WhisperRec: Latent Reasoning for Efficient Foundation Recommendation Models

Hao Jiang[1] *, Peiru Du[1] *, Pengfei Yao[1] *, Mengting Li[1†], Siyuan Lou[1], Kuo Cai[1], Sheng Yu[1], Qiang Luo[1‡], Jian Liang[1], Ruiming Tang[1‡], Fei Pan[1], Peng Jiang[1], Wenwu Ou[1]

[1]Kuaishou Technology, Beijing, China

{jianghao11, dupeiru, yaopengfei03, limengting03, lousiyuan05, caikuo, yusheng03, luoqiang, liangjian03, tangruiming, panfei05, jiangpeng, luocheng10 }@kuaishou.com

Abstract

Large language models (LLMs) have demonstrated strong reasoning capabilities, motivating their adoption as the backbone of foundation recommendation models (FRMs). Existing approaches typically improve recommendation through explicit Chain-of-Thought (CoT) reasoning under the *Think-then-Answer* paradigm. However, explicit CoT incurs substantial inference overhead due to lengthy reasoning generation and relies on manually designed templates that struggle to capture diverse and dynamic user interests in real-world recommendation. To address these limitations, we propose WhisperRec, an efficient latent reasoning framework for FRMs. The key idea is to compress explicit reasoning traces into several learnable latent reasoning tokens, enabling efficient *Latent Reason-then-Answer* inference without generating verbose reasoning traces. Specifically, WhisperRec first introduces Multi-View Adaptive CoT (MV-ACoT), which jointly constructs diverse, high-quality CoT supervision by exploring user interests from multiple perspectives and automatically adapts reasoning complexity to each instance. This strategy enables the teacher model to apply lightweight reasoning to simple cases and more complex reasoning to challenging cases. Building on a pre-trained FRMs, WhisperRec then employs a Three-stage Latent Reasoning Alignment paradigm to progressively internalize teacher-generated CoT into learnable latent token. Finally, a multi-stage curriculum-based post-training strategy effectively activates latent-token reasoning for downstream recommendation. Extensive experiments on an industrial-scale Kuaishou dataset and the public Kuaishou LLM-Rec benchmark show that WhisperRec consistently outperforms explicit CoT-based methods and traditional baselines. Compared with the explicit CoT *Think* and *No-Think* variants, WhisperRec improves SID@64 by 17.44% and 9.33%, respectively, while achieving more than 10× higher online inference throughput[1].

Introduction

Driven by the scaling of model capacity and data, LLMs have progressed from generation to knowledge-intensive reasoning and agentic decision making (Jaech et al. 2024; Guo et al. 2025). This progress has motivated researchers to transfer and inherit these capabilities to recommendation systems (Zhai et al. 2024; Zhang et al. 2024; Rajput et al. 2018). Recent generative recommendation models, exemplified by the OneRec family (Zhou et al. 2025c,d; Liu et al. 2025a; Zhou et al. 2025b), have demonstrated that Transformer-based architectures can scale recommendation modeling to support industrial-scale end-to-end deployment. However, they mostly make predictions through implicit representation computing, making their decision processes difficult to interpret or control (Zhou et al. 2017, 2019). Although several studies introduce latent reasoning mechanisms into recommendation (Tang et al. 2026; Dai et al. 2025), their reasoning still operates in a non-semantic latent space, making it difficult to determine whether the learned representations encode coherent, task-relevant reasoning.

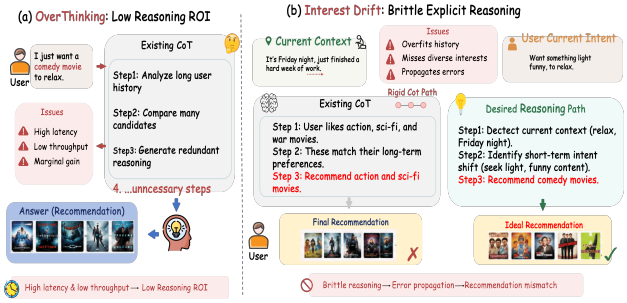

Figure 1: Limitations of explicit CoT for recommendation: overthinking of low reasoning ROI and interest drift.

A direct approach is to use LLMs as the backbone of foundation recommendation models (FRMs) and adapt them through recommendation-specific training (Yi et al. 2025b,a; Zhou et al. 2025b). Recent work, particularly OneReason (Team et al. 2026), demonstrates the sucess of the *Think-then-Answer* paradigm for LLM-based FRMs, in which the model reasons over user preferences and context before generating recommendations. However, recommendation reasoning cannot simply inherit generic CoT patterns (Zhou et al. 2025a; Zhang et al. 2026), as user interests are diverse and evolving, requiring structured reasoning from preference induction to intent inference and decision making. Despite its promise, explicit CoT-based FRMs face two key limitations.

(1) Overthinking and low reasoning ROI. As shown in Figure 1(a), explicit CoT often generates lengthy and redundant reasoning even for simple requests with clear intents. This increases inference latency and reduces throughput, while bringing only marginal recommendation gains.

(2) Interest drift and brittle explicit reasoning. As

[1]We will release our code upon publication.

*Equal contributions

†Work done during an internship at Kuaishou Technology

‡Corresponding authors

shown in Figure 1(b), fixed and single-path of CoT can overemphasize dominant historical preferences while overlooking short-term, context-dependent interests. Once even a minor reasoning step goes wrong, the error may propagate to the final recommendation and cause a mismatch.

To address these limitations, we propose WhisperRec, a latent reasoning framework for FRMs. Rather than generating explicit CoT for each recommendation decision, WhisperRec internalizes recommendation reasoning capabilities into a set of learnable latent tokens, enabling a *Latent-Reason-then-Answer* paradigm. This design preserves high-dimensional latent reasoning capacity while avoiding the latency and throughput bottleneck of autoregressive rationale generation. Specifically, WhisperRec first introduces Multi-View Adaptive CoT (MV-ACoT) to construct teacher-generated CoT supervision. MV-ACoT jointly explores user interests from multiple perspectives and automatically adapts reasoning complexity to individual instances, applying lightweight reasoning to simple cases and targeted reasoning to challenging ones. Building on a pretrained FRM, WhisperRec then employs a three-stage Latent Reasoning Alignment paradigm to progressively distill teacher-generated CoT into latent representations. Finally, a multi-stage curriculum-based post-training strategy activates latent-token reasoning for downstream recommendation, enabling efficient and robust recommendations under dynamic and complex user interest dynamics.

We conduct extensive experiments on an industrial-scale Kuaishou recommendation dataset and the public Kuaishou LLM-Rec benchmark. WhisperRec consistently outperforms competitive Foundation Recommendation Models and strong explicit CoT-based OneReason. In particular, it improves SID@64 by 17.44% over the explicit Think variant and by 9.33% over the No-Think variant, while delivering approximately **10×** higher online inference throughput than explicit reasoning. Further analyses show that MV-ACoT improves teacher CoT quality, leading to significant gains in recommendation performance. Latent Reasoning Alignment transfers explicit CoT knowledge into latent tokens, while multi-stage curriculum-based post-training further activates latent reasoning. We further observe a strong positive correlation between CoT quality and downstream recommendation performance, suggesting that reasoning benefits recommendation by improving decision-relevant representations rather than by generating lengthy natural-language rationales.

Our main contributions are summarized as follows:

- We propose WhisperRec, an efficient latent reasoning framework that introduces the *Latent Reason-then-Answer* paradigm for FRMs and improves performance.
- We develop MV-ACoT, a Latent Reasoning Alignment framework, and curriculum-based post-training to internalize explicit CoT into latent tokens for FRMs.
- Extensive experiments on industrial and public benchmarks show that WhisperRec achieves superior recommendation performance and approximately $10\times$ higher inference throughput than explicit CoT methods.

Related Work

Scaling Recommendation with LLMs

Traditional recommendation methods learn user representations from historical interactions and predict the next item (Hidasi et al. 2015; Kang and McAuley 2018; Sun et al. 2019). With the rise of LLMs, early LLM-based approaches directly prompted LLMs to make recommendation decisions (Geng et al. 2022; Gao et al. 2023; Hou et al. 2024). Although these methods leverage the world knowledge of LLMs, they remain limited in modeling collaborative signals. Recent studies have advanced recommendation systems toward foundation recommendation models along two complementary directions. The first develops recommendation-native backbones: TIGER represents items using semantic IDs (Rajput et al. 2023), HSTU formulates large-scale recommendation as sequence transduction (Zhai et al. 2024), and OneRec unifies retrieval and ranking in an end-to-end generative architecture (Deng et al. 2025). The second adapts pretrained LLMs to recommendation data. LC-Rec aligns linguistic knowledge with collaborative semantics (Zheng et al. 2024), while the RecGPT family develops transferable LLM-based recommenders through recommendation-domain pretraining (Yi et al. 2025b,a). Despite their success, these models mainly focus on behavior modeling and item generation, providing limited supervision for inferring latent user needs and making recommendation decisions.

Reasoning-Enhanced Recommendation

Recent work has begun to equip recommendation models with reasoning capabilities. Latent reasoning methods reduce inference costs by avoiding lengthy natural-language rationales. ReaRec recursively refines user representations in the hidden space (Tang et al. 2026), OnePiece introduces reasoning modules into industrial ranking models (Dai et al. 2025), OneSearch-V2 distills LLM reasoning into compact internal states (Chen et al. 2026), and REG4Rec explores multiple latent reasoning paths through self-reflection (Xing et al. 2025). In parallel, explicit reasoning methods expose intermediate reasoning processes in natural language. GOT4Rec (Long et al. 2024), Reason4Rec (Fang et al. 2025), and ThinkRec (Yu et al. 2026) construct user preference analyses or multi-path reasoning trajectories through supervised fine-tuning. RecOne (Kong et al. 2026), OneRec-Think (Liu et al. 2025b), and OpenOneRec (Zhou et al. 2025b) further combine CoT supervision with reinforcement learning to improve recommendation decisions. OneReason (Team et al. 2026) shows that simply introducing explicit CoT does not consistently activate reasoning in FRMs, and strengthens the reasoning process through item semantic awareness, user behavior cognition, and reinforcement learning. Although these methods advance reasoning-based recommendation, explicit reasoning is costly and prone to noisy rationales, whereas latent reasoning often lacks direct supervision for recommendation decisions. Our work bridges both paradigms by internalizing diverse, domain-specific reasoning trajectories into latent tokens through pre-training and post-training.

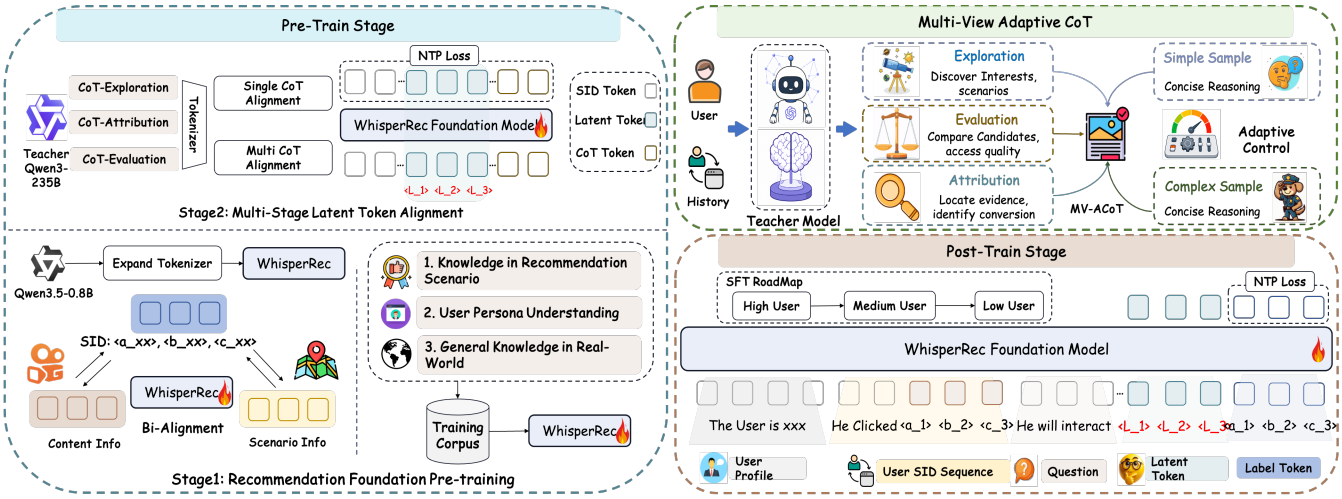

Figure 2: Overview of the WhisperRec, containing Multi-view Adaptive CoT, Latent Token Alignment and Post-training stage.

Methodology

Problem Formulation and Framework

For a user u, let P_u denote the user profile, H_u the chronologically ordered interaction history, $\mathcal{C}_u$ the optional spatiotemporal context, and i_u^+ the next interacted item. Each item i is represented by a Semantic ID (SID) sequence $\mathcal{S}_i = [s_i^{(1)}; s_i^{(2)}; s_i^{(3)}]$. The recommendation objective is to generate $\mathcal{S}_{i_u^+}$ from information available before the target interaction. Neither i_u^+ nor its SID is included in the input.

As shown in Figure 2, WhisperRec has three components. First, Multi-View Adaptive Chain-of-Thought (MV-ACoT) constructs complementary supervision for intent exploration, candidate evaluation, and conversion attribution. Second, a three-stage alignment procedure uses these rationales to train a small set of contextualized latent tokens. Third, the post-training jointly optimizes standard and latent token recommendation modes. During online inference, the model consumes the user context followed by the latent tokens and directly generates the target SID. It therefore avoids autoregressive generation of natural-language rationales.

Multi-View Adaptive CoT

We propose Multi-View Adaptive CoT to provide high-quality, recommendation-oriented CoT supervision, enabling the FRMs to acquire latent reasoning capabilities.

Multi-View CoT Task Design We argue that effective recommendation CoT should capture decision-relevant signals without unnecessary complexity. However, under multi-interest, context-dependent, and noisy user behaviors, a single reasoning task may lead to uncontrolled divergence, inaccurate candidate assessment, or hindsight rationalization. MV-ACoT addresses this issue by constructing complementary **Exploration**, **Evaluation**, and **Attribution** traces, while adapting reasoning complexity to sample-level decision difficulty to avoid overthinking on easy cases and insufficient analysis on hard ones.

Given a user profile U, historical behavior sequence H, candidate ad item i_c, difficulty of instance d, ad item i_c and target item i^+, the teacher model P_θ generates reasoning traces T and auxiliary judgment signals under different CoT prompt functions.

View 1: Exploration. The exploration view receives no candidate or target item. Given only the profile and history, it identifies plausible commercial intents:

$$(R_u^{\text{exp}}, \mathcal{I}_u) \sim p_{\text{T}}(\cdot \mid \pi_{\text{exp}}(P_u, H_u; d_u)), \tag{1}$$

where R_u^{exp} is the rationale, $\mathcal{I}_u$ is a set of potential intents. This strategy encourages model to infer diverse user needs, interest shifts, and exploratory recommendation directions, thereby expanding the latent intent space.

View 2: Evaluation. Given a candidate i_c, the evaluation view assesses whether it is supported by the user's interests:

$$(R_{u,c}^{\text{eval}}, y_{u,c}, q_{u,c}) \sim p_{\text{T}}(\cdot \mid \pi_{\text{eval}}(P_u, H_u, \mathcal{C}_u, i_c; d_{u,c})), \tag{2}$$

where $y_{u,c}$ is a categorical matching judgment and $q_{u,c} \in [0, 1]$ is the teacher confidence. This strategy simulates user-item pair evaluation in recommendation ranking and allows negative judgments when the evidence is insufficient.

View 3: Attribution. For a positive interaction i_u^+, the attribution view examines which preceding evidence supports the observed outcome:

$$(R_u^{\text{attr}}, a_u, q_u^{\text{attr}}) \sim p_{\text{T}}\big(\cdot \mid \pi_{\text{attr}}(P_u, H_u, \mathcal{C}_u, i_u^+; d_u)\big), \tag{3}$$

where a_u denotes the attribution category and $q_u^{\text{attr}} \in [0, 1]$ is the attribution confidence. This strategy anchors reasoning on the observed conversion and captures evidence chains behind realized commercial intent.

The three strategies provide complementary supervision. Exploration supports open-ended intent discovery, evaluation enables candidate discrimination without assuming correctness, and attribution grounds reasoning in conversion evidence while mitigating hindsight rationalization. Together, they provide generative, discriminative, and evidence-based CoT supervision, reducing the bias of a single reasoning task.

Adaptive CoT Generation. Beyond multi-view construction, MV-ACoT adapts reasoning complexity to sample-level

decision difficulty. Unlike fixed-template CoT, which applies the same reasoning process to all samples, MV-ACoT estimates the complexity of a user context from behavioral signal strength, weakly related behaviors, and noisy interactions:

$$d = g(U, H), \quad d \in \{\textsc{Low}, \textsc{High}\}, \tag{4}$$

where $g(\cdot)$ denotes the complexity assessment function.

Given the difficulty d, the teacher model generates a reasoning trace conditioned on an adaptive prompt π_d:

$$T \sim P_\theta\left(T \mid U, H, i_c, i^+; \pi_d, \mathcal{C}\right), \tag{5}$$

where $\mathcal{C}$ denotes recommendation constraints, including behavioral strength, fulfillment mode, spatiotemporal context, and industry-specific factors. For high-uncertainty samples, π_d expands reasoning across multiple factors to distinguish competing intents and identify noisy signals. For low-uncertainty samples, it prunes unnecessary dimensions and focuses on dominant conversion signals. Thus, MV-ACoT allocates its reasoning budget according to decision difficulty rather than uniformly generating longer CoT. Appendix C summarizes the prompt construction principles.

Latent Token Alignment

Starting from the pre-trained FRMs $\mathcal{F}_{\text{base}}$, we conduct a three-stage Latent Reasoning Alignment procedure, containing single-path and multi-path latent reasoning alignment and recommendation-oriented context alignment. The first two stages distill teacher-generated CoT into latent tokens under teacher model supervision, while the final stage aligns the learned latent token with next-item prediction.

FRMs Pre-training We initialize from Qwen3.5-0.8B and extend its vocabulary with Semantic ID tokens. Each item v is represented by a three-level SID sequence with Res-Kmeans:

$$\mathcal{S}_v = [s_v^{(1)}; s_v^{(2)}; s_v^{(3)}], \tag{6}$$

where $s_v^{(l)}$ denotes the SID at level l, encoding item side multi-model information (Luo et al. 2025). We bidirectionally align $\mathcal{S}_v$ with the dense item's caption C_v:

$$\mathcal{L}_{\text{sem}} = -\log p_\theta(\mathcal{S}_v \mid C_v) - \log p_\theta(C_v \mid \mathcal{S}_v). \tag{7}$$

Moreover, we introduce diverse CPT tasks to enhance the FRMs, including recommendation-domain knowledge such as spatiotemporal context, user persona understanding, and general-domain knowledge. Together, these tasks construct a capable foundation model for recommendation.

Stage I: Single-View Latent Reasoning Alignment We first warm up the latent tokens using one high-confidence, target-free rationale for each training context. Let $v_u^\star$ denote the selected reasoning view and $\langle v_u^\star \rangle$ its corresponding control token. The training sequence is constructed as

$$[X_u; \mathcal{L}; \langle v_u^\star \rangle; \widetilde{R}_u^{(v_u^\star)}]. \tag{8}$$

We apply the training loss only to the rationale tokens:

$$\mathcal{L}_{\text{single}} = -\mathbb{E}_u \sum_{t=1}^{|\widetilde{R}_u^{(v_u^\star)}|} \log p_\phi\left(\widetilde{r}_{u,t}^{(v_u^\star)} \mid X_u, \mathcal{L}, \langle v_u^\star \rangle, \widetilde{r}_{u,<t}^{(v_u^\star)}\right). \tag{9}$$

Placing the view token after the shared latent tokens ensures that the latent slots are derived from a common recommendation context, rather than being conditioned on view-specific textual instructions.

Stage II: Multi-Views Latent Reasoning Alignment We then use every retained MV-ACoT rationale. Let $\mathcal{V}_u \subseteq \{\text{exp}, \text{eval}, \text{attr}\}$ be the available views for user u. The multi-view objective is Let $C_{u,t}^{(v)} = (X_u, \mathcal{L}, v, \widetilde{r}_{u,<t}^{(v)})$ denote the decoding context at step t. The multi-view objective is

$$\begin{aligned} \ell_u^{(v)} &= -\frac{1}{|\widetilde{R}_u^{(v)}|} \sum_{t=1}^{|\widetilde{R}_u^{(v)}|} \log p_\phi\left(\widetilde{r}_{u,t}^{(v)} \mid C_{u,t}^{(v)}\right), \\ \mathcal{L}_{\text{multi}} &= \mathbb{E}_u\left[\frac{1}{|\mathcal{V}_u|} \sum_{v \in \mathcal{V}_u} \ell_u^{(v)}\right]. \end{aligned} \tag{10}$$

Each rationale is paired with a view token indicating exploration, evaluation, or attribution. We average token losses within each rationale so that long traces do not receive greater weight. Training all three views with the same contextualized slots encourages a shared user representation that supports all three reasoning objectives.

Stage III: Recommendation-Oriented Context Alignment The preceding stages teach latent tokens to represent CoT knowledge, but do not directly optimize recommendation decisions. We therefore align latent token with target-item prediction. In addition to the user profile P_u and historical SID sequence H_u, we introduce recommendation contextual information $\mathcal{C}_u$ (e.g., temporal and geographical signals). The target SID sequence is excluded from the input:

$$X_u^{\text{ctx}} = [\mathcal{Z}; P_u; H_u; \mathcal{C}_u]. \tag{11}$$

The model predicts the observed target SID sequence $\mathcal{S}_u^+$:

$$\mathcal{L}_{\text{align}} = -\mathbb{E}\left[\log p_\theta\left(\mathcal{S}_u^+ \mid X_u^{\text{ctx}}\right)\right]. \tag{12}$$

This stage makes latent tokens a bridge between user context and recommendation decisions, yielding the context-aware latent reasoning FRM $\mathcal{F}_{\text{latent-ctx}}$.

Post-Training and Inference

We further adapt $\mathcal{F}_{\text{latent-ctx}}$ to the target recommendation scenario through multi-stage curriculum-based recommendation SFT and latent reasoning SFT.

Multi-stage Curriculum-based Recommendation SFT The primary objective of this stage is to equip WhisperRec with recommendation capabilities in the target scenario. We first train the model to predict the next target SID sequence from standard recommendation inputs:

$$X_u^{\text{std}} = [P_u; H_u]. \tag{13}$$

Here, P_u provides persistent user attributes, while H_u captures behavioral preferences. The training objective is

$$\mathcal{L}_{\text{SFT}} = -\mathbb{E}\left[\log p_\theta\left(\mathcal{S}_u^+ \mid X_u^{\text{std}}\right)\right]. \tag{14}$$

To smoothly equips a LLMs with recommendation capability across industrial scenarios with varying degrees of user activity sparsity, we introduce a **multi-stage curriculum learning strategy**. Specially, we employ a curriculum from high-activity to low-activity users:

$$\mathcal{D}_{\text{high}} \rightarrow \mathcal{D}_{\text{medium}} \rightarrow \mathcal{D}_{\text{low}}. \tag{15}$$

Backbone	Method	CoT	OneReason Public Benchmark								Industrial Dataset							
			SID@1	SID@16	SID@32	SID@64	ID@1	ID@16	ID@32	ID@64	SID@1	SID@16	SID@32	SID@64	ID@1	ID@16	ID@32	ID@64
ID-based	GRU4Rec	/	–	–	–	–	0.0119	0.0147	0.0157	0.0182	–	–	–	–	0.0158	0.1057	0.1454	0.1883
	BERT4Rec	/	–	–	–	–	0.0122	0.0153	0.0161	0.0189	–	–	–	–	0.0174	0.1074	0.1480	0.1932
	HSTU	/	–	–	–	–	0.0114	0.0146	0.0165	0.0195	–	–	–	–	0.0145	0.1039	0.1414	0.1812
SID-based	TIGER	/	0.0051	0.0061	0.0076	0.0078	0.0106	0.0124	0.0137	0.0156	0.0714	0.1000	0.1197	0.1432	0.0092	0.0401	0.0598	0.0859
	OneRec	/	0.0049	0.0063	0.0076	0.0082	0.0108	0.0135	0.0152	0.0177	0.0708	0.1031	0.1252	0.1468	0.0116	0.0447	0.0650	0.0900
LLM-based	OneReason	/	0.0055	0.0077	0.0079	0.0081	0.0234	0.0244	0.0258	0.0272	0.0762	0.1527	0.1603	0.1658	0.2254	0.4522	0.4750	0.4941
LLM-based	OneReason+CoT NoThink	OR	0.0065	0.0103	0.0104	0.0104	0.0262	0.0306	0.0314	0.0334	0.0809	0.1639	0.1739	0.1823	0.2293	0.4499	0.4727	0.4936
		Ada	0.0070	0.0113	0.0124	0.0130	0.0270	0.0341	0.0364	0.0387	0.0873	0.1746	0.1832	0.1913	0.2365	0.4620	0.4850	0.5049
		Exp.	0.0068	0.0112	0.0113	0.0113	0.0228	0.0286	0.0318	0.0374	0.0881	0.1728	0.1819	0.1894	0.2385	0.4621	0.4845	0.5032
		Eval.	0.0079	0.0139	0.0172	0.0213	0.0252	0.0357	0.0486	0.0623	0.0894	0.1800	0.1896	0.1962	0.2356	0.4630	0.4881	0.5065
		Attr.	0.0071	0.0125	0.0165	0.0208	0.0255	0.0379	0.0462	0.0601	0.0897	0.1793	0.1876	0.1943	0.2392	0.4640	0.4860	0.5040
		Merge	0.0082	0.0131	0.0178	0.0209	0.0258	0.0362	0.0471	0.0612	0.0861	0.1720	0.1790	0.1851	0.2385	0.4640	0.4846	0.5038
LLM-based	OneReason+CoT Think	OR	0.0063	0.0073	0.0080	0.0089	0.0235	0.0263	0.0278	0.0305	0.0601	0.1405	0.1563	0.1697	0.1838	0.4080	0.4450	0.4759
		Ada	0.0064	0.0074	0.0086	0.0093	0.0233	0.0267	0.0296	0.0311	0.0714	0.1486	0.1590	0.1688	0.2077	0.4329	0.4606	0.4844
		Exp.	0.0065	0.0082	0.0084	0.0087	0.0235	0.0285	0.0291	0.0299	0.0672	0.1190	0.1391	0.1465	0.2013	0.3992	0.4301	0.4512
		Eval.	0.0065	0.0081	0.0087	0.0096	0.0234	0.0276	0.0299	0.0328	0.0696	0.1169	0.1332	0.1487	0.2083	0.3575	0.3952	0.4303
		Attr.	0.0066	0.0078	0.0081	0.0084	0.0235	0.0273	0.0282	0.0296	0.0682	0.1287	0.1452	0.1617	0.2029	0.3739	0.4088	0.4458
		Merge	0.0072	0.0084	0.0086	0.0091	0.0241	0.0281	0.0288	0.0303	0.0728	0.1301	0.1400	0.1505	0.2127	0.3985	0.4275	0.4567
LLM-based	WhisperRec	OR	0.0065	0.0082	0.0089	0.0092	0.0156	0.0253	0.0268	0.0282	0.0882	0.1771	0.1867	0.1939	0.2354	0.4632	0.4856	0.5061
		Ada	0.0075	0.0126	0.0138	0.0144	0.0224	0.0378	0.0403	0.0429	0.0898	0.1770	0.1869	0.1935	0.2376	0.4631	0.4867	0.5052
		Exp.	0.0072	0.0125	0.0137	0.0144	0.0179	0.0350	0.0373	0.0402	0.0930	0.1811	0.1908	0.1971	0.2425	0.4650	0.4883	0.5060
		Eval.	0.0087	0.0155	0.0189	0.0233	0.0300	0.0410	0.0560	0.0712	0.0946	0.1831	0.1931	0.1989	0.2436	0.4660	0.4891	0.5070
		Attr.	0.0072	0.0145	0.0150	0.0167	0.0299	0.0365	0.0478	0.0560	0.0902	0.1779	0.1869	0.1936	0.2396	0.4645	0.4879	0.5055
		Merge	**0.0089**	**0.0158**	**0.0193**	**0.0239**	**0.0331**	**0.0506**	**0.0584**	**0.0742**	**0.0949**	**0.1847**	**0.1957**	**0.1993**	**0.2440**	**0.4662**	**0.4895**	**0.5088**
		vs. SOTA (Δ%)	**+41%↑**	**+116%↑**	**+141%↑**	**+169%↑**	**+41%↑**	**+92%↑**	**+110%↑**	**+143%↑**	**+58%↑**	**+31%↑**	**+25%↑**	**+17%↑**	**+33%↑**	**+14%↑**	**+10%↑**	**+7%↑**

Table 1: Results on public and industrial datasets. ID-based baselines do not generate semantic IDs, and their SID results are marked as "–". OR/Ada/Exp./Eval./Attr. denote OneReason-CoT, Adaptive CoT, and MV-ACoT views.

Latent Reasoning-based Recommendation SFT We then activate latent reasoning by prepending the learned latent tokens to the same user context:

$$X_u^{\mathrm{LR}} = [\mathcal{Z}; P_u; H_u]. \tag{16}$$

Unlike context alignment, this stage omits auxiliary context signals to ensure that latent reasoning remains effective under standard online recommendation inputs. The model is optimized to predict the same target SID sequence:

$$\mathcal{L}_{\mathrm{LR}} = -\mathbb{E}\left[\log p_\theta\left(\mathcal{S}_u^+ \mid X_u^{\mathrm{LR}}\right)\right]. \tag{17}$$

To preserve performance in both latent-reasoning and standard recommendation modes, we mix latent reasoning and conventional SFT samples at a 1:1 ratio:

$$\mathcal{L}_{\mathrm{post}} = \frac{1}{2}\mathcal{L}_{\mathrm{LR}} + \frac{1}{2}\mathcal{L}_{\mathrm{SFT}}. \tag{18}$$

This objective enables the *Latent-Reason-then-Answer* paradigm when $\mathcal{Z}$ is provided, while retaining stable recommendation performance without latent tokens.

Inference Online inference uses X_u^{LR} and decodes only the three-level target SID:

$$\widehat{\mathcal{S}}_u = \arg\max_{\mathcal{S}} p_\phi(\mathcal{S} \mid P_u, H_u, \mathcal{L}). \tag{19}$$

The MV-ACoT teacher, view tokens, privileged item information, and textual rationale decoder are absent at inference. The reasoning overhead is bounded by the fixed number K of latent tokens rather than by the length of a generated natural-language CoT traces.

Experiment

We organize the experiments around four questions:

- **RQ1:** Does WhisperRec outperform ID-based and SID-based baselines, as well as state-of-the-art FRMs?
- **RQ2:** Does MV-ACoT provide higher-quality and complementary CoT supervision for improving FRMs?
- **RQ3:** Can latent reasoning further improve performance over explicit CoT reasoning?
- **RQ4:** Does latent reasoning preserve general ability while improving reasoning efficiency?

Dataset	SFT Train	CoT Train	Test
Kuaishou Locallife	100,063	59,113	20,393
Kuaishou LLM-Rec	60,000	30,000	10,000

Table 2: Statistics of the datasets used in our experiments.

Experiment Setting

Datasets and Models. We evaluate WhisperRec on two datasets. The public dataset is derived from the advertising domain of the Kuaishou LLM-Rec benchmark [2]. The industrial dataset is collected from Kuaishou local-service scenarios and contains seven days of real-world online user interaction data. The basic statistics of the two datasets are shown in Table 2. We use Qwen3.5-0.8B (Qwen Team 2026) as the FRM's backbone and extend its vocabulary with SID tokens and latent tokens. For the public benchmark, we initialize WhisperRec from the OneReason-0.8B-pretrain-competition checkpoint [3] (Team 2026), followed by latent reasoning pretraining and post-training.

Baselines We compare WhisperRec with three groups of baselines: (1) ID-based sequential recommenders, including GRU4Rec (Hidasi et al. 2015), BERT4Rec (Sun et al. 2019), and HSTU (Zhai et al. 2024); (2) SID-based generative recommenders, including TIGER (Rajput et al. 2023) and

[2] https://huggingface.co/datasets/OpenOneRec/Explorer_LLM_Rec_Competition

[3] https://huggingface.co/OpenOneRec/OneReason-0.8B-pretrain-competition

Dimension	OR	Ada	Expl.	Eval.	Attr.
Factuality	3.48	3.70	3.96	**4.04**	3.62
Evidence	3.17	3.68	4.00	**4.34**	4.13
Signal	3.19	3.35	3.74	**4.04**	3.52
Intent	3.76	3.86	4.52	4.59	**4.68**
Causal	2.74	3.27	4.02	**4.25**	3.67
Ad–User Fit	3.26	3.85	4.49	**4.58**	4.30
Spatiotemporal	3.34	3.87	4.32	4.44	**4.64**
Confidence	2.78	2.91	3.91	**4.11**	3.64
Task Fulfillment	2.56	3.36	4.48	**4.59**	4.08
Average	3.14	3.54	4.16	**4.35**	4.03

Table 3: LLM-as-Judge results of CoT strategies.

OneRec (Zhou et al. 2025c); and (3) OneReason-based variants, including OneReason and its CoT *No-Think* and *Think* versions, serve as state-of-the-art FRM baselines. For CoT analysis, we compare MV-ACoT with standard OneReason-CoT and Adaptive CoT.

Metrics and Implement For recommendation performance, we report SID Hit@K and ID Hit@K at $K = 16, 64$. For SID-based models, we map generated SIDs back to item IDs through SID decoding. For latent reasoning analysis, we report codebook retrieval Hit@1/2/3, inference QPS, semantic similarity, and MMLU scores. We implement our model using the Hugging Face `Trainer` with Fully Sharded Data Parallel (FSDP) for distributed training. During evaluation, we use vLLM with beam search for efficient decoding. More implementation details are provided in Appendix A.

Overall Results (RQ1)

Table 1 reports the overall recommendation performance on both the public benchmark and the industrial dataset. WhisperRec denotes our latent reasoning model trained with MV-ACoT supervision. Experimental results show that WhisperRec achieves the best performance across all metrics, outperforming traditional baselines and the state-of-the-art FRM OneReason. Compared with the explicit CoT Think and No-Think variants of OneReason, WhisperRec improves SID@64 by 17.44% and 9.33%, respectively. Its gains over OneReason and its explicit-CoT variants suggest that latent token reasoning is more effective than relying directly on verbose CoT supervision. Latent tokens may capture higher-level semantic reasoning patterns that better connect user behavior evidence, contextual intent, and target generation.

CoT Analysis (RQ2)

We analyze our MV-ACoT from two aspects, the quality of teacher-generated CoT traces and their downstream effect on recommendation performance.

LLM-as-a-Judge Evaluation. We evaluate generated reasoning traces using an LLM-as-a-Judge protocol. For each CoT construction view, we sample 100 instances and score them with a unified rubric covering nine dimensions: **factuality**, evidence selection, **signal strength**, **intent translation**, **causal logic**, **ad-user fit**, **spatiotemporal reasoning**, **confidence**, and **task fulfillment**. We use GPT-5.5 as the judge; the complete prompt and rubric are provided in Appendix D.

CoT Construction	Public Benchmark		Industrial Dataset	
	SID@64	ID@64	SID@64	ID@64
OneReason-CoT	0.0092	0.0282	0.1939	0.5061
Adaptive CoT	0.0144	0.0429	0.1935	0.5052
Exploration	0.0144	0.0402	0.1971	0.5060
Evaluation	0.0233	0.0712	0.1989	0.5070
Attribution	0.0167	0.0560	0.1936	0.5055
MV-ACoT–Merge	**0.0239**	**0.0742**	**0.1993**	**0.5088**

Table 4: Ablation of CoT construction in WhisperRec.

As shown in Table 3. The results support the value of multi-view CoT construction. Although Exploration does not always achieve the highest overall score, it produces diverse user intent hypotheses that target-conditioned tasks may overlook. Evaluation provides stable candidate-level analysis by assessing whether an item is supported by current user evidence. Attribution reveals association chains behind observed conversions, particularly improving intent translation and spatiotemporal reasoning. Together, these views expand the intent space, improve evidence-grounded discrimination, and strengthen conversion-oriented reasoning, yielding richer supervision than a single CoT task.

Ablation Study on CoT Effectiveness. We further examine whether MV-ACoT's complementary supervision translates into downstream gains for WhisperRec. As shown in Table 4, replacing OneReason-CoT with Adaptive CoT already yields clear improvements, indicating the importance of matching reasoning complexity to recommendation difficulty. By avoiding unnecessary long-context reasoning for simple samples and focusing analysis on ambiguous cases, Adaptive CoT provides cleaner supervision and reduces the risk of hallucinated rationales.

Single-view MV-ACoT variants further improve performance through task-specific reasoning objectives. Evaluation performs strongly because candidate-level discrimination directly aligns with user–item decision making. Exploration broadens potential intent hypotheses, whereas Attribution captures conversion-oriented evidence chains. These results show that each view provides distinct reasoning signals beyond generic or adaptive CoT. The integrated MV-ACoT-Merge setting further demonstrates the benefit of multi-view supervision. By combining intent exploration, evidence-grounded evaluation, and conversion attribution, it offers more comprehensive supervision than any single view. Figure 3 also shows that WhisperRec consistently outperforms OneReason+CoT Think under the same CoT construction strategies, suggesting that latent reasoning more effectively translates view-specific supervision into SID retrieval gains. Improvements across Adaptive CoT and MV-ACoT variants further indicate that complexity adaptation and multi-view objectives complement latent encoding.

Latent Reasoning Analysis (RQ3)

In this section, we examine whether latent reasoning can further improve over explicit CoT reasoning.

Recommendation Performance. We first compare UnCoT, CoT, and WhisperRec on the recommendation task.

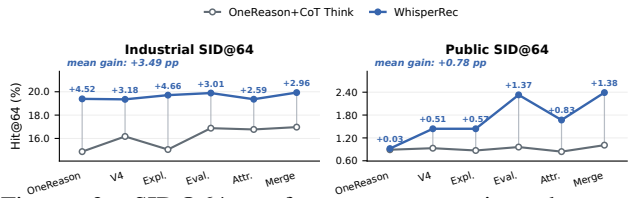

Figure 3: SID@64 performance comparison between OneReason Think and WhisperRec under different CoT.

As shown in Table 5, CoT yields limited and inconsistent gains over UnCoT, whereas latent reasoning substantially improves performance across all SID levels. In particular, Latent Reason improves Hit@L1 from 0.354 to 0.401 over explicit CoT and achieves the best results on Hit@L2 and Hit@L3. These results suggest that latent tokens more effectively encode reasoning information for recommendation.

Method	Hit@L1	Hit@L2	Hit@L3
OneReason-UnCoT	0.350	0.122	0.045
OneReason-CoT	0.354	0.124	0.043
WhisperRec	**0.401**	**0.143**	**0.071**

Table 5: Effectiveness of latent reasoning on the three-level codebook retrieval task.

Inference Efficiency. We compare the inference efficiency of UnCoT, CoT, and WhisperRec on 872 test instances. As shown in Table 6, Plain CoT requires autoregressive generation of explicit rationales, substantially increasing latency and reducing throughput. In contrast, WhisperRec achieves latency and throughput nearly identical to UnCoT, while being substantially more efficient than CoT. These results show that latent reasoning avoids the autoregressive serving overhead of explicit CoT generation, making it well suited for online recommendation systems.

Method	Time (s) ↓	QPS ↑
OneReason-UnCoT	**3.17**	**274.56**
OneReason-CoT	55.59	15.68
WhisperRec	3.17	274.29

Table 6: Inference efficiency comparison.

Interpretability. We evaluate whether latent tokens preserve the semantic content of explicit reasoning. For each MV-ACoT view, we reconstruct reasoning semantics from latent tokens and measure their cosine similarity to the corresponding annotated Plain CoT using Qwen3-Emb-8B. As shown in Table 7, the reconstructed CoT semantics achieve consistently high similarity across Exploration, Evaluation, and Attribution (0.788–0.804). These results provide evidence that latent tokens retain key CoT semantics while reasoning in the latent space. Together with the retrieval and efficiency results, this analysis answers RQ3: latent reasoning improves recommendation performance and retains semantically aligned reasoning information, while achieving inference efficiency close to non-CoT models. Case studies are provided in Appendix B.

Number of Latent Tokens. We analyze the effect of the number of latent tokens on recommendation performance. As shown in Figure 4, Hit improves as the number of tokens increases from one to three, indicating that multiple tokens help represent fine-grained reasoning information. Further increasing the number yields no consistent gains, likely due to redundancy. We therefore use three latent tokens by default to balance reasoning capacity and compactness.

Metric	Exploration	Evaluation	Attribution
Cosine Similarity	0.8027	**0.8043**	0.7878

Table 7: Similarity between reconstructed and explicit CoT.

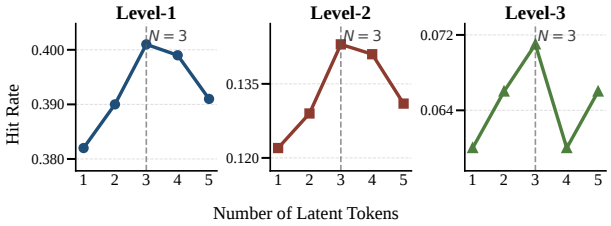

Figure 4: Ablation study on the number of latent tokens.

General Capability Analysis (RQ4)

We further examine whether latent reasoning preserves general model capabilities. Figure 5 compares explicit CoT training and latent reasoning training on MMLU and recommendation quality scores. Latent reasoning achieves higher scores across all MMLU categories and also improves the recommendation score, indicating that latent supervision causes less degradation to general abilities than direct explicit CoT training while still strengthening recommendation reasoning.

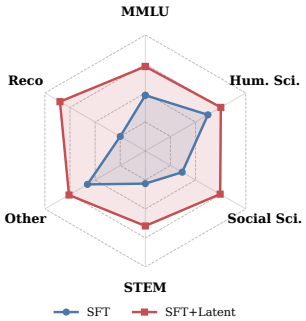

Figure 5: General and recommendation capability.

Conclusion

In this paper, we propose WhisperRec, a latent reasoning framework for foundation recommendation models. We argue that efficient reasoning is essential for deploying reasoning-enabled FRMs in industrial recommendation systems. Built on OneReason as a SOTA FRM baseline, WhisperRec addresses explicit-CoT limitations through MV-ACoT for diverse CoT construction, three-stage Latent Reasoning Alignment to internalize teacher CoT into latent tokens, and curriculum-based post-training for downstream recommendation. Experiments on public and industrial datasets show that WhisperRec outperforms conventional baselines and OneReason's *Think* and *No-Think* variants. By reasoning in the latent space, WhisperRec achieves over $10\times$ higher throughput than explicit-CoT methods, demonstrating its potential for large-scale online recommendation.

References

Chen, B.; Wang, S.; Ma, Y.; Liang, Z.; Zhang, X.; Lv, Y.; Yang, Y.; Dai, H.; Mao, L.; Zhao, T.; et al. 2026. OneSearch-V2: The Latent Reasoning Enhanced Self-distillation Generative Search Framework. *arXiv preprint arXiv:2603.24422*.

Dai, S.; Tang, J.; Wu, J.; Wang, K.; Zhu, Y.; Chen, B.; Hong, B.; Zhao, Y.; Fu, C.; Wu, K.; et al. 2025. Onepiece: Bringing context engineering and reasoning to industrial cascade ranking system. *arXiv preprint arXiv:2509.18091*.

Deng, J.; Wang, S.; Cai, K.; Ren, L.; Hu, Q.; Ding, W.; Luo, Q.; and Zhou, G. 2025. Onerec: Unifying retrieve and rank with generative recommender and iterative preference alignment. *arXiv preprint arXiv:2502.18965*.

Fang, Y.; Wang, W.; Zhang, Y.; Zhu, F.; Wang, Q.; Feng, F.; and He, X. 2025. Reason4rec: Large language models for recommendation with deliberative user preference alignment. *arXiv preprint arXiv:2502.02061*.

Gao, Y.; Sheng, T.; Xiang, Y.; Xiong, Y.; Wang, H.; and Zhang, J. 2023. Chat-rec: Towards interactive and explainable llms-augmented recommender system. *arXiv preprint arXiv:2303.14524*.

Geng, S.; Liu, S.; Fu, Z.; Ge, Y.; and Zhang, Y. 2022. Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In *Proceedings of the 16th ACM conference on recommender systems*, 299–315.

Guo, D.; Yang, D.; Zhang, H.; Song, J.; Wang, P.; Zhu, Q.; Xu, R.; Zhang, R.; Ma, S.; Bi, X.; et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Hidasi, B.; Karatzoglou, A.; Baltrunas, L.; and Tikk, D. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*.

Hou, Y.; Zhang, J.; Lin, Z.; Lu, H.; Xie, R.; McAuley, J.; and Zhao, W. X. 2024. Large language models are zero-shot rankers for recommender systems. In *European conference on information retrieval*, 364–381. Springer.

Jaech, A.; Kalai, A.; Lerer, A.; Richardson, A.; El-Kishky, A.; Low, A.; Helyar, A.; Madry, A.; Beutel, A.; Carney, A.; et al. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

Kang, W.-C.; and McAuley, J. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, 197–206. IEEE.

Kong, X.; Jiang, J.; Liu, B.; Xu, Z.; Zhu, H.; Xu, J.; Zheng, B.; Wu, J.; and Wang, X. 2026. Think before Recommendation: Autonomous Reasoning-enhanced Recommender. *Advances in Neural Information Processing Systems*, 38: 141209–141232.

Liu, Z.; Wang, S.; Wang, X.; Zhang, R.; Deng, J.; Bao, H.; Zhang, J.; Li, W.; Zheng, P.; Wu, X.; et al. 2025a. Onerec-think: In-text reasoning for generative recommendation. *arXiv preprint arXiv:2510.11639*.

Liu, Z.; Wang, S.; Wang, X.; Zhang, R.; Deng, J.; Bao, H.; Zhang, J.; Li, W.; Zheng, P.; Wu, X.; et al. 2025b. Onerec-think: In-text reasoning for generative recommendation. *arXiv preprint arXiv:2510.11639*.

Long, Z.; Wang, L.; Wu, S.; and Liu, Q. 2024. GOT4Rec: Graph of Thoughts for Sequential Recommendation. *arXiv preprint arXiv:2411.14922*.

Luo, X.; Cao, J.; Sun, T.; Yu, J.; Huang, R.; Yuan, W.; Lin, H.; Zheng, Y.; Wang, S.; Hu, Q.; et al. 2025. Qarm: Quantitative alignment multi-modal recommendation at kuaishou. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, 5915–5922.

Qwen Team. 2026. Qwen3.5: Towards Native Multimodal Agents.

Rajput, S.; Mehta, N.; Singh, A.; Hulikal Keshavan, R.; Vu, T.; Heldt, L.; Hong, L.; Tay, Y.; Tran, V.; Samost, J.; et al. 2023. Recommender systems with generative retrieval. *Advances in Neural Information Processing Systems*, 36: 10299–10315.

Rajput, S.; Mehta, N.; Singh, A.; Keshavan, R.; Vu, T.; Heldt, L.; Hong, L.; Tay, Y.; Tran, V. Q.; Samost, J.; et al. 2018. Recommender Systems with Generative Retrieval.

Sun, F.; Liu, J.; Wu, J.; Pei, C.; Lin, X.; Ou, W.; and Jiang, P. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, 1441–1450.

Tang, J.; Dai, S.; Shi, T.; Xu, J.; Chen, X.; Chen, W.; Wu, J.; and Jiang, Y. 2026. Think before recommend: Unleashing the latent reasoning power for sequential recommendation. *IEEE Transactions on Knowledge and Data Engineering*.

Team, O. 2026. OneReason Technical Report. *Technical Report*.

Team, O.; Yang, B.; Ding, B.; Chu, C.; Zang, D.; Pan, F.; Li, H.; Jiang, H.; Bao, H.; Wang, H.; et al. 2026. OneReason Technical Report. *arXiv preprint arXiv:2606.06260*.

Xing, H.; Deng, H.; Mao, Y.; Mu, L.; Hu, J.; Xu, Y.; Zhang, H.; Wang, J.; Wang, S.; Zhang, Y.; et al. 2025. Reg4rec: Reasoning-enhanced generative model for large-scale recommendation systems. *arXiv preprint arXiv:2508.15308*.

Yi, C.; Chen, D.; Guo, G.; Tang, J.; Wu, J.; Yu, J.; Zhang, M.; Chen, W.; Yang, W.; Luo, Y.; et al. 2025a. RecGPT-V2 Technical Report. *arXiv preprint arXiv:2512.14503*.

Yi, C.; Chen, D.; Guo, G.; Tang, J.; Wu, J.; Yu, J.; Zhang, M.; Dai, S.; Chen, W.; Yang, W.; et al. 2025b. Recgpt technical report. *arXiv preprint arXiv:2507.22879*.

Yu, Q.; Fu, K.; Lv, Z.; Zhang, S.; Wu, X.; Lin, C.; Wei, F.; Zheng, B.; and Wu, F. 2026. ThinkRec: Thinking-based recommendation via LLM. In *Proceedings of the ACM Web Conference 2026*, 5698–5709.

Zhai, J.; Liao, L.; Liu, X.; Wang, Y.; Li, R.; Cao, X.; Gao, L.; Gong, Z.; Gu, F.; He, M.; et al. 2024. Actions speak louder than words: Trillion-parameter sequential transducers for generative recommendations. *arXiv preprint arXiv:2402.17152*.

Zhang, B.; Luo, L.; Chen, Y.; Nie, J.; Liu, X.; Guo, D.; Zhao, Y.; Li, S.; Hao, Y.; Yao, Y.; et al. 2024. Wukong: Towards a scaling law for large-scale recommendation. *arXiv preprint arXiv:2403.02545*.

Zhang, L.; Huang, Y.; Lv, H.; Zhi, X.; Yin, M.; Ye, Y.; Guo, W.; Wang, H.; and Chen, E. 2026. Why Thinking Hurts: Diagnosing and Rectifying Linguistic Inertia in Large Language Models for Recommendation. *arXiv preprint arXiv:2602.16587*.

Zheng, B.; Hou, Y.; Lu, H.; Chen, Y.; Zhao, W. X.; Chen, M.; and Wen, J.-R. 2024. Adapting large language models by integrating collaborative semantics for recommendation. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, 1435–1448. IEEE.

Zhou, C.; Wang, M.; Ma, Y.; Wu, C.; Chen, W.; Qian, Z.; Liu, X.; Zhang, Y.; Wang, J.; Xu, H.; et al. 2025a. From perception to cognition: A survey of vision-language interactive reasoning in multimodal large language models. *arXiv preprint arXiv:2509.25373*.

Zhou, G.; Bao, H.; Huang, J.; Deng, J.; Zhang, J.; She, J.; Cai, K.; Ren, L.; Ren, L.; Luo, Q.; et al. 2025b. OpenOneRec Technical Report. *arXiv preprint arXiv:2512.24762*.

Zhou, G.; Deng, J.; Zhang, J.; Cai, K.; Ren, L.; Luo, Q.; Wang, Q.; Hu, Q.; Huang, R.; Wang, S.; et al. 2025c. Onerec technical report. *arXiv preprint arXiv:2506.13695*.

Zhou, G.; Hu, H.; Cheng, H.; Wang, H.; Deng, J.; Zhang, J.; Cai, K.; Ren, L.; Ren, L.; Yu, L.; et al. 2025d. Onerec-v2 technical report. *arXiv preprint arXiv:2508.20900*.

Zhou, G.; Mou, N.; Fan, Y.; Pi, Q.; Bian, W.; Zhou, C.; Zhu, X.; and Gai, K. 2019. Deep interest evolution network for click-through rate prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 5941–5948.

Zhou, G.; Song, C.; Zhu, X.; Fan, Y.; Zhu, H.; Ma, X.; Yan, Y.; Jin, J.; Li, H.; and Gai, K. 2017. Deep interest network for click-through rate prediction. *arXiv preprint arXiv:1706.06978*.

Appendix A: Dataset Construction and Parameters Setting

Implementation details

In our experiment, we perform full-parameter supervised fine-tuning on two complementary training streams, consisting of standard recommendation instances and chain-of-thought(CoT) augmented instances. Training is distributed over seven nodes with eight GPUs per node. We use a per-device batch size of 4 and one gradient accumulation step, resulting in an effective global batch size of 224. The model is optimized using AdamW with a peak learning rate of 1.0×10^{-4}, a cosine learning-rate schedule, a warmup ratio of 0.01, and a weight decay of 0.01. We train the model for two effective epochs in BF16 precision and enable gradient checkpointing to reduce memory consumption. The maximum training sequence length is set to 4,096 tokens. For the streaming dataset, the number of optimization steps is dynamically determined from the total number of samples in the two training streams:

$$N_{\text{step}} = \left\lceil \frac{(N_1 + N_2)E}{N_{\text{node}} N_{\text{GPU}} B_{\text{device}} N_{\text{acc}}} \right\rceil, \tag{20}$$

where N_1 and N_2 denote the numbers of samples in the two streams, $E = 2$ is the target number of epochs, $N_{\text{node}} = 7$, $N_{\text{GPU}} = 8$, $B_{\text{device}} = 4$, and $N_{\text{acc}} = 1$. We use a fixed random seed of 42 for all experiments.

Category	Hyperparameter	Value
Model configuration	Initialization	Qwen3.5-0.8B
	Fine-tuning strategy	Full-parameter fine-tuning
	Numerical precision	BF16
	Maximum sequence length	4,096 tokens
Semantic IDs	Number of semantic levels	3
	Codebook size per level	1,024
Optimization	Optimizer	AdamW
	Learning rate	1.0×10^{-4}
	Scheduler	Cosine
	Warmup ratio	0.01
	Weight decay	0.01
	Gradient checkpointing	Enabled
Training	Epochs	2
	Gradient accumulation steps	1
	Random seed	42
Streaming data	Number of training streams	2
	Maximum training steps	$\lceil 2(N_1 + N_2)/224 \rceil$
Checkpoint	Logging interval	10 steps
	Checkpoint interval	500 steps
	Maximum retained checkpoints	5
Hardware	Number of nodes	7
	GPUs per node	8

Table 8: Hyperparameters for multi-stream SFT

Construction of the supervised fine-tuning dataset.

We construct the supervised fine-tuning dataset from temporally ordered user-item interaction logs. Each item is associated with a hierarchical semantic identifier (SID), while each user is associated with a compact profile containing general preferences, local-life demands, and, when available, explicitly expressed intents.

For each user, we first sort all interactions chronologically and remove redundant interactions with the same item. When multiple interaction signals are associated with a duplicated item, we retain the strongest feedback according to the following priority:

$$\text{conversion} > \text{click} > \text{long view} > \text{impression}.$$

Each conversion event is subsequently treated as a prediction target. Specifically, for the k-th conversion event, all interactions occurring before it form the historical context, whereas the SID of the converted item serves as the supervision signal. In this way, a user with multiple conversion events may contribute multiple training instances without including the target item in its own input history.

To control the context length and emphasize recent interests, we retain at most the latest 50 interactions before each target event. Historical items are represented by their hierarchical SIDs and augmented with their corresponding behavior types, allowing the model to distinguish weak feedback, such as impressions, from stronger preference signals, such as clicks and conversions. The resulting input combines the compact user profile, local-life demand, explicit intent, and behavior-aware SID sequence. The output is the hierarchical SID of the next converted item. Formally, each training instance is written as

$$\mathcal{H}_{u,k} = \big[(s_i, a_i)\big]_{i=\max(1,\, k-L)}^{k-1}, \tag{21}$$

$$\mathcal{X}_{u,k} = [P_u;\, D_u;\, I_u;\, \mathcal{H}_{u,k}], \qquad \mathcal{Y}_{u,k} = s_k. \tag{22}$$

where P_u, D_u, and I_u denote the user profile, user demand, and explicit intent, respectively; s_i and a_i denote the SID and behavior type of the i-th interaction; and $L = 50$ is the maximum history length.

We discard instances whose user profile or target SID is unavailable. Historical items without valid SID mappings are filtered from the input sequence. Finally, all valid samples are stored in sharded JSONL files to support memory-efficient streaming and distributed training.

Construction of CoT supervision.

We employ Qwen3-235B-A22B-Instruct-2507[4] as the teacher model to generate CoT supervision. The teacher takes the user profile, local-life demand, explicit intent, and historical interactions as input. The subsequent converted item is provided only as privileged guidance, and the teacher is instructed not to reveal it directly. The resulting concise rationale is combined with the ground-truth SID to construct the CoT-enhanced supervised fine-tuning data.

[4] https://huggingface.co/Qwen/Qwen3-235B-A22B-Instruct-2507

Component	Configuration
Teacher model	Qwen3-235B-A22B-Instruct-2507
User information	User profile, user demand, and explicitly expressed intent
Interaction history	Item descriptions, behavior types, timestamps, and geographic context
Privileged guidance	Description of the subsequent converted item, used only to guide reasoning without directly revealing the target
Generated supervision	A concise rationale connecting historical behavior with the user's potential commercial intent
Final training target	Teacher-generated rationale combined with the ground-truth SID

Table 9: Configuration for constructing CoT supervision.

Appendix B: Case Study of Latent Reconstruction

Figure 6 presents a qualitative comparison between latent reconstruction and explicit CoT. Despite operating in a continuous latent space, the reconstructed trace preserves the main commercial semantics in the explicit rationale, including men's-health interest, potential lead-generation intent, price sensitivity, offline fulfillment, and dense L3 behavioral signals. The two traces are largely aligned on price sensitivity, consultation intent, and conversion-funnel stage, while differing in their assessment of geographic accessibility.

With an embedding similarity of approximately 0.80, this case provides qualitative evidence that latent tokens retain core decision-relevant semantics without directly reproducing the surface form of explicit CoT. The remaining divergence also suggests that latent reasoning can encode semantically aligned but non-identical decision paths.

Appendix C: MV-ACoT Prompt Construction Principles

Beyond the complementary Exploration, Evaluation, and Attribution tasks, MV-ACoT introduces business-aware prompt constraints to improve the quality and reliability of teacher-generated reasoning. As summarized in Table 10, these constraints ground the teacher model in local-service lead-ad recommendation, requiring it to translate content interests into actionable commercial demands and prioritize strong behavioral evidence over weak exposures. This prevents the teacher from relying on generic content preference descriptions that are insufficient for conversion-oriented recommendation decisions.

MV-ACoT further adapts reasoning paths to the strength, consistency, and uncertainty of user signals. It incorporates fulfillment mode, geographic constraints, industry-specific factors, and conversion stage when analyzing user intent. Moreover, evidence references and low-confidence or no-attribution outputs discourage hallucination, target leakage, and forced explanations. Together, these constraints enable concise reasoning for clear cases while supporting sufficiently detailed, evidence-grounded analysis for challenging recommendation scenarios.

Appendix D: CoT Quality Evaluation Prompt

We employ an LLM-as-a-Judge protocol to evaluate the quality of teacher-generated reasoning traces for local-service lead-ad recommendation. Rather than assessing whether a trace simply supports a candidate or conversion target, the protocol evaluates whether it derives a calibrated recommendation judgment from available user evidence. For each instance, the judge receives the task type, user history, supplementary user profile, candidate or converted advertisement, and generated reasoning trace.

User history is treated as the primary factual evidence. User-profile attributes and target information may provide context, but must not be backfilled into historical behaviors or used to fabricate supporting evidence. As summarized in Tables 11 and 12, the protocol combines task-specific requirements with a unified multi-dimensional scoring rubric. This design evaluates not only factuality and evidence selection, but also intent translation, causal logic, advertising fit, spatiotemporal reasoning, confidence calibration, and task fulfillment.

Hard Scoring Constraints. To enforce factual faithfulness, the judge applies explicit score caps. If a trace claims clicks, conversions, searches, consultations, or lead submissions absent from the user history, its factuality score cannot exceed 2. Similarly, backfilling candidate or target information into the user history also limits factuality to at most 2. Fabricated locations, behavior times, consumption stages, or explicit user needs limit the score to at most 3, with lower scores assigned to severe violations. These constraints prioritize evidence-grounded reasoning over target consistency or explanatory completeness.

Output Format. The judge returns a single JSON object containing the nine dimensions in Table 12. Each dimension includes an integer `score` from 1 to 5 and a brief `reason`. No markdown, additional fields, or invalid score values are permitted. Each evaluation instance is formatted as follows:

Evaluation Instance Format.

- **Task type:** `{{task_type}}`
- **Historical behaviors:** `{{history_text}}`
- **User profile:** `{{user_plain_text}}`
- **Candidate or target item:** `{{target_text}}`
- **Generated reasoning trace:** `{{reasoning_text}}`

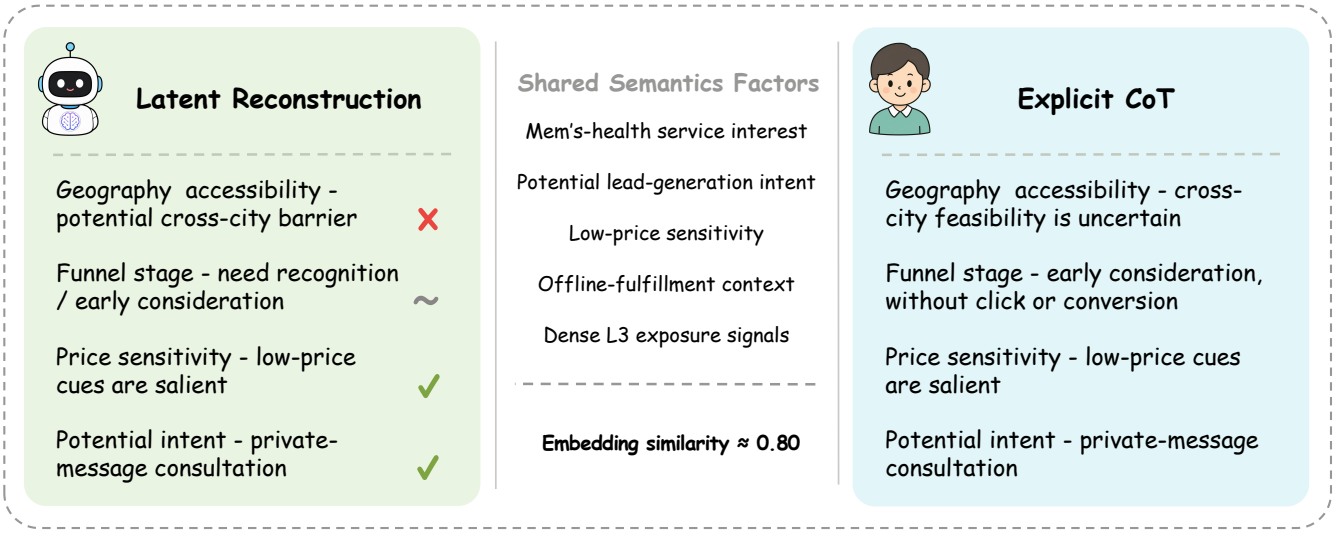

Figure 6: Comparison between latent reconstruction and explicit CoT. The two traces share core commercial semantics, including price sensitivity, consultation intent, and funnel-stage judgment, but differ in geographic accessibility. Symbols denote aligned (✓), partially aligned (∼), and divergent (×) factors.

Prompt Principle	**Design**	**Effect**
Role and task specification	Define the teacher model as a local-service and lead-ad recommendation expert, with the task restricted to lead-ad interaction and conversion reasoning.	Aligns the reasoning objective with the advertising recommendation scenario and avoids drifting to generic content recommendation or item description.
Interest-to-demand translation	Require the model to map content interests into commercial needs that can be served by lead advertisements.	Upgrades CoT from describing "what the user likes" to inferring "what commercial demand the user may have."
Behavior signal hierarchy	Categorize user behaviors into L1 conversion-level, L2 click-level, and L3 exposure-level signals, and prioritize evidence strength over frequency.	Prevents shallow exposures from being mistaken as strong intent and improves key evidence selection.
Adaptive reasoning path selection	Select fast convergence, competitive comparison, deep reasoning, or low-cost light-conversion reasoning according to the signal structure.	Enables concise reasoning for easy cases and multi-chain comparison for difficult cases, reducing redundant reasoning.
Fulfillment and industry knowledge	Distinguish online and offline fulfillment, and incorporate location, service radius, industry decision factors, and conversion stage.	Improves adaptation to local-service advertising, spatiotemporal constraints, and industry-specific conversion logic.
Evidence grounding and anti-hallucination	Require references to concrete item IDs and allow low-confidence, weak-attribution, or no-attribution conclusions.	Reduces fabricated behaviors, target leakage, and forced explanations, strengthening evidence grounding.

Table 10: Prompt construction principles of MV-ACoT for local-service lead-ad recommendation. Beyond the multi-view task design, the prompts incorporate domain-specific role constraints, demand-oriented intent translation, hierarchical behavior signals, adaptive reasoning paths, fulfillment-aware industry knowledge, and evidence-grounding mechanisms.

Component	Definition	Evaluation Focus
Task-specific requirements		
Exploration	Infer multiple potential commercial intents from user profiles and historical behaviors without observing a target advertisement.	Assess whether the trace identifies two or three plausible, diverse intent directions, translates interests into serviceable demands, and provides calibrated confidence or priority rankings.
Evaluation	Assess a candidate advertisement without assuming that it is a correct recommendation.	Assess whether the candidate is supported or rejected based on historical evidence, signal strength, user demand, and contextual constraints. Negative or low-confidence judgments are allowed.
Attribution	Explain an observed conversion to a target advertisement.	Assess whether historical evidence supports the conversion. When evidence is insufficient, weak or no attribution is preferred to post-hoc rationalization.
General evaluation principles		
Evidence priority	Treat `history_text` as the primary factual source and `user_plain_text` as supplementary information.	Profile attributes and target information must not override, alter, or be presented as historical behavioral facts.
Behavior hierarchy	Rank signals by conversion depth: conversion $>$ click $>$ exposure or viewing for more than three seconds.	Repeated shallow exposures should not be treated as stronger intent than fewer clicks or conversions.
Fulfillment awareness	Distinguish between online and offline fulfillment.	Offline services should consider location, service radius, accessibility, and demand timing; online services need not include geographic analysis unless a clear conflict exists.
Uncertainty awareness	Permit mismatch, low-confidence, weak-attribution, and no-attribution conclusions.	Reward factual faithfulness and calibrated reasoning rather than whether a trace supports the candidate or target.
Anti-hallucination	Prohibit fabricated clicks, conversions, searches, consultations, lead submissions, locations, behavior times, or explicit needs.	Prioritize factuality over target consistency or explanatory completeness; traces must not invent evidence to support a target.

Table 11: Task-specific requirements and general principles for LLM-based CoT quality evaluation.

Dimension	Evaluation Criterion	Scoring Anchors		
		Low (1)	**Medium (3)**	**High (5)**
Factuality	Faithfulness to historical behaviors without fabrication, exaggeration, chronological confusion, or target backfilling.	Substantial fabrication or severe detachment from history.	Mostly based on history but contains local misreadings or overgeneralization.	Fully grounded in history with no evident fabrication or backfilling.
Evidence selection	Selection of the most relevant, strong, recent, and decision-relevant historical evidence.	Relies on irrelevant evidence and ignores key signals.	Uses partially relevant evidence but includes noise or misses competing signals.	Accurately prioritizes key evidence and handles noise and competing directions.
Signal strength	Correct weighting of conversions, clicks, and shallow exposures according to conversion depth.	Treats weak exposure as decisive or ignores strong conversion evidence.	Generally distinguishes signal levels but occasionally overweights frequency.	Robustly weights behavior depth, direction concentration, and temporal order.
Intent translation	Translation from content interests into local-service needs and actionable advertising intent.	Remains entirely at the content-interest level.	Identifies a service category but provides limited conversion context.	Clearly identifies service needs, consumption scenarios, and potential conversion actions.
Causal logic	Coherence of the chain from historical behavior to need, commercial intent, and final judgment.	Contains severe logical gaps, contradiction, or forced causality.	The main logic holds but some inference steps or stage transitions are weak.	Forms a clear and well-supported reasoning chain with appropriate uncertainty.
Advertising fit	Alignment with lead-ad recommendation rather than ordinary content recommendation.	Only performs generic content matching.	Identifies an advertising category but lacks conversion or decision factors.	Characterizes service needs, conversion paths, and major commercial decision factors.
Spatiotemporal reasoning	Appropriate handling of time, location, fulfillment mode, service radius, and accessibility.	Ignores clear temporal or regional conflicts.	Partially considers time or location but insufficiently distinguishes fulfillment modes.	Applies restrained and appropriate spatiotemporal reasoning according to fulfillment type.
Confidence calibration	Consistency between conclusion strength or confidence and the quality of supporting evidence.	Is clearly overconfident, underconfident, or internally inconsistent.	Confidence is generally reasonable but locally insufficiently calibrated.	Confidence accurately reflects evidence strength, logical completeness, and task uncertainty.
Task fulfillment	Completion of the requirements associated with Exploration, Evaluation, or Attribution.	Fails to perform the requested reasoning task.	Completes the main task but omits diversity, confidence, attribution, or a clear judgment.	Fully follows the task-specific objective and required output structure.

Table 12: Scoring dimensions used by the LLM judge. Each dimension is rated on a five-point scale; representative low-, medium-, and high-quality anchors are provided.