STEER: STEERABLE RECOMMENDATION SYSTEMS WITH USER MEMORY AND THINK-THEN-RECOMMEND

Anonymous authors
Paper under double-blind review

ABSTRACT

Users are no longer resigned to passive consumption of algorithmic recommendations; they increasingly demand active agency over their digital feeds. Recent product launches—Instagram's "Tune Your Algorithm", Spotify's "Taste Profile", Threads' "Dear Algorithm"—all point to a fundamental gap: mainstream recommendation systems lack a principled mechanism for users to steer recommendations through open, natural-language prompts. We identify a steerable recommendation trilemma: existing paradigms are optimized for at most two of steerability, stateful personalization, and scalability—while hybrid approaches can partially address all three, no prior framework provides a principled, unified solution —collaborative filtering is personalized and scalable but not steerable; text retrieval is steerable and scalable but stateless with limited personalization; LLM-as-recommender is steerable and personalized but not scalable.

We present STEER, a framework that resolves this trilemma through two core innovations: (1) STEER *User Memory*, an LLM-powered user profile that makes the recommendation system's understanding of each user transparent, interpretable, and editable—serving as a shared representation between user and system; and (2) *Think-Then-Recommend* (TTR), a two-stage architecture that explicitly reasons over the user's natural-language steering prompt and memory, then executes a structured recommendation plan via scalable embedding-based retrieval.

We show empirically that personalization (from user memory) and steering (from prompts) provide strong complementary signals: their combination exceeds their individual contributions with large margins. We evaluate STEER at multiple scales: a new human-annotated benchmark (Open-Steer, 2,000+ participants across four domains), large-scale public benchmarks, and online deployment to hundreds of millions of users. Against methods consuming the same steering signal, STEER demonstrates significant gains (e.g., +37.3 points absolute HR@10 from 18.9% to 56.1%, over prompt-only retrieval on our largest dataset), confirming that explicit reasoning over transparent user memory provides value beyond simple query-item matching. Online A/B tests show statistically significant improvements in time spent (+0.40%) and app visits (+0.21%), validating that the framework scales to latency-sensitive production environments.

1 INTRODUCTION

1.1 THE DEMAND FOR STEERABLE ALGORITHMS

A quiet revolution is reshaping the relationship between users and recommendation algorithms. On platforms that lack explicit steering features, users have organically developed workarounds. On Xiaohongshu (RedNote) and TikTok, users strategically issue search queries not only to find specific content, but also hope that their recommendation algorithm would pick up this search interaction. This emergent behavior—using search as a proxy for preference articulation—is a compelling signal that illustrates the latent demand users want direct steering capabilities that the current recommendation paradigm does not provide.

In response to this increasing demand, major platforms are shipping features that let users *tell* the algorithm what they want, rather than relying solely on implicit behavioral signals, which gets part of

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

Table 1: The steerable recommendation tradeoff. Each existing paradigm is architecturally optimized for at most two of the three required properties; achieving the third typically requires ad-hoc extensions that sacrifice core strengths. STEER provides a unified framework that addresses all three by design.

Paradigm	Steerable	Stateful Personalization	Scalable
Collaborative Filtering	✗	✓	✓
Text Retrieval	✓	✗	✓
LLM-as-Recommender	✓	✓	✗
STEER (Ours)	✓	✓	✓

the job done but suffers from consistency, ambiguity and efficiency. That is, it is challenging from the system to differentiate one-off search (e.g. out of short-term curiosity) vs. search as a steering proxy for longer term interests. And even if search-based steering happens, there's no "contract" between the user and the system to ensure consistent delivery of the recommendations, and there's no efficient way for users to revise if the "steering" interpretation was off.

Instagram: Tune Your Algorithm. In early 2026, Instagram launched "Tune Your Algorithm," allowing users to reset their Reels recommendations and provide free-text descriptions of the content they wish to see (Meta Platforms, 2026). The feature was a response to aforementioned growing user demand of recommendation transparency and user agency, and represents one of the first deployments of open-ended natural language control in a billion-user recommendation system.

Spotify: Taste Profile Controls. Spotify has also progressively expanded its user-facing controls beyond simple "like" and "dislike" buttons, introducing genre tuning sliders, mood-based playlist customization, and free-text input for its AI-powered DJ feature (Spotify Technology, 2023). These controls acknowledge that a single embedding vector cannot capture the richness of musical taste.

Threads: Dear Algorithm. Meta's Threads introduced the "Dear Algorithm" feature, inviting users to write open letters to the recommendation system describing desired feed changes. The feature's viral adoption—users publicly sharing their "letters" as posts—demonstrated both the demand for algorithmic control and the naturalness of expressing preferences in free text.

1.2 THE STEERABLE RECOMMENDATION TRILEMMA

Despite this clear industry momentum, no existing recommendation paradigm is architecturally designed to simultaneously satisfy the three requirements that steerable systems demand. While ad-hoc extensions can partially bridge individual gaps (e.g., appending text features to collaborative filtering), they do so without a shared, editable user representation that enables coherent reasoning across all three axes:

1. **Steerability**: The system must accept and faithfully execute open-ended natural-language steering.
2. **Stateful Personalization**: Recommendations must reflect the user's long-term preferences across multiple interests and co-engagement patterns, not just the immediate query.
3. **Scalability**: The system must operate within the latency and compute constraints of production serving over corpora of hundreds of millions of items.

Existing paradigms achieve at most two of these three properties (Table 1): Collaborative filtering models effectively capture preferences through co-engagement patterns but provide no interface for natural-language intent—they are structurally non-steerable. Text retrieval methods (Karpukhin et al., 2020) can match a user's query against item descriptions but are stateless: two users issuing the same prompt receive identical results regardless of their history. LLM-based approaches (Sanner et al., 2023; Kolluri et al., 2026) can reason over both language and user context but require per-candidate inference, making them prohibitively expensive for large corpora.

1.3 OUR CONTRIBUTIONS

We present STEER, a framework that resolves the steerable recommendation trilemma through a principled decomposition into three necessary functions. Our contributions are:

1. **Problem Formalization & Benchmark**: We formalize steerable recommendation as a distinct problem class requiring transparency, editability, and intent-aware personalization. We introduce the **Open-Steer Benchmark**—a human-annotated evaluation set of 2,000+ participants across four domains with categorized steering intent types—to enable rigorous evaluation of steerable systems. (Section E)
2. **STEER User Memory**: An LLM-generated, human-readable user profile that serves as a *shared representation* between user and system. Unlike opaque embeddings, the memory is transparent (users can see what the system believes about them), editable (users can correct misunderstandings), and functional (it directly conditions downstream recommendation). Because a user's edit is materialized as an explicit natural-language intent that conditions retrieval, the memory functions as a live control surface rather than a passive profile; Table 3 quantifies the gain from supplying that intent alongside history. (Section 3)
3. **Think-Then-Recommend (TTR)**: A two-stage architecture that explicitly *reasons* about the user's steering intent in the context of their memory, then translates this reasoning into a structured plan executed by a scalable embedding-based retrieval backend. The decomposition enables: (a) interpretability (the reasoning trace is inspectable), (b) modularity (any retrieval backend can execute the plan), and (c) conflict resolution (the think stage balances stated intent against revealed preferences when they diverge). (Section 4)
4. **Multi-Scale Validation**: We organize baselines into three information-access tiers—history-only, steering-only (dense prompt retrieval), and full synthesis (STEER)—isolating the contribution of each signal. On our largest dataset, STEER (full) achieves 56.1% HR@10 versus 18.9% for prompt-only retrieval and 2.3% for memory-only, confirming that personalization and steering provide complementary signals. We validate across a new human-annotated benchmark (2,000+ participants), a large-scale public dataset, and production deployment to hundreds of millions of users, where online A/B tests show statistically significant gains in time spent (+0.40%) and app visits (+0.21%). (Section 5)

2 RELATED WORK

2.1 SEQUENTIAL RECOMMENDATION

Sequential recommendation models capture temporal patterns in user behavior and define the quality bar for personalized recommendation. GRU4Rec (Hidasi et al., 2016) pioneered recurrent neural networks for session-based recommendation. SASRec (Kang & McAuley, 2018) applied self-attention to model sequential dependencies. HSTU (Zhai et al., 2024) demonstrated that generative sequential models can achieve state-of-the-art quality in industrial settings. These models excel at personalization but cannot incorporate a user's natural-language intent—they predict the next item based on behavioral patterns alone, making them unsuitable for the steerable recommendation use case where users explicitly articulate what they want.

2.2 TEXT-BASED RETRIEVAL FOR RECOMMENDATION

Classical information retrieval methods such as BM25 (Robertson & Zaragoza, 2009) and TF-IDF (Salton & McGill, 1983) can match a user's natural-language prompt against item descriptions, providing a natural baseline for prompt-steerable recommendation. These methods respect the user's stated intent but disregard personalization: two users issuing the same prompt receive identical results regardless of their history, and co-engagement patterns (items consumed together by similar users) are completely ignored. Hybrid approaches combining text retrieval with collaborative signals exist (Zheng et al., 2017) but have not been evaluated in the TYA setting.

2.3 CONTROLLABLE AND STEERABLE RECOMMENDATION

Controllable recommendation aims to give users agency beyond implicit feedback. Traditional interfaces such as "See less often" buttons and domain-specific controls like genre toggles (Milli et al., 2021; Kleinberg et al., 2024; Agarwal et al., 2024; Yang et al., 2019) are rarely used in practice (Cunningham et al., 2025), are often ineffective when they are used (McCrosky et al., 2026), and users still frequently report a lack of agency over their recommender systems (Lukoff et al., 2021;

STEER Memory (User #4821)

This user is a dedicated fan of classic and contemporary science fiction films, with strong preferences for directors like Denis Villeneuve and Christopher Nolan. They consistently rate cerebral, visually ambitious films highly (avg 4.5 stars) and have recently shown growing interest in Korean cinema and psychological thrillers. They tend to avoid romantic comedies and mainstream superhero franchises.

Figure 1: Example STEER User Memory generated from a MovieLens user's rating history. The memory is readable, editable, and serves as context for steering.

2023). Recent work aims to broaden user control by introducing value-based toggles that use LLMs to rerank social media content according to selected values (e.g., "cheerful," "knowledge") (Kolluri et al., 2026; Jahanbakhsh et al., 2025). Relatedly, El Malki et al. (2025) leverage LLMs to create custom Bluesky feeds but emphasize the importance of real-time, in-context control: "rather than treating feed-building as a separate setup activity, future systems should support lightweight, iterative changes directly within the feed interface."

3 STEER MEMORY: LLM-POWERED USER UNDERSTANDING

A central contribution of this work is the STEER *User Memory*, a concise natural-language summary that captures a user's preference profile as understood by the recommendation system. Unlike traditional user embeddings—opaque high-dimensional vectors—a STEER user memory is a human-readable summary that describes what topics, styles, and attributes a user tends to prefer, what they avoid, and how their preferences have evolved. Critically, the memory is designed to be *shared*: the user can read it, edit it, and use it as a starting point for steering prompts.

3.1 STEER USER MEMORY AS A TRANSPARENCY MECHANISM

The STEER User Memory directly addresses the transparency demands driving industry features like Instagram's "Tune Your Algorithm" and Spotify's "Taste Profile". Users often express frustration not just with *what* the algorithm recommends, but with *why*—what does the system think it knows about them? The summary from the user memory makes this explicit. In the steerable recommendation system (Section 4), a user can read their user memory, identify misunderstandings ("I don't actually dislike romantic comedies—I just haven't watched many recently"), and either edit the memory directly or issue a steering prompt that corrects the system's understanding.

We validated the transparency mechanism through multiple rounds of user research: moderated lab studies with dozens of participants and continuous online surveys across tens of thousands of users, measuring perceived control, satisfaction with recommendation quality, and stated preference for the app, with and without STEER. Users exposed to STEER reported a 4.2% statistically significant lift in perceived sense of control over the recommendation system relative to a held-out control group; a regression analysis identified memory accuracy and real-time responsiveness to steering as the two strongest drivers of this lift (full protocol and instrument in Appendix G.4). Survey responses further indicate that the quality of interest summary is critical, accounting for a 10.5% statistically significant difference in reported satisfaction with STEER. Representative feedback: "it makes me feel like I'm in control ... saves me time" (P3); "This is honestly, like, a dream come true ..." (P7).

3.2 MEMORY GENERATION AND SCALING

We generate STEER User Memory via an LLM as a structured summary of a user's interaction history. For each user, we collect their most recent K interactions (e.g. likes, long views, reshares, ratings, purchases, skips, etc) along with item representation from meta data and LLM based enrichment (e.g. title, category, summaries). STEER Memory captures multiple facets of user at a high level, including interests, styles, temporal trends, and avoid patterns. The resulting user memory can take different forms for visualization, depending on the needs of downstream use cases such as being input to recommendation systems, enabling transparency and user agency as human editable summaries and/or interest pills. Figure 1 shows an example summary for a MovieLens user as an illustration.

We establish "Episodic Memory" of the timestamped log of interactions with content enrichment as discussed above to allow richer context for recommendations. Each of the high level attributes such as "interests" in STEER memory is then mapped to a subset of the interaction sequence as "evidence" for grounding, for filtering, de-duping, and ranking the interests based on the aggregated engagement rates and similarity of the interaction sequences in the semantic embedding space, respectively.

Generating STEER User Memory for every recommendation request is prohibitively expensive. We employ a caching strategy where the user memory are precomputed and refreshed when (a) the number of new interactions since the last calculation exceeds a threshold τ, or (b) a fixed time window has elapsed. In our experiments, we set $K = 50$ (interactions for generation) and $\tau = 10$ (refresh threshold). For production deployment, STEER User Memory can be computed asynchronously and stored alongside user profiles, amortizing LLM cost across many recommendation requests.

To further support launching STEER User Memory at scale, we performed Knowledge distillation from a large LLM (e.g. LLaMA 405B) down to a smaller LLM (e.g. LLaMA 3B). As discussed later in the paper, enabling 5-10x cost reduction with on-par performance. In particular, we performed multi-step reasoning / chain-of-thought to ensure high quality of teacher models and careful construction of representative training data with stratified sampling, leading to even stronger STEER user memory 3B LLM compared to vanilla 405B.

4 THINK-THEN-RECOMMEND FOR PERSONALIZED STEERING

The Think-Then-Recommend (TTR) paradigm decomposes steerable recommendation into two explicit stages: a *think stage* that reasons about the user's steering intent, and a *recommend stage* that executes the resulting plan. This separation, inspired by chain-of-thought reasoning in LLMs (Wei et al., 2022), enables interpretability (the reasoning is inspectable), modularity (the recommend stage can use any scoring backend), and composability (multiple steering signals can be reconciled in the think stage before reaching the scorer).

4.1 THINK STAGE: INTENT REASONING

The think stage is implemented as an instruction-tuned LLM whose role is to infer the user's current intent from heterogeneous signals. It conditions on two inputs:

1. The user's **STEER Memory** (Section 3), a structured representation that unifies the user's preference profile summarized from historical behavior, and the user's "Episodic Memory" of interaction history, a temporally ordered sequence of recent behavioral signals with item side enrichment. By folding raw interactions into the memory, $\mathcal{M}_u$ exposes both stationary preferences and recent dynamics through a single interface.
2. An optional **steering prompt** p provided by the user (e.g., "I want to discover classic European cinema" or "show me fewer superhero movies and more indie dramas").

These inputs are combined into an LLM prompt that elicits a reasoning trace

$$r \;=\; \mathrm{LLM}_\theta(\mathcal{M}_u,\, p)\,,$$

where r is a free-form natural language summary of what the user is likely to want at this moment and why. The LLM weighs the steering prompt against the user's stationary preferences encoded in STEER Memory, and interprets the user interaction history as evidence of either a transient or a durable shift in taste. The resulting trace r serves as the sole interface to the recommend stage.

When no steering prompt is provided ($p = \varnothing$), the think stage reasons solely over $\mathcal{M}_u$, defaulting to the user's established preferences. This ensures that STEER remains competitive with non-steerable baselines even in the absence of explicit steering.

4.2 RECOMMEND STAGE: REASONING EXECUTION

The recommend stage consumes the reasoning trace from the think stage and returns a candidate list of N items via retrieval. Ranking, blending, and policy-level constraints are deferred to downstream

stages of the serving pipeline; here we focus exclusively on producing a high-quality candidate set that faithfully reflects the inferred intent.

In principle, several backends are compatible with the think stage's output: the trace could be parsed into feature-level adjustments over a base recommender, or used to condition an LLM that re-scores a recall set. Both alternatives, however, either lossily project the free-form reasoning onto a fixed schema or incur an LLM invocation per candidate. We instead adopt an *embedding-based retrieval* formulation, in which the reasoning trace is encoded into a dense intent vector and matched against a precomputed content index.

To realize this formulation, we adapt the Think-Then-Embed (TTE) framework of (Cui et al., 2026) so that the reasoning produced by the think stage is seamlessly absorbed into the retrieval embedding rather than treated as an auxiliary input. Specifically, TTE's reasoning-conditioned encoder ψ_θ is re-purposed to consume the trace r, the user's STEER Memory $\mathcal{M}_u$, and the steering prompt p, projecting them jointly into a unit-norm intent embedding

$$\mathbf{z} \;=\; \psi_\theta(r,\, \mathcal{M}_u,\, p) \;\in\; \mathbb{R}^d, \qquad \|\mathbf{z}\|_2 = 1.$$

Embedding $\mathcal{M}_u$ and p alongside r allows the encoder to recover personalization and steering nuances that may have been abstracted away during the generation of r, ensuring that both the user's preference profile and user-specified intent are preserved verbatim in the retrieval space. The same encoder ψ_θ is also used as the content encoder to embed every item in the corpus $\mathcal{I}$ into the same space offline, and the top-N candidates are retrieved by cosine similarity:

$$\mathcal{R}_N(\mathbf{z}) = \underset{i \in \mathcal{I}}{\text{Top-N}} \left\langle \mathbf{z},\, \psi_\theta(i) \right\rangle.$$

The encoder ψ_θ is trained with a contrastive objective that aligns intent embeddings with the embeddings of items the user subsequently engaged with. Concretely, given a batch of B (user, positive item) pairs $\{(u_b, i_b^+)\}_{b=1}^B$ with in-batch negatives, we minimize the symmetric InfoNCE loss

$$\mathcal{L} \;=\; -\frac{1}{B}\sum_{b=1}^{B} \log \frac{\exp\big(\langle \mathbf{z}_b, \psi_\theta(i_b^+)\rangle/\tau\big)}{\sum_{j=1}^{B} \exp\big(\langle \mathbf{z}_b, \psi_\theta(i_j^+)\rangle/\tau\big)},$$

where τ is a learned temperature.

Our formulation offers two practical benefits. First, the reasoning trace is preserved as a dense representation rather than collapsed onto a hand-designed projection, retaining the expressiveness of the upstream LLM. Second, retrieval reduces to an approximate nearest-neighbor lookup over a standard vector index, which is sublinear in corpus size and incurs no per-candidate LLM invocations, leaving overall serving latency dominated by a single forward pass through think stage.

5 EXPERIMENTS

We evaluate STEER across two key components: (1) the STEER User Memory inferred interest quality, which as highlighted before is critical for the user satisfaction with STEER, the foundation on enabling the sense of transparency, and the key input to Think-Then-Recommend to enable personalized steering; (2) Think-Then-Recommend recommendation quality, this consists of extensive evaluation comparing against strong recommendation model baseline such as HSTU, across multiple scale ranging from thousands of users to hundreds of millions of users, including user surveys, public benchmarks, and online testing from real production traffic.

5.1 DATASETS

We evaluate STEER across three diverse settings: a new human-annotated benchmark, a large-scale synthetic dataset, and a massive production environment. Detailed data statistics and collection protocols are provided in Appendix E.

Open-Steer Benchmark (New benchmark). We introduce the Open-Steer Benchmark to evaluate open-ended recommendation steering. It consists of human-annotated data from over 2,000 participants across four domains: Movies, Books, TV Shows, and Songs. Each entry contains a user's *interaction history*, a free-text *steering prompt* (e.g., "I want mind-bending thrillers"), and a set of

3–10 *ground-truth matches* that satisfy the prompt but are not in the history. This benchmark will be released publicly to facilitate research in steerable systems.

GISTBench (Fostiropoulos et al., 2026). To assess performance on synthetic but large-scale behavioral data, we use GISTBench. It provides privacy-preserving interaction logs (watch time, skips, likes) from a short-form video platform, enriched with semantic content descriptors.

Production Dataset with Real Steering Prompt. We evaluate STEER on a proprietary dataset from a major social media platform comprising 6.5M users with real steering prompts. This dataset includes multi-faceted user profiles and interaction histories (UIH) annotated with engagement types and temporal tags. We use semantic descriptors of posts to allow the LLM to reason over user interests. Steering prompts in this dataset are real user-written inputs collected from the platform's steering interface. For offline evaluation, we pair each user's steering prompt with their subsequent engagement in a held-out temporal window, treating engaged content that aligns with the stated intent as ground-truth positives. To ensure a reliable ground-truth signal for intent alignment, we further validate them against human judgment (details in Appendix E).

5.2 EXPERIMENTAL FRAMEWORK AND BASELINES

To rigorously evaluate the contribution of each signal, we categorize our baselines into three tiers based on the information they consume. This framework allows us to isolate the value of behavioral history, explicit steering, and the "Think" stage synthesis.

Tier 1: History-only Baselines (Passive) These models represent the traditional recommendation paradigm where the system must infer intent solely from past interactions ($\mathcal{H}$).

- **SASRec & HSTU**: State-of-the-art sequential models that rely on interaction IDs. These serve as behavioral anchors to demonstrate how purely passive systems perform when evaluated against specific, active steering intents.
- **STEER (w/o prompt)**: A semantic "History-only" baseline. Unlike ID-based models, this variant uses our LLM-generated User Memory to reason over the user's long-term style and interests. It serves as a more competitive behavioral baseline that shares the same semantic item representations as our full system.

Tier 2: Prompt-only Baselines (Active) These models ignore historical context and rely strictly on the user's explicit steering signal (p).

- **Prompt-only Dense Retrieval**: This baseline performs dense retrieval using only the raw steering prompt. It evaluates whether the task can be solved by simple semantic matching without considering the user's historical profile or preferences.

Tier 3: The STEER Framework (Full Synthesis)

- **STEER (Full)** is our proposed solution. It consumes both the STEER Memory (History) and the Steering Prompt (Intent). By employing the *Think-Then-Recommend* paradigm, it synthesizes these dual signals to generate a recommendation plan that is both contextually relevant to the user's identity and precisely aligned with their current steering request.
- **STEER (No Think)** isolates the contribution of the explicit thinking step. This ablation encodes the raw concatenation of user memory and the steering prompt, holding the encoder, item descriptors, training data, and retrieval objective identical to the full model.

5.3 METRICS

We evaluate recommendation quality using four standard metrics at cutoff $K = 10$: **Hit Rate (HR@K)**, the fraction of users for whom at least one relevant item appears in the top-K. **Normalized Discounted Cumulative Gain (NDCG@K)**, which assigns higher credit to relevant items ranked earlier. **Precision (P@K)**, the fraction of top-K items that are relevant. **Recall (R@K)**, the fraction of all relevant items that appear in the top-K.

5.4 STEER MEMORY RESULTS

The STEER User Memory (Section 3) requires high-quality generation at serving time. We train a compact LLaMA 3.2 3B model for memory generation using a STEER-informed training curriculum, involving multi-task data augmentation (48+ synthetic task variants) and engagement-weighted stratified sampling aligned with STEER objectives.

We evaluate memory quality on a held-out golden set of user primary interests (N=40,286 users), using an LLM judge for semantic precision and recall. As shown in Table 2, our STEER-trained 3B model achieves an F1 score of 0.725. This significantly outperforms the vanilla LLaMA 405B (0.528) despite the latter having over $100\times$ more parameters. In production environments, this efficiency translates to a $6\times$ reduction in serving latency compared to the 405B variant.

Table 2: Memory generation quality on primary interest golden set.

Model	Precision	Recall	F1	Rel. Cost
LLaMA 3B SFT (STEER)	**0.765**	**0.714**	**0.725**	$1\times$
LLaMA 405B (vanilla)	0.739	0.451	0.528	$100\times$
LLaMA 3B (vanilla)	0.588	0.397	0.428	$1\times$

Removing the STEER training framework while maintaining the 3B architecture results in a vanilla baseline with an F1 of 0.428. The performance gap (0.725 vs. 0.428) is primarily driven by a collapse in recall ($0.714 \rightarrow 0.397$), confirming that high-quality steerable memory depends on our specific training framework rather than mere model scale (Cui et al., 2026).

5.5 THINK-THEN-RECOMMEND RESULTS

Table 3 presents recommendation results across the three datasets. STEER (full) achieves the best performance on every dataset and metric by a substantial margin.

Across all three datasets, STEER (full) consistently outperforms sequential baselines, with the gap widening as the content pool and interaction histories grow. The w/o prompt configuration validates STEER Memory as an effective user representation in its own right on Open-Steer and the internal corpus (see Appendix C.1 for the GISTBench exception), while the steering prompt provides the largest marginal gain, particularly when collaborative filtering signals are diluted by corpus size. The complementary nature of the two inputs confirms the benefit of fusing them within a single thinking stage. We additionally ablate the explicit reasoning trace with a no-think variant. Results show that removing thinking degrades every metric on all three datasets, with the largest drops on the internal corpus, where memory is longest and most heterogeneous, confirming that the gains stem from intent-aware memory selection rather than from additional input alone. The thinking stage acts as an intent-conditioned filter over memory, selecting and contextualizing the history segments relevant to the active steering intent; without it, naive concatenation admits substantial noise from unrelated memory.

5.6 ONLINE RESULTS

We deployed the STEER system in the production recommendation system of a large-scale social media platform, where users can view and edit their memory with natural languages, to enable recommendation transparency and user agency. We report the results of a rigorous A/B test conducted over a multi-month period (details in Appendix G).

Compared to the control group, the introduction of explicit user steering yielded statistically significant positive movements across key engagement metrics with outsized wins [1]:

- **App-level Time Spent**: We observed a $+0.40\%$ ($\pm 0.37\%$) increase in overall time spent on the platform. This suggests that providing users with direct algorithmic control improves the recommendation quality and diversity, and more satisfaction.

[1] Due to the very large user base, small percentage of improvements corresponds to changing behavior of millions of users; for example, typical launches are of 20-30% of the impact from the STEER launch.

Table 3: Recommendation quality categorized by information access tiers across three datasets. Tier 1 methods use only history ($\mathcal{H}$), Tier 2 uses only the steering prompt (p), and Tier 3 (Full) synthesizes both. Best in **bold**, second best underlined.

		Open-Steer				GISTBench				Internal			
Tier	**Method**	HR@10	NDCG@10	P@10	R@10	HR@10	NDCG@10	P@10	R@10	HR@10	NDCG@10	P@10	R@10
Tier 1: History-only	SASRec (Kang & McAuley, 2018)	17.729	4.705	3.409	5.158	0.942	0.129	0.098	0.126	0.024	0.014	0.002	0.022
	HSTU (Zhai et al., 2024)	17.362	6.744	4.575	7.092	0.747	0.100	0.077	0.104	0.024	0.010	0.002	0.018
	STEER (w/o prompt)	27.510	7.560	5.120	7.710	0.120	0.020	0.010	0.020	2.332	0.806	0.399	0.979
Tier 2: Prompt-only	STEER (prompt only)	23.670	6.140	3.980	6.470	6.940	1.620	0.880	1.950	18.858	9.584	2.221	14.570
Tier 3: Full	STEER (no think)	26.440	6.740	4.350	7.340	8.830	1.990	1.110	2.410	27.600	14.980	3.320	21.720
	STEER (full)	**36.930**	**11.430**	**7.550**	**11.610**	**9.580**	**2.170**	**1.220**	**2.630**	**56.124**	**35.532**	**7.190**	**43.710**

Point estimates only; bootstrap standard deviations for every metric are reported in Table 5 (Appendix F).

- **App Visits**: The treatment group demonstrated a $+0.21\%$ ($\pm 0.19\%$) lift in app-level visits, indicating that the availability of a steering mechanism encourages users to return to the application more frequently, due to the transparency and user agency from the new recommendation paradigm.

Surveyed across tens of thousands of users, the treatment group also reported a $\sim 4.2\%$ statistically significant lift in perceived sense of control over the recommendation system, with *memory accuracy* and *real-time responsiveness to steering* identified as the top two driving factors.

These online results confirm that STEER is not only highly effective in offline benchmarks but also successfully scales to latency-sensitive production environments, driving meaningful improvements in user satisfaction and overall ecosystem health.

6 Conclusion and Future Work

We identified the *steerable recommendation trilemma*—existing paradigms achieve at most two of steerability, stateful personalization, and scalability—and presented STEER, a framework that resolves it through two complementary innovations. STEER User Memory provides a transparent, editable, shared representation of the system's user understanding, enabling meaningful user agency for the first time in large-scale recommendation. Think-Then-Recommend decomposes intent reasoning from retrieval execution, allowing the system to balance stated user intent against revealed behavioral preferences through explicit, inspectable reasoning.

Our key empirical finding is that personalization and steering provide strong complementary signals: combining user memory with steering prompts through a thinking stage yields gains that exceed their individual contributions with large margins. This holds consistently across a new human-annotated benchmark (Open-Steer), large-scale behavioral data (GISTBench), and a production dataset of 6.5M users and 22.8M items. Online deployment to hundreds of millions of users confirms that these offline gains translate to statistically significant improvements in engagement (+0.40% time spent, +0.21% app visits), validating that the framework operates within production latency constraints.

We acknowledge several limitations that suggest directions for future research:

- **LLM inference cost at the think stage.** The think stage requires a forward pass through a 2B-parameter LLM per recommendation request. While we amortize user memory generation through caching, the per-request reasoning cost remains non-trivial. Distillation of the think stage into a smaller model, or caching reasoning traces for common prompt patterns, are promising directions for further cost reduction.
- **Memory staleness and drift.** The current caching strategy refreshes user memory after $\tau = 10$ new interactions or a fixed time window. For users with rapidly evolving interests (e.g., during major life events or trending topics), this refresh rate may be insufficient. Adaptive refresh policies that trigger on detected preference shifts are a natural extension.
- **Adversarial and edge-case steering.** Appendix H describes the mechanisms STEER uses to resolve intra-prompt contradictions, cross-session conflicts, and underspecified prompts. We have not, however, systematically evaluated these mechanisms, nor have we evaluated robustness to adversarial prompts, which remains as critical future work we plan to pursue.

AI USE STATEMENT

Large language models are a core, non-standard component of the method proposed in this work rather than an incidental tool: they generate the STEER User Memory (Section 3) and perform intent reasoning in the think stage of Think-Then-Recommend (Section 4). The models used, their sizes, the training curriculum, and all hyperparameters are described in Section 5.4 and Appendix A. We additionally used generative AI tools for writing assistance—copy-editing, rephrasing, and LaTeX formatting—and for routine code assistance in the evaluation harness. We did not use generative AI tools to generate research ideas, to produce experimental results, or to write the literature review; all cited work was located and read by the authors. All AI-assisted text and code were reviewed and verified by the authors, and all reported numbers were produced by the experimental pipeline described in the paper. We take responsibility for the final content of this work, including its text, claims, and artifacts.

ETHICS STATEMENT

This work involves human subjects and production user data, and we describe the safeguards applied to each.

Human subjects. The Open-Steer Benchmark was collected from 2,059 survey participants (Appendix E), and the transparency findings in Section 3.1 draw on lab studies and online surveys. Participation was voluntary and informed, and the studies were conducted under internal institutional review processes that govern participant privacy and consent. We report only aggregate results.

Privacy. In the internal production dataset, all user and post identifiers are hashed with a salted SHA-256 algorithm, and no personally identifiable information is passed to the STEER LLM components; posts are represented to the think stage by platform-level topical tags and semantic embeddings rather than raw content.

Societal impact. STEER is intended to increase user agency by making the system's understanding of each user transparent and editable, which can mitigate algorithmic opacity and filter bubbles. We also acknowledge risks: steering could be used to reinforce existing beliefs and construct echo chambers more efficiently; the LLM components may inherit biases from pre-training data, affecting which interests are recognized or prioritized; and transparent user profiles pose privacy risks if improperly secured. Production deployment includes safety filters and the conflict-resolution mechanisms of Appendix H, but as noted in Section 6 we have not systematically evaluated robustness to adversarial steering prompts, and we view a principled study of steering robustness as necessary future work.

Existing assets. We use the GISTBench dataset (Fostiropoulos et al., 2026) and the LLaMA and Qwen (Bai et al., 2025) model families in accordance with their respective licenses, with citations given in Section 5 and Appendix A.

REPRODUCIBILITY STATEMENT

We have aimed to make every reported result reproducible from the paper itself. Appendix A specifies the encoder backbone, the fine-tuning protocol (frozen vision encoder, LoRA adaptation, bf16 precision), the optimizer and learning-rate schedule, the sequence length, the batch size, and the exact GPU configuration used for each dataset; baselines use the implementations and hyperparameters from their original papers. Appendix E gives the full collection protocol for the Open-Steer Benchmark, the filtering and label definitions used for GISTBench, and the anonymization, enrichment, and temporal-split procedure for the internal production dataset. Appendix F reports the complete results of Table 3 with bootstrap standard deviations for every metric, and Section C.2 reports the backbone-scale and UIH-length ablations. The Open-Steer Benchmark and the source code for the STEER framework will be released publicly with documentation and usage instructions. The internal production dataset and the corresponding online A/B results cannot be released, as they derive from proprietary user data; the two external datasets are sufficient to reproduce the paper's central claim that memory and steering contribute complementary signals.

REFERENCES

Arpit Agarwal, Nicolas Usunier, Alessandro Lazaric, and Maximilian Nickel. System-2 recommenders: Disentangling utility and engagement in recommendation systems via temporal point-processes. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency*, pp. 1763–1773, 2024.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Xuanming Cui, Jianpeng Cheng, Hong-You Chen, Satya Narayan Shukla, Abhijeet Awasthi, Xichen Pan, Chaitanya Ahuja, Shlok Kumar Mishra, Taipeng Tian, Qi Guo, Ser-Nam Lim, Aashu Singh, and Xiangjun Fan. Think then embed: Generative context improves multimodal embedding. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=AKXXMK5YTI.

Tom Cunningham, Sana Pandey, Leif Sigerson, Jonathan Stray, Jeff Allen, Bonnie Barrilleaux, Ravi Iyer, Mohit Kothari, Behnam Rezaei, Sanjay Kairam, et al. Ranking by engagement and non-engagement signals: Learnings from industry. *Annals of the New York Academy of Sciences*, 2025.

Omar El Malki, Marianne Aubin Le Quéré, Andrés Monroy-Hernández, and Manoel Horta Ribeiro. Bonsai: Intentional and personalized social media feeds. *arXiv preprint arXiv:2509.10776*, 2025.

Iordanis Fostiropoulos, Muhammad Rafay Azhar, Abdalaziz Sawwan, Boyu Fang, Yuchen Liu, Jiayi Liu, Hanchao Yu, Qi Guo, Jianyu Wang, Fei Liu, et al. Gistbench: Evaluating llm user understanding via evidence-based interest verification. *arXiv preprint arXiv:2603.29112*, 2026.

Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2016.

Farnaz Jahanbakhsh, Dora Zhao, Tiziano Piccardi, Zachary Robertson, Ziv Epstein, Sanmi Koyejo, and Michael S. Bernstein. Value alignment of social media ranking algorithms. *arXiv preprint arXiv:2509.14434*, 2025.

Dietmar Jannach, Ahtsham Manzoor, Wanling Cai, and Li Chen. A survey on conversational recommender systems. *ACM Computing Surveys*, 54(5):1–36, 2021.

Chenyan Jia, Michelle S. Lam, Minh Chau Mai, Jeffrey T. Hancock, and Michael S. Bernstein. Embedding democratic values into social media AIs via societal objective functions. *Proc. ACM Hum.-Comput. Interact.*, 8(CSCW1), 2024.

Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, pp. 197–206, 2018.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781, 2020.

Omar Khattab and Matei Zaharia. ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 39–48, 2020.

Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. The challenge of understanding what users want: Inconsistent preferences and engagement optimization. *Management Science*, 70(9): 6336–6355, 2024.

Akaash Kolluri, Renn Su, Farnaz Jahanbakhsh, Dora Zhao, Tiziano Piccardi, and Michael S. Bernstein. Alexandria: A library of pluralistic values for realtime re-ranking of social media feeds. In *Proceedings of the International Conference on Web and Social Media (ICWSM)*, 2026.

Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

Xiaoxi Li, Jiajie Jin, Yujia Zhou, Yuyao Zhang, Peitian Zhang, Yutao Zhu, and Zhicheng Dou. From matching to generation: A survey on generative information retrieval. *ACM Transactions on Information Systems*, 43(3), 2025.

Kai Lukoff, Ulrik Lyngs, Himanshu Zade, J. Vera Liao, James Choi, Kaiyue Fan, Sean A. Munson, and Alexis Hiniker. How the design of YouTube influences user sense of agency. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021.

Kai Lukoff, Ulrik Lyngs, Karina Shirokova, Raveena Rao, Larry Tian, Himanshu Zade, Sean A. Munson, and Alexis Hiniker. SwitchTube: A proof-of-concept system introducing "adaptable commitment interfaces" as a tool for digital wellbeing. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, 2023.

Jesse McCrosky, Ranadheer Malla, Aapo Tanskanen, and Chico Camargo. Does this button work? investigating YouTube's ineffective user controls. In *Proceedings of the ACM Web Conference 2026 (WWW '26)*, pp. 1563–1573. ACM, 2026. doi: 10.1145/3774904.3792183.

Meta Platforms. Control your Instagram Reels algorithm. `https://about.instagram.com/blog/announcements/reels-algorithm-control`, 2026. Instagram Blog. Accessed: 2026-03-01.

Smitha Milli, Luca Belli, and Moritz Hardt. From optimizing engagement to measuring value. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, pp. 714–722, 2021.

Ronak Pradeep, Kai Hui, Jai Gupta, Adam D Lelkes, Honglei Zhuang, Jimmy Lin, Donald Metzler, and Vinh Q Tran. How does generative retrieval scale to millions of passages? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1305–1321, 2023.

Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Tran, Jonah Samost, Maciej Kula, Ed Chi, and Maheswaran Sathiamoorthy. Recommender systems with generative retrieval. In *Advances in Neural Information Processing Systems*, volume 36, pp. 10299–10315, 2023.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 4(1–2):1–174, 2009.

Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, New York, 1983.

Scott Sanner, Krisztian Balog, Filip Radlinski, Ben Wedin, and Lucas Dixon. Large language models are competitive near cold-start recommenders for language- and item-based preferences. *arXiv preprint arXiv:2307.14225*, 2023.

Spotify Technology. Spotify debuts a new AI DJ, right in your pocket. `https://newsroom.spotify.com/2023-02-22/spotify-debuts-a-new-ai-dj-right-in-your-pocket/`, 2023. Spotify Newsroom. Accessed: 2026-03-01.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022.

Likang Wu, Zhi Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, Qi Liu, Hui Xiong, and Enhong Chen. A survey on large language models for recommendation. *World Wide Web*, 27(5):60, 2024.

Longqi Yang, Michael Sobolev, Yu Wang, Jenny Chen, Drew Dunne, Christina Tsangouri, Nicola Dell, Mor Naaman, and Deborah Estrin. How intention informed recommendations modulate choices: A field study of spoken word content. In *The World Wide Web Conference (WWW)*, pp. 2169–2180, 2019.

Jiaqi Zhai, Lucy Liao, Xing Liu, Yueming Wang, Rui Li, Xuan Cao, Leon Gao, Zhaojie Gong, Fangda Gu, Michael He, Yinghai Lu, and Yu Shi. Actions speak louder than words: Trillion-parameter sequential transducers for generative recommendations. *arXiv preprint arXiv:2402.17152*, 2024.

Lei Zheng, Vahid Noroozi, and Philip S. Yu. Joint deep modeling of users and items using reviews for recommendation. *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining (WSDM)*, pp. 425–434, 2017.

A Implementation Details

We instantiate ψ_θ from Qwen-3-VL (Bai et al., 2025). The model is fine-tuned separately on each of the three datasets with standard InfoNCE loss. The vision encoder is frozen, and the language backbone is adapted with LoRA. Training is performed in bf16. For the internal production dataset we train on 64 NVIDIA A100 GPUs, and for the two external datasets on 32 A100 GPUs. We set the maximum sequence length to 16,384 tokens and optimize with AdamW using a learning rate of 2×10^{-4}, $\beta = (0.9, 0.95)$, and weight decay of 0.1. All models are trained for a single epoch with a global batch size of 1024. All sequential baselines use the implementations and hyperparameters from their respective papers, tuned on our datasets.

B User Memory Details

We generate STEER User Memory via an LLM as a structured summary of a user's interaction history. For each user, we collect their most recent K interactions (e.g. likes, long views, reshares, ratings, purchases, skips, etc) along with item representation from meta data and LLM based enrichment (e.g. title, category, summaries). STEER Memory captures multiple facets of user at a high level, including interests, styles, temporal trends, and avoid patterns.

1. **Primary interests**: dominant categories and topics the user engages with (e.g., "science fiction novels," "indie rock albums"). We also make the interest hierarchical with varying granularity, which ranges from topics, subtopics, named entities, keywords (e.g. Sports | Basketball | Steph Curry | Steph Curry highlights, interviews, playing golf, time with family, etc).

2. **Style preferences**: attributes such as informational, entertaining, funny, commercial, price sensitivity, brand loyalty, aesthetic taste, or content quality / complexity.

3. **Temporal trends**: how preferences have evolved recently (e.g., "shifting from action to documentary films"), rising interest in "FIFA world cup" recently for a user with primary interests in "Soccer", "NBA Playoffs" for a basketball fan, "Met Gala" for a user who is big on "Fashion" or interested in "Celebrity". This also includes seasonality (e.g. Winter vs Summer, holidays) and contextual preferences (e.g. day vs night, week day vs weekend).

4. **Avoidance patterns**: categories or attributes that the user consistently ignores or rates negatively. this is particularly important for recommendation systems, as they heavily rely on co-engagement patterns and are inherently poor at modeling negative events. With LLM reasoning, we are able to better interpret and cross-check different user behavioral signals.

C Additional Results

C.1 Results by Dataset

Open-Steer dataset. STEER without a steering prompt outperforms both sequential baselines by 55% relative in HR@10 (27.51 vs 17.73), indicating that reasoning over STEER Memory and UIH alone can achieve substantial retrieval gains. With steering enabled, STEER (full) attains 36.93 HR@10 and 11.43 NDCG@10, representing 108% and 70% relative improvements over the strongest baseline in each metric respectively. The no-think ablation is informative here: at 26.44 HR@10 it sits slightly below the history-only configuration (27.51) despite consuming strictly more information. With only five history items and a short prompt, naive concatenation gives the encoder no way to weight the two signals; the think stage bridges the gap and leads to a 10.5-point margin over the same inputs.

GISTBench. SASRec (Kang & McAuley, 2018) and HSTU (Zhai et al., 2024) both fall below 1% HR@10. STEER (prompt only), equivalent to text search with no personalization, lifts HR@10 to 6.94, and the full configuration reaches 9.58. This highlights the value of explicit steering when collaborative filtering signals are sparse. We note that GISTBench is the one setting where STEER (w/o prompt) is not competitive, reaching 0.12 HR@10 against SASRec's 0.94. Memory-only retrieval is therefore not uniformly strong across domains, and we scope the claim accordingly: on this dataset the steering signal, not the memory, carries performance. Characterizing which content domains support effective memory-only retrieval is left to future work.

Internal dataset. With a pool of over 22M items and longer interaction histories, the gap between sequential baselines and STEER widens further. SASRec (Kang & McAuley, 2018) and HSTU (Zhai et al., 2024) achieve 0.02% HR@10, while STEER (w/o prompt) reaches 2.33%, confirming that STEER Memory provides an effective user representation even without steering. STEER (full) achieves 56.12% HR@10 and 35.53% NDCG@10. The tier progression is clearest here: prompt-only reaches 18.86, adding history without the think stage reaches 27.60 (+8.7 points), and the full configuration reaches 56.12 (+28.5 points over no-think, +37.3 over prompt-only). That the largest single jump comes from adding the think stage — not from adding history — indicates the benefit lies in intent-conditioned selection over memory rather than in the additional input itself.

C.2 Ablation: Model Scaling

We investigate two scaling axes on the internal dataset under the full STEER configuration: (1) backbone capacity, comparing the 2B-parameter Qwen-3-VL used in our main experiments with an 8B-parameter variant, and (2) the length of user engagement history (UIH) supplied to the think stage, sweeping the K most recent UIH entries from 10 to full.

Table 4: Ablations on the internal benchmark: backbone scale (2B vs. 8B, both with full UIH) and UIH length K at fixed 2B backbone. Best per group in bold.

	Variant	HR@10	NDCG@10	P@10	R@10
Backbone	2B (full UIH)	56.124	35.532	7.190	43.710
	8B (full UIH)	**63.499**	**40.928**	**8.256**	**49.487**
UIH length (2B)	K=10	30.470	17.100	3.670	23.700
	K=20	32.320	18.450	3.900	24.930
	K=50	36.580	21.330	4.530	27.700
	K=100	42.240	25.240	5.340	31.850
	Full UIH (= 2B above)	**56.124**	**35.532**	**7.190**	**43.710**

Scaling from 2B to 8B improves HR@10 by 7.4 percentage points and NDCG@10 by 5.4 percentage points, suggesting that the think stage's reasoning quality remains a bottleneck at the 2B scale and benefits from additional model capacity. Truncating UIH degrades HR@10 monotonically, leading to a 25.6 percentage-point gap between full history (56.1) and K=10 (30.5). NDCG@10 follows the same pattern, falling from 35.5 (full) to 17.1 (K=10). This indicates that richer engagement context contributes to retrieval quality. We report 2B with the full UIH as our primary configuration given the more favorable latency profile for production deployment.

D More Related Research

D.1 LLM-based Recommendation

There has been increasing research on leveraging LLMs for recommendation (Wu et al., 2024) and information retrieval more broadly (Karpukhin et al., 2020; Khattab & Zaharia, 2020). Prior work has used LLMs to directly score item relevance based on natural-language preferences (Sanner et al., 2023), but the high computational cost limits such approaches to re-ranking small candidate sets (Jia et al., 2024; Kolluri et al., 2026). Generative retrieval methods train models to directly generate item identifiers (Rajput et al., 2023; Li et al., 2025), but face scalability challenges with large corpora (Pradeep et al., 2023). A key distinction in STEER is the role of the LLM: rather than using it as the sole recommender, the LLM first constructs a *user memory*—an explicit preference profile—and then reasons over this memory and the steering prompt, with recommendation output produced through generation, embedding-based retrieval, or re-ranking by a sequential model. This separation enables STEER to leverage both the language understanding of LLMs and the personalization strength of sequential models.

D.2 User Modeling with Natural Language

Traditional user modeling represents preferences as embedding vectors or interaction sequences (Koren et al., 2009). Conversational recommendation systems build user models through dialogue (Jannach et al., 2021). Our *user memory* concept extends this line of work by using an LLM to construct explicit, natural-language preference profiles from consumption histories. Unlike embedding-based representations, user memory is readable, editable, and directly usable in downstream reasoning.

E Dataset Details

In this section, we provide exhaustive details regarding the datasets used for evaluation, including the data collection protocol for the Open-Steer Benchmark and the preprocessing steps for the Internal Production Dataset.

E.1 Open-Steer Benchmark Collection

The Open-Steer Benchmark was developed to address the lack of human-centric evaluation sets for open-ended recommendation steering. We conducted a large-scale public survey to capture diverse user intents.

Survey Design Participants were presented with a "strict evaluation triad" to ensure the resulting data could be used for rigorous benchmarking:

1. **Interaction History**: Each participant listed exactly five items (movies, books, etc.) they had recently consumed and enjoyed. This provides the STEER system with a baseline user profile. Compared with short-form video or social media, natural consumption frequencies for discrete media (movies/books/songs) are inherently lower. Setting history length K=5 isolated history-length confounding variables and maximized data quality given a fixed survey budget.
2. **Open-Ended Steering Prompt**: Users were asked to write a natural language request for what they wanted to see *next*. We explicitly encouraged nuance (e.g., "I want a book that feels like a cold winter morning in Scandinavia").
3. **Ground-Truth Matches**: Participants provided 3–10 items that matched their prompt but **excluded** items from their history or those mentioned in their prompt. This ensures the benchmark evaluates the system's ability to discover *new* content that fulfills a specific intent.

Participant Demographics We collected responses from 2,059 total participants. The benchmark covers four domains: Movies, Books, TV Shows, and Songs.

- **Batch A (Movies & Books,** $N = 1,045$**)**: 51.39% North America, 29.28% Asia, 11.77% Europe, 7.27% South America. Gender: 50.2% Male, 46.9% Female, 1.1% Non-binary, 1.8% Prefer not to answer.
- **Batch B (TV Shows & Songs,** $N = 1,014$**)**: 49.51% North America, 21.89% Asia, 14.79% Europe, 13.81% South America. Gender: 51.8% Female, 45.8% Male, 1.3% Non-binary, 1.1% Prefer not to answer.

E.2 GISTBENCH PREPROCESSING

GISTBench (Fostiropoulos et al., 2026) is utilized as a privacy-preserving proxy for behavioral interaction data. We filter the dataset to include users with at least 10 positive interactions. Implicit positive signals are defined as "completed views" (watch time $\geq$ 90% of video duration), while "skips" (watch time $<$ 2s) are treated as implicit negatives.

E.3 INTERNAL PRODUCTION DATASET DETAILS

The production dataset consists of **6.5M users** and **22.8M posts** sampled from a contiguous window in 2026.

- **Anonymization**: All user and post identifiers are hashed using a salted SHA-256 algorithm. No PII (Personally Identifiable Information) is processed by the STEER LLM components.
- **Content Enrichment**: Posts are represented by their existing platform-level topical tags and high-level semantic embeddings. The "Think" stage receives a text-based summary of these tags (e.g., "Cooking, Italian Cuisine, Pasta Recipes") rather than raw pixels or text to maintain efficiency.
- **Evaluation Split**: We use a temporal split where the first 14 days of history are used to generate the STEER Memory, and the final 3 days are used to extract "Demonstrated Interests" for the retrieval evaluation.
- **Metric Validation.** To ensure a highly reliable ground-truth signal for whether engaged content aligns with a user's stated intent, we explicitly validated our alignment metric against human judgment using a robust two-step verification process:
  - *Human Evaluation Dataset.* We collected an evaluation dataset consisting of 10K randomly sampled ($\langle$intent, content$\rangle$) pairs explicitly labeled by human annotators.
  - *Quantitative Model Performance.* We deployed a specialized topicality model—a fine-tuned 2B parameter LLM—which achieves a strong AUC-ROC of 0.85 evaluated against this human-annotated ground truth.
  - *Evaluation Independence.* This 2B LLM serves as an independent semantic evaluator separate from STEER's internal components and does not share the same semantic descriptors used by STEER during generation or retrieval. This structurally guarantees an independent, non-circular evaluation.
- **Mitigation of Exposure Bias.** Exposure bias is explicitly handled through our data collection infrastructure to ensure the topicality model evaluates true semantic alignment rather than merely learning platform artifacts. Our training and evaluation datasets are constructed via stratified sampling across all intents and topics. This balancing strategy prevents rare or niche categories from being drowned out by highly exposed or viral items. By smoothing the distribution across the entire space, the model learns a generalized semantic alignment mapping instead of biasing toward historically highly exposed items or intents in the production system.

F FULL EXPERIMENTAL RESULTS WITH STATISTICAL SIGNIFICANCE

Table 5 presents the complete evaluation results across all three datasets, including bootstrap standard deviations ($\pm$) for all metrics. These results correspond to the summary provided in Table 3 of the main text.

Table 5: Complete Recommendation Quality with Bootstrap Standard Deviations. Best results are in **bold**, and second-best are underlined.

Tier	Method	Open-Steer HR@10	Open-Steer NDCG@10	Open-Steer P@10	Open-Steer R@10	GISTBench HR@10	GISTBench NDCG@10	GISTBench P@10	GISTBench R@10	Internal HR@10	Internal NDCG@10	Internal P@10	Internal R@10
Tier 1: History-only	SASRec	$17.729_{\pm 1.365}$	$4.705_{\pm 0.471}$	$3.409_{\pm 0.338}$	$5.158_{\pm 0.494}$	$0.942_{\pm 0.046}$	$0.129_{\pm 0.007}$	$0.098_{\pm 0.005}$	$0.126_{\pm 0.008}$	$0.024_{\pm 0.007}$	$0.014_{\pm 0.005}$	$0.002_{\pm 0.001}$	$0.022_{\pm 0.007}$
	HSTU	$17.362_{\pm 1.290}$	$6.744_{\pm 0.674}$	$4.575_{\pm 0.497}$	$7.092_{\pm 0.668}$	$0.747_{\pm 0.048}$	$0.100_{\pm 0.007}$	$0.077_{\pm 0.005}$	$0.104_{\pm 0.007}$	$0.024_{\pm 0.007}$	$0.010_{\pm 0.003}$	$0.002_{\pm 0.001}$	$0.018_{\pm 0.006}$
	STEER (w/o p)	$\underline{27.510}_{\pm 1.503}$	$\underline{7.560}_{\pm 0.533}$	$\underline{5.120}_{\pm 0.360}$	$\underline{7.710}_{\pm 0.518}$	$0.120_{\pm 0.013}$	$0.020_{\pm 0.003}$	$0.010_{\pm 0.003}$	$0.020_{\pm 0.003}$	$2.332_{\pm 0.068}$	$0.806_{\pm 0.041}$	$0.399_{\pm 0.029}$	$0.979_{\pm 0.044}$
Tier 2: Prompt-only	STEER (p only)	$23.670_{\pm 1.441}$	$6.140_{\pm 0.528}$	$3.980_{\pm 0.316}$	$6.470_{\pm 0.538}$	$6.940_{\pm 0.102}$	$1.620_{\pm 0.031}$	$0.880_{\pm 0.015}$	$1.950_{\pm 0.036}$	$18.858_{\pm 0.177}$	$9.584_{\pm 0.133}$	$2.221_{\pm 0.067}$	$14.570_{\pm 0.159}$
Tier 3: Full	STEER (no think)	$26.440_{\pm 1.498}$	$6.740_{\pm 0.532}$	$4.350_{\pm 0.426}$	$7.340_{\pm 0.556}$	$\underline{8.830}_{\pm 0.111}$	$\underline{1.990}_{\pm 0.032}$	$\underline{1.110}_{\pm 0.017}$	$\underline{2.410}_{\pm 0.038}$	$\underline{27.600}_{\pm 0.185}$	$\underline{14.980}_{\pm 0.162}$	$\underline{3.320}_{\pm 0.088}$	$\underline{21.720}_{\pm 0.181}$
	STEER (full)	$\mathbf{36.930}_{\pm 1.656}$	$\mathbf{11.430}_{\pm 0.717}$	$\mathbf{7.550}_{\pm 0.474}$	$\mathbf{11.610}_{\pm 0.694}$	$\mathbf{9.580}_{\pm 0.117}$	$\mathbf{2.170}_{\pm 0.033}$	$\mathbf{1.220}_{\pm 0.018}$	$\mathbf{2.630}_{\pm 0.038}$	$\mathbf{56.124}_{\pm 0.224}$	$\mathbf{35.532}_{\pm 0.216}$	$\mathbf{7.190}_{\pm 0.116}$	$\mathbf{43.710}_{\pm 0.225}$

G ONLINE A/B STUDY: PROTOCOL AND EXTENDED RESULTS

This appendix documents the full protocol of the production A/B study summarized in the main text, including how users were exposed to STEER, the construction of the control condition, the complete metric hierarchy, the survey instrument used to measure agency, and an analysis of novelty decay over the study period.

G.1 USER EXPOSURE

Users are exposed to STEER through dedicated UI entry points: an explicit *Steer Your Feed* module, and inline transparency cards that display the memories currently active in shaping the user's recommendations. These surfaces allow users to view and edit their memory profiles directly. Through these interfaces, millions of active users have interacted with, viewed, and edited their explicit memory profiles.

G.2 CONTROL CONDITION AND DURATION

The control group is a long-term 5% holdout of users who have access to neither the STEER UI nor its underlying steering capabilities. The holdout has run continuously for over three months, which substantially exceeds standard novelty-decay windows, typically 2–3 weeks.

G.3 GUARDRAIL TRACKING

We monitor a multi-tiered hierarchy of metrics, tracking first-order effects (memory views, edits), immediate steering response (engagement with post-steer recommendations), and strict system guardrails (overall app-level visitation patterns, session lengths, and technical latency) to ensure no ecosystem degradation.

G.4 QUANTIFYING AGENCY BEYOND ENGAGEMENT

Engagement metrics alone cannot validate a claim about agency, so we measured perceived control directly via continuous user satisfaction surveys across tens of thousands of users. Users exposed to STEER reported a $\sim 4.2\%$ statistically significant lift in perceived sense of control over the recommendation system relative to the holdout. A subsequent linear regression identified *memory accuracy* and *real-time responsiveness to steering* as the top two factors driving this increase, directly supporting the paper's framing.

G.5 ENGAGEMENT RESULTS AND NOVELTY DECAY

Against the holdout, treatment users showed a $+0.40\%$ ($\pm 0.37\%$) increase in app-level time spent and a $+0.21\%$ ($\pm 0.19\%$) increase in app visits. Given the platform's user base, improvements of this magnitude correspond to changed behavior for millions of users; typical launches deliver 20–30% of the impact measured for the STEER launch.

We did observe an initial novelty spike during the first two weeks of deployment, which yielded higher peak metrics. As user behavior normalized, the metrics stabilized, and a major, statistically significant fraction of the initial lift remained sustained throughout the long-term holdout. The figures reported above are measured in this post-stabilization regime and therefore reflect durable utility rather than transient novelty.

No monitored guardrail regressed over the test window, and technical latency remained within production service budgets, confirming that the *Think-Then-Recommend* paradigm is deployable under real-time constraints.

H HANDLING AMBIGUOUS OR CONFLICTING STEERING PROMPTS

Steering prompts are free-form natural language, so they can be internally contradictory, inconsistent with earlier instructions, or simply underspecified. STEER resolves these three cases hierarchically, distinguishing conflicts that must be surfaced to the user from those the system can resolve on its own.

Simultaneous/Intra-Prompt Conflicts. When a single prompt contains an explicit internal contradiction (e.g., *"show me more sports but fewer athletic events"*), the pre-processing layer is fine-tuned to detect the logical impasse. Rather than propagating an incoherent intent downstream, the system surfaces an interactive disambiguation prompt in the UI asking the user to resolve the preference. Deferring to the user here is deliberate: the conflict is genuine, and any automatic resolution would silently discard half of the stated intent.

Sequential/Inter-Prompt Conflicts. When instructions conflict across time (e.g., *"show me more soccer"* followed days later by *"show me less soccer"*), memory is treated as a dynamic state rather than an accumulating log: the most recent explicit directive overrides older directional intents on the same topic. This reflects the assumption that a later instruction represents an active shift in preference rather than a contradiction to be arbitrated.

Semantic ambiguity. For inherently vague prompts (e.g., "show me interesting things"), the system leverages the *Think* stage, taking into account the user's historical memory, to properly contextualize the prompt for personalized interpretations.