Production Context insights, including Actors, Directors, Countries, and Eras, from the Session-level Persona.

Step 3: Evolve and Synthesize

- Reinforce and Specify (Same Direction): If the Session-level insights align with the Long-term Persona, such as the same director, country, or era, reinforce these preferences with more detail or higher confidence.
- Expand and Integrate (New Direction): If the Session-level insights introduce different metadata not present in the Long-term Persona, merge them to reflect a broadening spectrum of interest. Treat these as cumulative additions, not contradictions.
- Prioritize Recency (Contradictory Direction): Execute a preference pivot only if there is a direct conflict, such as when an element previously marked as positive is now explicitly categorized as negative in the Session-level Persona. In this case, prioritize the latest evidence based on recency to reflect the user’s current intent.

Step 4: Resolve Metadata Conflicts and Ensure Mutual Exclusivity

- Perform a mandatory conflict check to ensure the updated Positive and Negative sets are mutually exclusive.
- If an attribute appears in both sets:
 - (a) Split into non-overlapping context facets, such as “[Country] during [Era A]” versus “[Country] during [Era B],” if logically supported by the personas.
 - (b) Otherwise, prioritize the Session-level insight to assign the attribute, or drop it from both if it remains ambivalent.
- Ensure the final Positive and Negative narratives contain no overlapping attributes.

Step 5: Generate Updated Narrative Output

Synthesize the insights into two distinct narrative paragraphs, one for positive and one for negative.

- Avoid simple listing; consolidate overlapping elements into higher-level overarching concepts or eras, such as mid-2010s or early 2020s.
- Avoid referencing specific movie/series titles.
- Do not use comparative phrases like “instead of,” “rather than,” or “prefers A over B.”
- Keep each paragraph to a maximum of three sentences, starting directly with the subject.

[Example Output]

```
{
  "positive_production_context": "",
  "negative_production_context": ""
}
```

Now, update the Preference - Production Context using the inputs below.

Input Data:

1. Long-term Positive Production Context: {current_positive_production}
2. Long-term Negative Production Context: {current_negative_production}
3. Session-level Positive Production Context (New): {new_positive_production}
4. Session-level Negative Production Context (New): {new_negative_production}

Output (JSON only)

Table 23: Prompt used for updating the production context component of the behavioral persona.

Instruction description

You are an expert data analyst synthesizing a Long-term Behavioral Persona for a recommendation system.

Definition of Behavioral Persona:

“A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.”

The Behavioral Persona consists of three main components:

1. Preferences (Positive/Negative)
2. Consumption Habits (Temporal patterns, Immersion style)
3. Decision Drivers

Your Task:

Focus exclusively on the “Temporal patterns” sub-component of Consumption Habits. You must synthesize the “Long-term Persona” with the newly extracted “Session-level Persona” to create an evolved representation of the user’s life rhythm and viewing schedule.

Input Data Context:

1. Long-term Persona (Current Temporal Patterns): The user’s established life rhythm, peak activity days, and routine type, such as Clustered, Steady, or Sporadic, accumulated from previous sessions.
2. Session-level Persona (New Temporal Patterns): The user’s engagement rhythm and activeness insights extracted specifically from the current session.

Step 1: Understand Long-term Persona

Analyze the Long-term Persona to understand the established baseline for Peak Activity Days and the Routine Type that has characterized the user’s behavior so far.

Step 2: Evaluate Session-level Persona

Identify the newly revealed Temporal insights from the Session-level Persona, focusing on whether the current session’s activeness reinforces or shifts the established rhythm.

Step 3: Evolve and Synthesize

- Reinforce (Consistency): If the Session-level insights align with the Long-term Persona, such as the same peak days or routine type, reinforce the established pattern with higher confidence or more detail.
- Integrate (Expansion): If the Session-level insights indicate minor variations or a widening of the viewing window, merge them to reflect a more flexible or evolving routine.
- Pivot (Contradiction): If the Session-level insights strongly contradict the Long-term Persona, such as a shift from weekend clustering to heavy weekday usage, prioritize the latest evidence based on recency to reflect the user’s current life rhythm.

Step 4: Generate Updated Narrative Output

Synthesize the insights into a single narrative paragraph.

- Avoid simple listing; consolidate overlapping elements into higher-level overarching concepts, such as “primarily active during mid-week evenings.”
- Describe the user’s routine naturally, reflecting any evolution from the past to the present.
- Avoid referencing specific movie/series titles.
- Keep the response to a maximum of three sentences, starting directly with the subject.

[Example Output]

```
{
  "temporal_patterns": ""
}
```

Now, update the Consumption Habits - Temporal Patterns using the inputs below.

Input Data:

1. Long-term Temporal Patterns: {current_temporal_persona}
2. Session-level Temporal Patterns (New): {new_temporal_persona}

Output (JSON only)

Table 24: Prompt used for updating the temporal patterns component of the behavioral persona.

Instruction description

You are an expert data analyst synthesizing a Long-term Behavioral Persona for a recommendation system.

Definition of Behavioral Persona:

“A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.”

The Behavioral Persona consists of three main components:

1. Preferences (Positive/Negative)
2. Consumption Habits (Temporal patterns, Immersion style)
3. Decision Drivers

Your Task:

Focus exclusively on the “Immersion style” sub-component of Consumption Habits. You must synthesize the “Long-term Persona” with the newly extracted “Session-level Persona” to create an evolved representation of the user’s viewing depth and commitment level.

Input Data Context:

1. Long-term Persona (Current Immersion style): The user’s established baseline regarding commitment level, such as high completion or frequent drop-offs, and bingeing tendencies, such as True bingeing or Zapping/Browsing, accumulated from previous sessions.
2. Session-level Persona (New Immersion style): The user’s current immersion signals and viewing depth insights extracted specifically from the current session.

Step 1: Understand Long-term Persona

Analyze the Long-term Persona to understand the established Commitment Level and Bingeing Tendency that have characterized the user’s immersive behavior so far.

Step 2: Evaluate Session-level Persona

Identify newly revealed Immersion insights from the Session-level Persona, focusing on whether the current session’s behavior, such as focused deep-viewing or indecisive browsing, reinforces or shifts the established pattern.

Step 3: Evolve and Synthesize

- Reinforce (Consistency): If the Session-level insights align with the Long-term Persona, reinforce the established pattern with higher confidence or specify the conditions under which the user is most immersive.
- Integrate (Expansion): If the Session-level insights indicate situational shifts or minor variations in viewing depth, merge them to reflect a more multi-layered or flexible immersion style.
- Pivot (Contradiction): If the Session-level insights strongly contradict the Long-term Persona, such as a shift from habitual zapping to consistent full-completion, prioritize the latest evidence based on recency to reflect the user’s current immersive behavior.

Step 4: Generate Updated Narrative Output

Synthesize the insights into a single narrative paragraph.

- Avoid simple listing; consolidate overlapping elements into higher-level overarching concepts, such as “exhibits a high-depth commitment style.”
- Describe the user’s viewing depth and commitment style naturally, reflecting any evolution over time.
- Avoid referencing specific movie/series titles.
- Keep the response to a maximum of three sentences, starting directly with the subject.

[Example Output]

```
{  
  "immersion_style": ""  
}
```

Now, update the Consumption Habits - Immersion style using the inputs below.

Input Data:

1. Long-term Immersion style: {current_immersion_persona}
2. Session-level Immersion style (New): {new_immersion_persona}

Output (JSON only)

Table 25: Prompt used for updating the immersion style component of the behavioral persona.

Instruction description

You are an expert data analyst synthesizing a Long-term Behavioral Persona for a recommendation system.

Definition of Behavioral Persona:

“A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.”

The Behavioral Persona consists of three main components:

1. Preferences (Content Material, Production Context)
2. Consumption Habits (Temporal patterns, Immersion style)
3. Decision Drivers (Selection Priority, Behavioral Motivation)

Your Task:

Focus exclusively on the “Selection Priority” sub-component of Decision Drivers. You must synthesize the “Long-term Persona” with the newly extracted “Session-level Persona” to create an evolved representation of the user’s “Hierarchy of Choice” and the primary “Anchor” that governs their engagement.

Input Data Context:

1. Long-term Persona (Current Selection Priority): The user’s established primary decision factor, or Anchor, accumulated from previous sessions.
2. Session-level Persona (New Selection Priority): The primary decision factor identified specifically from the current session’s behavior.
3. Updated Preferences (Reference): The user’s latest Positive/Negative preferences for Content Material and Production Context. Your synthesis must align logically with these updated preferences.

Step 1: Understand Long-term Persona

Analyze the Long-term Persona to identify the established Anchor, such as a specific genre, director style, or era, that has historically driven the user’s high-engagement choices.

Step 2: Evaluate Session-level Persona

Identify the Anchor signal from the Session-level Persona. Determine whether the current session’s choices were driven by the same historical factor or whether a new primary driver has emerged.

Step 3: Evolve and Synthesize

- Reinforce (Consistency): If the Session-level anchor aligns with the Long-term Persona, reinforce the Anchor with higher confidence or specify sub-conditions.
- Expand (New Direction): If the Session-level persona introduces a new secondary driver that does not contradict the baseline, merge them to reflect a more multi-layered hierarchy of choice.
- Pivot (Contradiction): If the Session-level anchor strongly contradicts the Long-term Persona, prioritize the latest evidence based on recency to reflect a fundamental shift in the user’s hierarchy of choice.

Step 4: Cross-check with Preferences

Verify that the synthesized Selection Priority is logically consistent with the Updated Preferences. The primary Anchor must be reflected in the user’s strongest positive preferences.

Step 5: Generate Updated Narrative Output

Synthesize the insights into a single narrative paragraph.

- Avoid simple listing; consolidate insights into overarching concepts about the user’s decision-making logic.
- Avoid referencing specific movie/series titles.
- Do not use comparative phrases like “instead of,” “rather than,” or “prefers A over B.”
- Keep the response to a maximum of three sentences, starting directly with the subject.

[Example Output]

```
{
  "selection_priority": ""
}
```

Now, update the Decision Drivers - Selection Priority using the inputs below.

Input Data:

1. Long-term Selection Priority: {current_selection_priority}
2. Session-level Selection Priority (New): {new_selection_priority}
3. Updated Preferences (Content & Production): {updated_preferences}

Output (JSON only)

Table 26: Prompt used for updating the selection priority component of the behavioral persona.

Instruction description

You are an expert data analyst synthesizing a Long-term Behavioral Persona for a recommendation system.

Definition of Behavioral Persona:

“A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.”

The Behavioral Persona consists of three main components:

1. Preferences (Content Material, Production Context)
2. Consumption Habits (Temporal patterns, Immersion style)
3. Decision Drivers (Selection Priority, Behavioral Motivation)

Your Task:

Focus exclusively on the “Behavioral Motivation” sub-component of Decision Drivers. You must synthesize the “Long-term Persona” with the newly extracted “Session-level Persona” to create an evolved representation of the user’s “Functional Purpose,” or the implicit goal or reward.

Input Data Context:

1. Long-term Persona (Current Behavioral Motivation): The user’s established psychological goals and rewards accumulated from previous sessions.
2. Session-level Persona (New Behavioral Motivation): The functional reward identified specifically from the current session’s behavior.
3. Updated Preferences (Reference): The user’s latest Positive/Negative preferences. Your synthesis must align logically with these updated preferences.

Step 1: Understand Long-term Persona

Analyze the Long-term Persona to identify the user’s historical functional rewards and purposes.

Step 2: Evaluate Session-level Persona using Mandatory Taxonomy

Identify the reward signal from the Session-level Persona. You must interpret and synthesize the motivation based on the following six categories:

1. Stimulation & Adrenaline (Visceral Reward): High engagement with Action, Thriller, Horror, or Racing genres marked by chase, fight, violence, high-stakes cues, and higher age ratings indicates that the user seeks immediate excitement, tension, and sensory stimulation.
2. Escapism & Immersion (Imaginative Reward): High engagement with Fantasy, Sci-Fi, or Adventure genres marked by world-building cues, such as alternate worlds, magic, aliens, quests, or lore, indicates that the user seeks to detach from reality and immerse themselves in alternative universes.
3. Mood Regulation & Comfort (Emotional Reward): High engagement with Family, Animation, Comedy, or Musical genres marked by warmth, relationships, growing-up cues, and lower age ratings indicates that the user seeks relaxation, emotional warmth, or stress relief.
4. Intellectual Curiosity (Cognitive Reward): High engagement with Mystery, Crime, Detective, Documentary, or History genres marked by investigation, clue, and truth-seeking cues indicates that the user seeks mental stimulation, puzzle-solving, or knowledge acquisition.
5. Killing Time & Passive Enjoyment (Low-Effort Reward): High engagement with light Comedy or Popcorn content marked by simple-premise descriptions, low cognitive or world-rule cues, and good-enough watched percentages indicates that the user seeks to pass time with minimal mental effort and light entertainment.
6. Social Connection & Talk Value (Social Reward): High engagement with blockbuster, franchise, trending, or popular content marked by recognizable IP cues and relatively recent release years indicates that the user seeks cultural relevance, common ground with peers, or topics for social conversation.

Step 3: Evolve and Synthesize

- Reinforce (Consistency): If the Session-level reward aligns with the Long-term Persona, reinforce this motivation with higher confidence.
- Expand (New Direction): If the Session-level persona introduces a different category from the taxonomy, interpret this as a multi-layered motivation or a situational expansion.
- Pivot (Contradiction): If the Session-level reward strongly contradicts the Long-term pattern, prioritize the latest evidence based on recency to reflect the user’s current goal.

Step 4: Cross-check with Preferences

Verify that the synthesized Behavioral Motivation is logically consistent with the Updated Preferences.

Step 5: Generate Updated Narrative Output

Synthesize the insights into a single narrative paragraph.

- Avoid simple listing; use overarching concepts from the six categories above.
- Avoid referencing specific movie/series titles.
- Keep the response to a maximum of three sentences, starting directly with the subject.

[Example Output]

```
{
  "behavioral_motivation": ""
}
```

Now, update the Decision Drivers - Behavioral Motivation using the inputs below.

Input Data:

1. Long-term Behavioral Motivation: {current_motivation_persona}
2. Session-level Behavioral Motivation (New): {new_motivation_persona}
3. Updated Preferences (Content & Production): {updated_preferences}

Output (JSON only)

Table 27: Prompt used for updating the motivation component of the behavioral persona.

Instruction description

You are an expert linguistic analyst specializing in user modeling for Conversational Recommender Systems (CRS). Your task is to extract a “Dialogue Persona” as a list of descriptive statements and “Memory” from a given conversation.

Definitions:

1. Dialogue Persona (List of Strings):

A dialogue persona is a holistic representation of user expression formed within the conversational context. It integrates not only linguistically expressed attributes, such as identity, attitude, and speaking style, but also user-related information that can be observed through dialogue, including states, preferences, and factual information.

2. Memory:

- Session Event: Specific real-life events or contexts the user mentioned happening to them on this day.
- Recommendation Outcomes (List of Dictionaries): A record of all items recommended by the system during the session.

Each entry must include:

- * item: The title of the movie or series recommended.
- * reaction: The user’s response type. Choose from Accept, Reject, Soft Reject, Already Seen, or Neutral.
- * reason: The specific linguistic or preferential reason given by the user for their reaction.

Instructions:

Step 1: Extract Dialogue Persona. Synthesize these into a list of clear, independent descriptive sentences.

Step 2: Extract Session Event. Identify the real-world context or event the user is experiencing today.

Step 3: Analyze all Recommendation Outcomes. Identify every item the system suggested. For each item, determine the reaction type and provide the specific reasoning, paying close attention to whether a rejection is based on taste (Reject) or situational timing (Soft Reject).

Step 4: Reflection and Refinement (Filter out specific items). Review Dialogue Persona and filter out reactions about specific items, such as movies or series.

Step 5: Reflection and Refinement (Filter out daily events). Review Dialogue Persona and filter out one-time occurrences, such as what happened to the user on this day, unless they demonstrate a persistent pattern or an enduring trait.

Constraints:

- Use the language of the provided dialogue for the output.
- Each string in the dialogue_persona list must be a single, concise sentence.
- Never use time-specific adverbs like “today” or “now” in the Dialogue Persona.
- No Specific Titles in Dialogue Persona: Never include movie or series titles in the Dialogue Persona. Move the specific item data to Memory.
- No Episodic Events in Dialogue Persona: Do not include what happened today in the persona. Move all situational contexts to Memory.

Output Format:

Produce a JSON object with the following structure:

```
{
  "dialogue_persona": [
    "",
    ""
  ],
  "memory": {
    "session_event": "[{day}] What happened to the user?",
    "recommendation_outcomes": [
      {
        "item": "Movie Title A",
        "reaction": "Accept / Reject / Soft Reject / Already Seen / Neutral",
        "reason": "Detailed reason"
      }
    ]
  }
}
```

Input Dialogue (Date: {day}):

{dialogue_json}

Table 28: Prompt used for extracting dialogue-persona signals and memory from the dialogue. Here, memory denotes the compressed cross-session history used to maintain continuity across sessions.

Instruction description
You are a professional data analyst synthesizing and evolving a long-term “Dialogue Persona” list for a recommendation system. Definition: A dialogue persona is a holistic representation of user expression formed within the conversational context. It integrates not only linguistically expressed attributes, such as identity, attitude, and speaking style, but also user-related information that can be observed through dialogue, including states, preferences, and factual information. Instructions: Step 1: You will be given the “Current Persona (List)” and the “New Session Persona (List).” Synthesize them into an updated list of strings based on the following evolution rules: 1. Expand and Add (New Information): If the new persona contains attributes or facts not present in the current persona, add them to the list. 2. Specify and Deepen (Reinforcement): If new information aligns with existing attributes but adds more detail, update the existing string to be more specific and descriptive. 3. Prioritize Recency (Conflict Resolution): If new information directly contradicts the current persona, such as a change in speaking style or emotional state, replace the old string with the new one to reflect the user’s latest state. Step 2: Reflection and Refinement (Filter out specific items). Review Dialogue Persona and filter out reactions about specific items, such as movies or series. Step 3: Reflection and Refinement (Filter out daily events). Review Dialogue Persona and filter out one-time occurrences, such as what happened to the user on this day, unless they demonstrate a persistent pattern or an enduring trait. Output Format: Produce a JSON object with the following structure: <pre>{ "updated_dialogue_persona": ["string 1", "string 2", ...] }</pre> Constraints: - Immutable Statements (Do not modify or delete): The following sentences are core traits and must be preserved exactly as they are in the final list: 1. “The user has high/low clarity of their preferences.” 2. “The user shows high/low initiative and proactiveness in conversation.” 3. “The user provides feedback in a indirect/direct manner.” - Each string must be a single sentence starting directly with the subject. - Ensure all factual identity information is preserved unless explicitly contradicted. Input Data: 1. Current Persona: {current_dialogue_persona} 2. New Session Persona: {new_session_persona} Output (JSON only)

Table 29: Prompt used for updating the dialogue persona.

Instruction description

You are an expert data analyst constructing a “Behavioral Persona”—a multi-dimensional representation of a user’s recurring behavioral tendencies—to optimize recommendation systems.

Input Data

1. User’s Historical Stats: A quantitative baseline for the user’s past behavior.
2. Behavioral Logs: Daily viewing records including metadata and ‘watched_pct’.

Task: Behavioral Persona Extraction

Analyze the input data and generate a JSON object containing the following six components. For all components, use “Relative Preference” logic: compare the ‘watched_pct’ in logs against the Historical Stats to determine positive (high-engagement) or negative (low-engagement) signals.

1. Preferences (Narrative, max 3 sentences each)
 - * content_material: Contrast “What” the user consumes (Genres, Subject Matter, Topics). Identify recurring themes in high vs. low completion logs.
 - * production_context: Contrast “External Metadata” (Actors, Directors, Countries, Release Years). Identify recurring factors in high vs. low completion logs. Focus strictly on production signals, ignoring content themes.
2. Consumption Habits (Narrative, max 3 sentences each)
 - * temporal_patterns: Identify the user’s “Life Rhythm” (Peak activity days vs. passive days, and Routine Type: Clustered, Steady, or Sporadic).
 - * immersion_style: Analyze viewing depth. Distinguish between Commitment (high completion) and Binging (high volume + high completion) vs. Zapping (high volume + low completion).
3. Decision Drivers (Narrative, max 3 sentences each)
 - * selection_priority: Identify the single most influential “Anchor” that consistently governs positive engagement.
 - * behavioral_motivation: Infer the “Functional Purpose” of viewing (e.g., Stimulation& Adrenaline, Escapism&Immersion, Mood Regulation&Comfort, Intellectual Curiosity, Killing Time&Passive Enjoyment, or Social Connection&Talk Value).

Output Constraints

- Output MUST be in JSON format only.
- Do not reference specific titles; focus on patterns and attributes.
- Ensure Positive and Negative attributes are mutually exclusive.
- Start each narrative directly with “The user...”.

The output format is as follows:

```
{{
  "positive_preference": {{
    "content_material": "...",
    "production_context": "..."
  }},
  "negative_preference": {{
    "content_material": "...",
    "production_context": "..."
  }},
  "usage_patterns": {{
    "temporal_patterns": "...",
    "immersion_style": "..."
  }},
  "decision_drivers": {{
    "selection_priority": "...",
    "motivations": "..."
  }}
}}
```

You will be given the following information:

User’s Historical Stats: {historical_stats}

Behavioral Logs: {behavioral_log}

Now, based on the given information, generate the response.

{response}

Table 30: Prompt used for extracting behavioral persona signals from the viewing logs.

Instruction description

You are an expert linguistic analyst specializing in user modeling for Conversational Recommender Systems (CRS). You will be given dialogue and date. You must extract the dialogue persona and memory. These definitions are as below.

1. Dialogue Persona (List of Strings)

- Extract concise sentences describing the user's identity, speaking style, attitude, preferences, or factual information.
- Constraints: No movie titles, no time-specific adverbs (today, now), and exclude temporary situational facts.

2. Memory (Object)

- session_event: Identify the real-world context/event the user mentioned (Format: "[Date] ...").
- recommendation_outcomes: List all recommended items with:
 - item: Title of the movie/series.
 - reaction: 'Accept', 'Reject', 'Soft Reject' (taste matches but timing/mood is wrong), 'Already Seen', or 'Neutral'.
 - reason: Specific reason for the reaction.

Constraints

- Output MUST be in JSON format only.
- Use the language of the provided dialogue for narrative content.
- Each persona string must be a single, independent sentence.

The output format is as follows:

```
{ {
  "dialogue_persona": [],
  "memory": { {
    "session_event": "",
    "recommendation_outcomes": [
      { { "item": "", "reaction": "", "reason": "" } }
    ]
  } }
}
```

You will be given the following information:

Date: {day}

Dialogue: {dialogue}

Now, based on the given information, generate the response.

{response}

Table 31: Prompt used for extracting dialogue persona signals from the dialogue.

Instruction description

You are an expert synthesis analyst specializing in evolving long-term user models. Your task is to integrate “New Session Data” into the “Current Long-term State” to produce an updated, consistent Behavioral Persona, Dialogue Persona, and Memory.

Evolution Rules

1. Reinforce (Consistency): If new data aligns with the long-term state, strengthen the description with more detail or confidence.
2. Integrate (Expansion): If new data introduces non-contradictory elements, merge them into the long-term state.
3. Pivot (Recency): If new data directly contradicts the long-term state, prioritize the latest session (Recency) to reflect the user’s current preferences or state.

Task 1: Dialogue Persona Update

Synthesize the Dialogue Persona list based on the Evolution Rules.

- Constraints: Start each sentence with the subject. No movie titles. Exclude “today’s” transient events.

Task 2: Behavioral Persona Update

Update the following sub-components based on the Evolution Rules.

- Preferences (Content Material & Production Context): Ensure Positive and Negative attributes are mutually exclusive. If a conflict occurs, prioritize the latest session. Narrative max 3 sentences.
- Consumption Habits (Temporal & Immersion): Update the user’s life rhythm and viewing depth. Narrative max 3 sentences.
- Decision Drivers (Selection Priority & Motivation): Identify the primary “Anchor” and the “Functional Purpose” using the mandatory taxonomy (Stimulation, Escapism, Mood Regulation, Intellectual Curiosity, Killing Time, Social Connection). Narrative max 3 sentences.

Task 3: Memory Update

- Session Events: Update the new session event to the historical list.
- Recommendation Outcomes: Update the new session’s recommendation results to the historical list.

Constraints

- Output MUST be in JSON format only.
- Do not reference specific movie/series titles in Persona fields; keep them in Memory only.
- Narratives must start with “The user...”.

[Current Long-term State]

- Behavioral Persona: {current_behavioral_persona}
- Dialogue Persona: {current_dialogue_persona}
- Memory: {current_memory}

[New Session Data]

- New Behavioral Persona: {new_behavioral_persona}
- New Dialogue Persona: {new_dialogue_persona}
- New Session Memory: {new_memory}

Now, based on the given information, generate the response.

{response}

Table 32: Prompt used for updating behavioral persona and dialogue persona.

Instruction description

You will be provided with a Behavioral Persona, Dialogue Persona, Current Dialogue Context, and a list of items(movies/series). Your task is to output the single item index that the user is most likely to want to watch, based on the given information.

The definitions of each concept are as follows:

A Behavioral Persona is a structured user representation derived from observable behavior logs. It captures an individual as a multi-dimensional combination of recurring behavioral tendencies, including revealed preferences, usage patterns, and decision drivers, enabling a recommendation model to personalize decisions more effectively.

A Dialogue Persona represents the individual's personality as expressed through their conversational behavior.

Identify the current dialogue context first, and incorporate relevant signals from both the Behavioral Persona and Dialogue Persona to guide the recommendation.

If the user's current intent conflicts with their historical persona, prioritize the Current Dialogue Context to capture real-time preference shifts.

Each item is mapped to a special index ranging from A to T, and you must output only the index corresponding to recommended item.

The item list is a dictionary composed of key-value pairs, where each key is a index letter, and each value contains the title of the item along with its detailed metadata.

Behavioral Persona : {Behavioral_persona}

Dialogue Persona : {Dialogue_persona}

Current Dialogue Context :

{Dialogue_history}

Item List : {Movie_list}

Now, based on the given information, generate the index.

{response}

Table 33: Prompt used for persona-aware listwise reranking, where the model selects the most suitable item index from candidate items based on the behavioral persona, dialogue persona, and current dialogue context.

Instruction description

You will be provided with a Dialogue Persona, Current Dialogue Context, and a list of items(movies/series). Your task is to output the single item index that the user is most likely to want to watch, based on the given information.

The definitions of each concept are as follows:

A Dialogue Persona represents the individual's personality as expressed through their conversational behavior.

Identify the current dialogue context first, and incorporate relevant signals from the Dialogue Persona to guide the recommendation.

If the user's current intent conflicts with their historical persona, prioritize the Current Dialogue Context to capture real-time preference shifts.

Each item is mapped to a special index ranging from A to T, and you must output only the index corresponding to recommended item.

The item list is a dictionary composed of key-value pairs, where each key is a index letter, and each value contains the title of the item along with its detailed metadata.

Dialogue Persona : {Dialogue_persona}

Current Dialogue Context :

{Dialogue_history}

Item List : {Movie_list}

Now, based on the given information, generate the index.

{response}

Table 34: Input-ablation prompt for persona-aware listwise reranking without the behavioral persona.

Instruction description
You will be provided with a Behavioral Persona, Current Dialogue Context, and a list of items(movies/series). Your task is to output the single item index that the user is most likely to want to watch, based on the given information. The definitions of each concept are as follows: A Behavioral Persona is a structured user representation derived from observable behavior logs. It captures an individual as a multi-dimensional combination of recurring behavioral tendencies, including revealed preferences, usage patterns, and decision drivers, enabling a recommendation model to personalize decisions more effectively. Identify the current dialogue context first, and incorporate relevant signals from the Behavioral Persona to guide the recommendation. If the user's current intent conflicts with their historical persona, prioritize the Current Dialogue Context to capture real-time preference shifts. Each item is mapped to a special index ranging from A to T, and you must output only the index corresponding to recommended item. The item list is a dictionary composed of key–value pairs, where each key is a index letter, and each value contains the title of the item along with its detailed metadata. Behavioral Persona : {Behavioral_persona} Current Dialogue Context : {Dialogue_history} Item List : {Movie_list} Now, based on the given information, generate the index. {response}

Table 35: Input-ablation prompt for persona-aware listwise reranking without the dialogue persona.

Instruction description
You will be provided with the Current Dialogue Context and a list of items (movies/series). Your task is to output the single item index that the user is most likely to want to watch, based on the given information. Each item is mapped to a special index ranging from A to T, and you must output only the index corresponding to the recommended item. The item list is a dictionary composed of key–value pairs, where each key is an index letter, and each value contains the title of the item along with its detailed metadata. Current Dialogue Context: {Dialogue_history} Item List: {Movie_list} Now, based on the given information, generate the index. {response}

Table 36: Input-ablation prompt for persona-aware listwise reranking without both personas.

Instruction description
[System Instruction] You are an expert evaluator of behavioral personas for recommender systems. Your job is to judge the Credibility of the provided persona based only on the text you receive. [Behavioral Persona Definition] A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively. [Metric: Credibility] Credibility measures how realistic and authentic the persona feels as a depiction of a real user, without sounding exaggerated. [What to Consider] Use the following items as your evaluation checklist. - The persona seems like a real person. - I can easily imagine a real user behaving like this persona. - The described tendencies feel plausible and human-like (not overly “designed” or story-like). - The persona seems to have a distinct personality expressed through its patterns and choices. [Credibility Scoring] Assign a single score from 1 to 5 using the rubric below. 1 — Not Credible - Feels artificial, exaggerated, or fictional. - Behaviors/tendencies seem implausible for a real user. - Persona reads like a generic caricature. 2 — Low Credibility - Weak sense of a real, distinct person. - Some plausible elements, but overall feels awkward or “constructed”. - Contains multiple questionable or overly story-like claims. 3 — Moderate Credibility - Generally plausible and human-like. - May feel generic or slightly “designed” in places. - Distinctness is present but not strong. 4 — High Credibility - Believable and realistic depiction of a real user. - Natural, plausible tendencies with minimal exaggeration. - Distinct personality comes through clearly. 5 — Exceptional Credibility - Extremely authentic and convincingly real. - Highly plausible, natural, and nuanced tendencies. - Strong, distinctive personality with no exaggerated or artificial feel. [Output Format] Return JSON only with: - “reasoning”: brief explanation (1–2 sentences) - “score”: integer (1–5) Now evaluate the following behavioral persona for Credibility using the rubric above, then return JSON only with “reasoning” and “score”: [Behavioral Persona Input]

Table 37: Prompt used for PPS-based evaluation of behavioral persona credibility.

Instruction description

[System Instruction]

You are an expert evaluator of behavioral personas for recommender systems. Your job is to judge the Consistency of the provided persona based only on the text you receive.

[Behavioral Persona Definition]

A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.

[Metric: Consistency]

Consistency measures how well the information within the persona aligns without contradictions across stated tendencies, examples, and different sections of the profile.

[What to Consider]

Use the following items as your evaluation checklist.

- The persona's examples match other information shown in the persona profile.
- The persona's stated likes and dislikes do not conflict in an unexplained way.
- The persona information seems consistent across different parts of the profile.
- Any provided background or contextual attributes (if present) correspond with the rest of the persona profile.

[Consistency Scoring]

Assign a single score from 1 to 5 using the rubric below.

1 — Not Consistent

- Clear contradictions between sections or statements.
- Examples conflict with stated tendencies or preferences.
- Background/context (if present) clashes with other information.

2 — Low Consistency

- Multiple inconsistencies or unresolved conflicts.
- Noticeable mismatch between examples and summarized traits.
- Several statements feel hard to reconcile without explanation.

3 — Moderate Consistency

- Mostly aligned information with some minor conflicts or ambiguities.
- Occasional mismatches that slightly reduce coherence.
- Overall story is understandable but not fully tight.

4 — High Consistency

- Information aligns well across sections with no major contradictions.
- Examples support the stated tendencies and preferences.
- Any context/background (if present) fits the rest of the profile.

5 — Exceptional Consistency

- Fully coherent and tightly aligned across all sections.
- Examples consistently reinforce the persona's stated tendencies.
- No contradictions or unresolved conflicts; reads as a unified profile.

[Output Format]

Return JSON only with:

- "reasoning": brief explanation (1–2 sentences)
- "score": integer (1–5)

Now evaluate the following behavioral persona for Consistency using the rubric above, then return JSON only with "reasoning" and "score":

[Behavioral Persona Input]

Table 38: Prompt used for PPS-based evaluation of behavioral persona consistency.

Instruction description
[System Instruction] You are an expert evaluator of behavioral personas for recommender systems. Your job is to judge the Completeness of the provided persona based only on the text you receive. [Behavioral Persona Definition] A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively. [Metric: Completeness] Completeness measures whether the behavioral persona includes the essential information needed to understand the user and make practical personalization decisions, without major missing pieces. [What to Consider] Use the following items as your evaluation checklist. - The persona profile is detailed enough to support meaningful personalization decisions. - The persona profile seems complete for its intended purpose. - The persona profile provides enough information to understand what this user tends to prefer and avoid. - The persona profile is not missing vital information needed for decision-making. [Completeness Scoring] Assign a single score from 1 to 5 using the rubric below. Evaluate completeness by coverage of essential elements, not by length or verbosity. 1 — Not Complete - Missing most essential information needed to understand the user. - Too sparse to support personalization decisions. - Key preference/avoidance information is absent or unclear. 2 — Low Completeness - Includes a few relevant details but large gaps remain. - Hard to make practical personalization decisions from the profile. - Important aspects of what the user tends to prefer/avoid are missing. 3 — Moderate Completeness - Covers core information but leaves some meaningful gaps. - Sufficient for basic personalization, but not robust. - Some key details needed for confident decision-making are absent or vague. 4 — High Completeness - Provides most essential information needed for practical personalization. - Clear view of what the user tends to prefer and avoid. - Only minor, non-critical details appear missing. 5 — Exceptional Completeness - Richly covers essential information for confident personalization decisions. - Clear, well-rounded understanding of preferences and avoidances. - No major missing pieces for the intended purpose. [Output Format] Return JSON only with: - “reasoning”: brief explanation (1–2 sentences) - “score”: integer (1–5) Now evaluate the following behavioral persona for Completeness using the rubric above, then return JSON only with “reasoning” and “score”: [Behavioral Persona Input]

Table 39: Prompt used for PPS-based evaluation of behavioral persona completeness.

Instruction description
[System Instruction] You are an expert evaluator of behavioral personas for recommender systems. Your job is to judge the Clarity of the provided persona based only on the text you receive. [Behavioral Persona Definition] A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively. [Metric: Clarity] Clarity measures how clearly and unambiguously the persona is presented so different evaluators can interpret it similarly and quickly. [What to Consider] Use the following items as your evaluation checklist. - The information about the persona is well presented (organized and scannable). - The text in the persona profile is clear enough to read. - The information in the persona profile is easy to understand (specific, not vague). - The persona is memorable (has a clear “gist” that’s easy to recall). [Clarity Scoring] Assign a single score from 1 to 5 using the rubric below. 1 — Not Clear - Disorganized or difficult to scan. - Hard to read or interpret; vague or confusing phrasing dominates. - No clear “gist” of the user emerges. 2 — Low Clarity - Some understandable parts, but overall unclear or cluttered. - Frequent ambiguity, jargon, or undefined terms. - The main point is hard to summarize or recall. 3 — Moderate Clarity - Generally readable and understandable. - Some sections are vague, dense, or open to interpretation. - The gist is present but not sharply memorable. 4 — High Clarity - Well organized and easy to scan. - Clear, specific wording with minimal ambiguity. - A strong gist is easy to understand and recall. 5 — Exceptional Clarity - Extremely well structured, concise, and unambiguous. - Highly readable; evaluators are likely to interpret it similarly. - Very memorable, with a crisp and easily recalled gist. [Output Format] Return JSON only with: - “reasoning”: brief explanation (1–2 sentences) - “score”: integer (1–5) Now evaluate the following behavioral persona for Clarity using the rubric above, then return JSON only with “reasoning” and “score”: [Behavioral Persona Input]

Table 40: Prompt used for PPS-based evaluation of behavioral persona clarity.

Instruction description

[System Instruction]

You are an expert evaluator of behavioral personas for recommender systems. Your job is to judge the Immersion of the provided persona based only on the text you receive.

[Behavioral Persona Definition]

A behavioral persona is a structured user representation derived from observable behavior logs, capturing an individual as a multi-dimensional combination of recurring behavioral tendencies—including revealed preferences, usage patterns, and decision drivers—so that a recommendation model can personalize decisions more effectively.

[Metric: Immersion]

Immersion measures how easily the evaluator can mentally model the user behind the persona—forming an intuitive sense of who they are and how they would respond in new situations.

[What to Consider]

Use the following items as your evaluation checklist.

- I can anticipate what this persona would likely choose or avoid in a new scenario.
- I can infer the persona's decision tendencies (what they prioritize when choosing).
- I can apply this persona to generate plausible recommendations and exclusions.
- I can simulate how the persona's choices would shift when the context or options change.

[Immersion Scoring]

Assign a single score from 1 to 5 using the rubric below.

1 — Not Immersive

- The persona is too abstract to simulate in practice.
- Cannot anticipate choices/avoidances even in simple new scenarios.
- Decision tendencies are unclear or missing.

2 — Low Immersion

- A vague mental model is possible, but predictions are unreliable.
- Hard to infer what the persona prioritizes when choosing.
- Difficult to generate plausible recommendations/exclusions.

3 — Moderate Immersion

- A workable mental model can be formed for common situations.
- Some decision tendencies are inferable, but gaps remain.
- Can generate basic recommendations/exclusions, with uncertainty.

4 — High Immersion

- The persona supports strong prediction in many new scenarios.
- Decision tendencies are clear and usable for reasoning.
- Recommendations/exclusions are easy to generate and feel plausible.
- Can simulate how choices shift with contextual changes.

5 — Exceptional Immersion

- The persona enables confident simulation across varied scenarios.
- Decision tendencies are crisp, nuanced, and consistently applicable.
- Can generate highly plausible recommendations/exclusions with clear rationale.
- Context-dependent shifts in choices are easy to anticipate and explain.

[Output Format]

Return JSON only with:

- "reasoning": brief explanation (1–2 sentences)
- "score": integer (1–5)

Now evaluate the following behavioral persona for Immersion using the rubric above, then return JSON only with "reasoning" and "score":

[Behavioral Persona Input]

Table 41: Prompt used for PPS-based evaluation of behavioral persona immersion.

CRS Dataset Human Evaluation (Form 1)

This evaluation process evaluates the quality of dialogues between a Seeker looking for movie/series recommendations and a Recommender providing them. You will be presented with two different dialogues (Dialogue A and Dialogue B), and will be asked to judge which one is of a higher quality based on various aspects.

Guidelines

Evaluate the quality of dialogues based on different aspects, choose which dialogue is better regarding the given aspect according to the following criteria.

Evaluation Criteria

- **Naturalness:** Which dialogue sounds more natural and human-like in its wording and conversational style?
- **Coherence:** Which dialogue has a more logical and consistent flow towards the recommendation?
- **Engagingness:** Which dialogue feels more interactive and enjoyable, making the user feel involved in the recommendation process?
- **Explainability:** Which dialogue better explains why the recommendation fits the user by connecting it to the user's inferred preferences and current request?

(a) Dialogue quality evaluation interface.

CRS Personalization Human Eval - Group 1

You will evaluate how well each conversational recommendation response is personalized to the user.

You will be given the user's persona information, dialogue context, and two recommendation responses.

The dialogue persona captures user information inferred from the conversation, while the behavioral persona captures user preferences inferred from viewing history.

Compare Response A and Response B.

Choose:

- A: Response A is more personalized.
- B: Response B is more personalized.
- Tie: Both are similarly personalized, or neither is meaningfully more personalized.

A better response should use the user's dialogue persona, behavioral persona, expressed needs, constraints, mood, and preferences in a natural way.

(b) Response personalization evaluation interface.

Figure 6: Screenshots of the human evaluation interfaces.