KARMA: Knowledge-Action Regularized Multimodal Alignment for Personalized Search at Taobao

Zhi Sun
Taobao & Tmall Group of Alibaba
Hangzhou, China
sunzhi.sun@taobao.com

Wenming Zhang
Taobao & Tmall Group of Alibaba
Hangzhou, China
zhangwenming.zwm@taobao.com

Yi Wei
Taobao & Tmall Group of Alibaba
Hangzhou, China
songwen.wy@alibaba-inc.com

Liren Yu
Taobao & Tmall Group of Alibaba
Hangzhou, China
yuliren.ylr@taobao.com

Zhixuan Zhang
Taobao & Tmall Group of Alibaba
Hangzhou, China
zhibing.zzx@taobao.com

Dan Ou
Taobao & Tmall Group of Alibaba
Hangzhou, China
oudan.od@taobao.com

Haihong Tang
Taobao & Tmall Group of Alibaba
Hangzhou, China
piaoxue@taobao.com

Abstract

Large Language Models (LLMs) are equipped with profound semantic knowledge, making them a natural choice for injecting semantic generalization into personalized search systems. However, in practice we find that directly fine-tuning LLMs on industrial personalized tasks (e.g. next item prediction) often yields suboptimal results. We attribute this bottleneck to a critical Knowledge–Action Gap: the inherent conflict between preserving pre-trained semantic knowledge and aligning with specific personalized actions by discriminative objectives. Empirically, action-only training objectives induce Semantic Collapse, such as attention "sinks". This degradation severely cripples the LLM's generalization, failing to bring improvements to personalized search systems.

We propose KARMA (Knowledge–Action Regularized Multimodal Alignment), a unified framework that treats semantic reconstruction as a train-only regularizer. KARMA optimizes a next-interest embedding for retrieval (Action) while enforcing semantic decodability (Knowledge) through two complementary objectives: (i) history-conditioned semantic generation, which anchors optimization to the LLM's native next-token distribution, and (ii) embedding-conditioned semantic reconstruction, which constrains the interest embedding to remain semantically recoverable.

On Taobao search system, KARMA mitigates semantic collapse (attention-sink analysis) and improves both action metrics and semantic fidelity. In ablations, semantic decodability yields up to +22.5 HR@200. With KARMA, we achieve +0.25 CTR AUC in ranking, +1.86 HR in pre-ranking and +2.51 HR in recalling. Deployed online with low inference overhead at ranking stage, KARMA drives +0.5% increase in Item Click.

CCS Concepts

• **Information systems** → **Retrieval models and ranking**; • **Computing methodologies** → *Natural language processing*.

Keywords

Personalized Search, Large Language Models, Semantic Collapse, Multimodal Alignment, Representation Learning

ACM Reference Format:
Zhi Sun, Wenming Zhang, Yi Wei, Liren Yu, Zhixuan Zhang, Dan Ou, and Haihong Tang. 2026. KARMA: Knowledge-Action Regularized Multimodal Alignment for Personalized Search at Taobao. In *Proceedings of The 49th International ACM SIGIR Conference (SIGIR '2026).* ACM, New York, NY, USA, 5 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

In personalized search systems, ranking models are predominantly optimized using post-hoc user feedback(e.g. CTR prediction)[3, 4, 14, 16]. This optimization paradigm inevitably privileges post-hoc memorization features (item IDs, co-occurrence statistics, etc.) over semantic generalization features. While memorization features precisely capture user preferences for popular items, yielding excellent performance on head traffic, they expose a critical vulnerability: when faced with sparse feedback scenarios, such as long-tail queries or cold-start items, this lack of semantic generalization leads to a sharp decline in ranking quality. This naturally motivates leveraging the rich, pre-trained semantic knowledge of LLMs to compensate for the system's generalization capabilities[6, 9, 11, 15].

Although integrating LLMs is intuitively appealing, directly fine-tuning LLMs for personalized ranking often yields limited performance gains. We attribute this empirical bottleneck to a critical Knowledge-Action Gap: the inherent conflict between rich pre-trained semantics (Knowledge) and user behavioral patterns driven by discriminative feedback (Action). This gap creates a severe trade-off: over-optimizing for Action severely distorts the pre-trained

SIGIR '2026, Melbourne | Naarm, Australia

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/XXXXXXX.XXXXXXX

semantic space, while freezing the LLM yields representations too coarse-grained for personalization.

Constrained by latency, industrial systems typically compress historical items (text and image) into single embeddings[2, 13] for the next-item prediction task[7]. However, this embedding-only adaptation triggers Semantic Collapse. We observe that attention maps degenerate into "barcode-like" patterns, acting as a destructive attention sink. By exploiting shortcut cues to distinguish positives from negatives, the LLM bypasses genuine semantic modeling, leading to severe semantic degradation.

This paper proposes a principle for continuous-token personalization: semantic decodability should be enforced as a train-only regularizer, rather than treated as an auxiliary generation capability. We introduce KARMA (Knowledge–Action Regularized Multimodal Alignment), which learns a next-interest embedding optimized for retrieval and ranking (Action), while preserving LLM semantics (Knowledge) via two non-conflicting decodability objectives. The first decodes the target item's semantics from the behavioral history, anchoring training to the LLM's native token distribution. The second decodes the same semantics conditioned explicitly on the model-produced next-interest embedding, enforcing an embedding bottleneck that prevents ID-like shortcut solutions. When multimodal supervision is available, we further reconstruct frozen visual semantic features as an additional grounding signal. Figure 1 shows the architecture of KARMA.

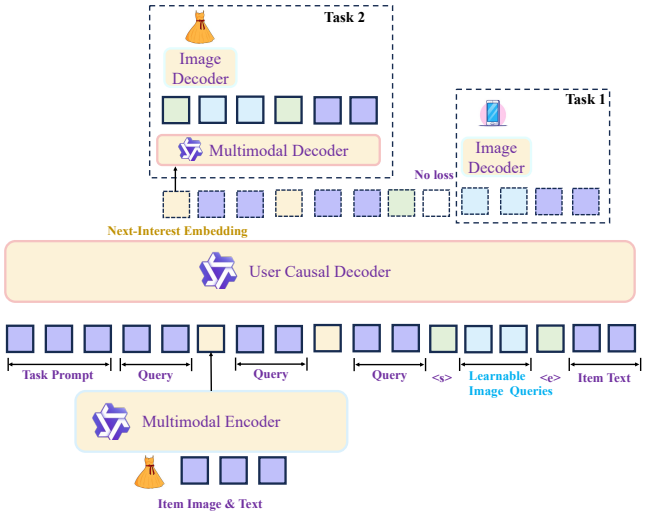


Figure 1: KARMA architecture: A train-only regularizer that bridges the Knowledge–Action Gap.

Contributions. (1) We identify Semantic Collapse as a key failure mode of LLM personalization under continuous-token interfaces, supported by attention-sink analysis and semantic metrics. (2) We propose KARMA, which bridges the Knowledge–Action Gap by enforcing semantic decodability as a train-only regularizer for action-aligned retrieval embeddings. (3) We demonstrate substantial offline and online gains in a large-scale search engine with zero inference overhead.

2 Methodology

2.1 Continuous-Token Personalized Search

At impression time t, user u issues query q_t with a chronological sequence of positive interactions

$$\mathcal{H}_t = \{(q_1, i_1), \ldots, (q_{t-1}, i_{t-1})\} \tag{1}$$

where each item i contains multimodal content $\mathbf{x}_i = (\mathbf{x}_i^{\text{text}}, \mathbf{x}_i^{\text{img}})$. Our goal is to compute a dense next-interest representation $\mathbf{h}_t \in \mathbb{R}^d$ for retrieval and ranking.

Efficiency constraint. Feeding raw tokens of all historical items into an LLM is infeasible in industrial search. We therefore adopt a continuous-token interface: an item encoder compresses each item into a single embedding token

$$\mathbf{e}_i = E_\phi(\mathbf{x}_i) \in \mathbb{R}^d \tag{2}$$

and a decoder-only Transformer D_θ performs causal modeling over the history token sequence $[\mathbf{e}_{i_1}, \ldots, \mathbf{e}_{i_{t-1}}]$ (optionally conditioned on the current query via a lightweight query embedding). The final hidden state is used as the next-interest vector:

$$\mathbf{S}_{t-1} = D_\theta([\mathbf{e}_{i_1}, \ldots, \mathbf{e}_{i_{t-1}}]), \qquad \mathbf{h}_t = \mathbf{S}_{t-1}[-1] \tag{3}$$

Item embeddings $\mathbf{e}_i$ are indexed for ANN retrieval and used for dot-product scoring.

This interface is computationally attractive, but it is also where we observe Semantic Collapse: when trained with purely discriminative objectives, the model may treat $\mathbf{e}_i$ as opaque identifiers and converge to shortcut, ID-centric solutions.

2.2 KARMA: Knowledge–Action Regularization via Decodability

KARMA is built on one requirement: the representation $\mathbf{h}_t$ should be action-aligned for ranking, while remaining semantically decodable back to the target item's original semantics. We optimize the following objective:

$$\mathcal{L}_{\text{KARMA}} = \mathcal{L}_{\text{act}} + \lambda_{\text{dec}} \mathcal{L}_{\text{dec}} \tag{4}$$

where $\mathcal{L}_{\text{act}}$ injects behavioral supervision (Action), and $\mathcal{L}_{\text{dec}}$ regularizes the solution space by enforcing decodability (Knowledge).

Two decodability paths. We implement $\mathcal{L}_{\text{dec}}$ with two complementary paths. (i) History-conditioned semantic generation decodes the target semantics directly from the behavioral history, anchoring optimization to the LLM's native next-token distribution. (ii) Embedding-conditioned semantic reconstruction decodes the same semantics conditioned explicitly on the model-produced next-interest embedding $\mathbf{h}_t$, enforcing $\mathbf{h}_t$ to remain semantically recoverable and preventing ID-like shortcut minima. These objectives regularize different parts of the system and can be jointly optimized.

Train-only regularization. Decoding heads are attached only during training and removed at inference, yielding zero additional online latency compared to standard embedding-based retrieval.

2.3 Action Alignment Objective

Given the ground-truth next clicked item i_t, we rank its embedding $\mathbf{e}_{i_t}$ above negatives $\{\mathbf{e}_j\}_{j\in\mathcal{N}_t}$, where $\mathcal{N}_t$ contains exposed-but-unclicked items (hard negatives) and in-batch negatives. We use a pairwise cross-entropy objective:

$$\mathcal{L}_{\text{act}} = \sum_t \sum_{j\in\mathcal{N}_t} -\log\sigma\big(\mathbf{h}_t^\top \mathbf{e}_{i_t} - \mathbf{h}_t^\top \mathbf{e}_j\big) \tag{5}$$

This form matches logged ranking supervision with heterogeneous negative hardness and is empirically more stable than temperature-scaled InfoNCE in our setting.

2.4 Knowledge Regularization: Multimodal Decodability

2.4.1 Task 1: History-Conditioned Semantic Generation. Let $\mathbf{y}_{i_t} = [w_1, \ldots, w_L]$ denote the target item's text tokens. We decode the target text from the behavioral history:

$$\mathcal{L}_{\text{gen}} = -\sum_{l=1}^{L} \log p_\theta(w_l \mid w_{<l}, \mathcal{H}_t) \tag{6}$$

This term anchors training to the LLM's native token-level objective, mitigating catastrophic forgetting and discouraging the continuous-token interface from discarding recoverable semantics.

2.4.2 Task 2: Embedding-Conditioned Semantic Reconstruction. Our main anti-collapse constraint requires the next-interest embedding to be semantically decodable:

$$\mathcal{L}_{\text{recon}} = -\sum_{l=1}^{L} \log p_\theta(w_l \mid w_{<l}, \mathbf{h}_t) \tag{7}$$

By enforcing an explicit embedding bottleneck, shortcut solutions that treat continuous item embeddings as opaque identifiers become suboptimal.

2.4.3 Visual Decodability as a Shared Modality Decoder. When multimodal supervision is available, we reconstruct a frozen visual semantic feature $\mathbf{v}_{i_t}$ (from a pretrained vision encoder) with a conditional diffusion/flow-matching head g_ψ. The head can be conditioned on either history states (Task 1) or $\mathbf{h}_t$ (Task 2). We optimize a standard diffusion/flow-matching training loss:

$$\mathcal{L}_{\text{img}} = \mathbb{E}_\tau\big[\ell_{\text{diff}}(g_\psi; \mathbf{v}_{i_t}, \mathbf{c}_t, \tau)\big] \tag{8}$$

where $\mathbf{c}_t$ denotes the conditioning signal and ℓ_{diff} follows standard formulations.

2.4.4 Final Decodability Regularizer. We instantiate the decodability regularizer as

$$\mathcal{L}_{\text{dec}} = \mathcal{L}_{\text{gen}} + \mathcal{L}_{\text{recon}} \qquad \mathcal{L}_{\text{dec}}^{\text{mm}} = \mathcal{L}_{\text{gen}} + \mathcal{L}_{\text{recon}} + \lambda_{\text{img}}\mathcal{L}_{\text{img}} \tag{9}$$

2.5 Bridging the Modality Gap with Staged Alignment

Continuous item embeddings are out-of-distribution relative to the LLM's discrete-token pretraining. We therefore adopt a two-stage schedule: (i) a semantic warm-up that teaches the decoder to decode item semantics from isolated $\mathbf{e}_i$, aligning the continuous-token interface to the LLM space; and (ii) joint training on behavioral sequences with $\mathcal{L}_{\text{KARMA}}$, where action alignment learns personalization and decodability prevents collapse.

2.6 Inference: Train-Heavy, Infer-Light

At inference, KARMA uses only the efficient embedding path: encode history items into $\mathbf{e}$ tokens, run the causal decoder to obtain $\mathbf{h}_t$, and score candidates by dot-product similarity with item embeddings. All decoding heads are disabled, incurring zero additional online latency.

3 Experiments

We evaluate KARMA in **Taobao Search**, one of the largest e-commerce search engines. The experiments are organized around three questions: (i) whether action-only training under continuous tokens exhibits semantic collapse, (ii) whether train-only decodability regularization mitigates collapse and improves retrieval/ranking, and (iii) how multimodal reconstruction and diffusion-based embedding generation behave under retrieval requirements. We additionally report online A/B results to validate practical impact under strict latency constraints.

3.1 Experimental Setup

Data. We use anonymized search logs with chronological splits. Each instance contains a user history of positive interactions, the next clicked item, and exposed-but-unclicked items as hard negatives.

Metrics. We report action metrics (gAUC for ranking discrimination; HR@K for retrieval/pre-ranking) and a lightweight semantic probe JS@50 (term-level Jaccard overlap between the ground-truth item and retrieved top-K items), used to detect shortcut learning when action metrics remain competitive.

Implementation. Unless specified, we use Qwen3-0.6B [12] as the user decoder with the continuous-token interface; item encoders are Qwen3-1.7B (text) or Qwen3-VL-2B (multimodal) [1]. Decoding/reconstruction heads are train-only and removed at inference.

3.2 Main Results: Decodability Regularization Mitigates Semantic Collapse

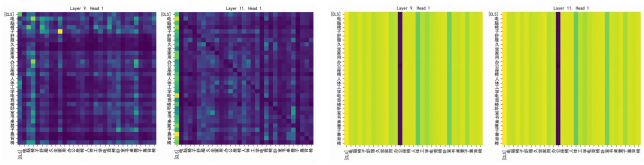
Table 1 summarizes the effect of KARMA components, reported as improvements over the action-only baseline trained with Pairwise-CE. We highlight two observations.

(1) Action-only training is prone to shortcut learning. Under the continuous-token interface, optimizing only $\mathcal{L}_{\text{act}}$ yields reasonable action metrics but comparatively weak semantic fidelity (JS@50), consistent with the model relying on non-semantic shortcuts rather than retaining decodable semantics in the interest embedding.

(2) Embedding-conditioned decodability is the primary anti-collapse constraint. History-conditioned semantic generation (Task 1) provides a modest action gain but does not directly constrain the retrieval bottleneck $\mathbf{h}_t$. In contrast, embedding-conditioned reconstruction (Task 2) enforces that $\mathbf{h}_t$ remains semantically recoverable, leading to substantially larger retrieval improvements. Combining Task 1 and Task 2 yields the best overall performance, suggesting they regularize complementary failure modes.

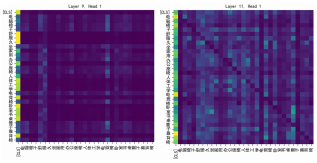
Table 1: Impact of decodability regularization (Δ over action-only Pairwise-CE).

Variant	ΔgAUC	ΔHR@200	ΔHR@1000	ΔJS@50
Text-only decodability				
+ Task 1: History→Text generation ($\mathcal{L}_{\text{gen}}$)	+0.43	+4.81	+3.46	-0.87
+ Task 2: $\mathbf{h}_t$ →Text reconstruction ($\mathcal{L}_{\text{recon}}$)	+0.43	+19.19	+19.31	+2.65
KARMA: Task 1 + Task 2	**+0.97**	**+22.57**	**+21.19**	+2.26
Multimodal extension (incremental over the best text-only KARMA)				
+ Visual semantic reconstruction	**+1.38**	**+10.83**	**+8.41**	+0.40



(a) Frozen pretrained item encoder (no task training).

(b) Action-only training with $\mathcal{L}_{\text{act}}$ exhibits attention sinks (barcode-like patterns).



(c) KARMA reduces attention sinks and restores more distributed attention.

Figure 2: Attention maps from certain layers (Layers 9/11, Head 1) of the item encoder under three training regimes.

3.3 Diagnosing Collapse: Attention Sinks as Qualitative Evidence

To better understand the failure mode, we visualize attention maps in the item encoder. Figure 2 shows that action-only training often produces sparse, barcode-like attention patterns that concentrate mass on a small subset of positions. This *attention sink* behavior is consistent with shortcut learning under discriminative supervision: the encoder may over-focus on a few highly predictive cues while ignoring broader semantic content. With KARMA, attention becomes more distributed, suggesting that decodability regularization encourages representations that remain grounded in recoverable item semantics.

Note on interpretation. We treat attention sinks as a diagnostic signal rather than a causal proof. The main quantitative evidence remains the consistent gains in Table 1 across action metrics and semantic fidelity proxies, together with the fact that all regularizers are removed at inference (zero online overhead).

3.4 Multimodal Grounding and the Mode–Mean Dilemma

Multimodal grounding. When multimodal supervision is available, we add a lightweight diffusion/flow-matching head to reconstruct frozen visual semantic features of the target item. As shown in Table 1, visual reconstruction provides additional gains beyond text-only KARMA, indicating that cross-modal decodability offers complementary grounding signals for personalization.

Table 2: Generating retrieval embeddings with diffusion (Δ over the best text-only KARMA).

Embedding Generator	ΔgAUC	ΔHR@200	ΔHR@1000	ΔJS@50
AR + MSE	**+0.63**	**+4.21**	**+1.70**	**+0.85**
AR + DDPM[5]	+0.17	+0.80	-1.01	+0.37
AR + EDM[8]	-0.12	-2.95	-3.30	-0.33
AR + Flow-Matching[10]	+0.02	-0.83	-2.68	+0.26

Negative result: diffusion is a poor generator for retrieval embeddings. We further tested whether diffusion can replace discriminative objectives to generate the retrieval embedding $\mathbf{h}_t$. Table 2 reports results *relative to the best text-only KARMA*. Diffusion-based embedding generation underperforms, while a simple AR+MSE objective performs better.

Insight: mode-seeking vs. mean-seeking. This discrepancy reflects a boundary between generative modeling and retrieval representation learning. Diffusion objectives are naturally *mode-seeking*—they aim to sample high-fidelity instances from a conditional distribution. In contrast, retrieval embeddings are *mean-seeking*—they should act as a stable centroid that aggregates multiple plausible future intents to support nearest-neighbor retrieval. Consequently, diffusion is effective as a semantic reconstruction regularizer, but suboptimal as the generator of retrieval embeddings.

3.5 Online Deployment: Funnel-Wide Gains with Zero Inference Overhead

We applied KARMA across three stages of the system: ranking, pre-ranking, and recalling. The offline performance is presented as follows:

Ranking: Compute the similarity between the user's historical items and the target item as an input feature for the CTR model. +0.25 AUC.

Pre-ranking: Weighted fusion of pre-ranking scores and KARMA scores. +1.86 HR@500.

Recalling: Serving as an embedding-based retrieval. +2.51 HR @5000.

We deployed KARMA in production. Since all decodability heads are used only during training, the online serving path is identical to the action-only baseline. Over a 14-day A/B test at ranking stage, KARMA drives +0.5% increase in Item Click.

4 Conclusions

In this paper, we show that action-only discriminative training can trigger Semantic Collapse (e.g., attention sinks) and degrade semantic fidelity, limiting generalization. KARMA enforces *train-only* semantic decodability via history-conditioned generation and embedding-conditioned reconstruction (optionally with visual feature reconstruction), preventing ID-like shortcut solutions while keeping the inference path unchanged. Across large-scale Taobao offline experiments and online A/B tests, KARMA consistently improves retrieval and ranking with zero serving overhead; we also find diffusion is better used as a multimodal semantic regularizer than as a mean-seeking retrieval-embedding generator.

References

[1] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. 2025. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631* (2025).

[2] Junyi Chen, Lu Chi, Bingyue Peng, and Zehuan Yuan. 2024. Hllm: Enhancing sequential recommendations via hierarchical large language models for item and user modeling. *arXiv preprint arXiv:2409.12740* (2024).

[3] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*. 7–10.

[4] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM conference on recommender systems*. 191–198.

[5] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.

[6] Jian Jia, Yipei Wang, Yan Li, Honggang Chen, Xuehan Bai, Zhaocheng Liu, Jian Liang, Quan Chen, Han Li, Peng Jiang, et al. 2025. LEARN: knowledge adaptation from large language model to recommendation for practical industrial application. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 11861–11869.

[7] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*. IEEE, 197–206.

[8] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. 2022. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems* 35 (2022), 26565–26577.

[9] Dong-Ho Lee, Adam Kraft, Long Jin, Nikhil Mehta, Taibai Xu, Lichan Hong, Ed H Chi, and Xinyang Yi. 2024. Star: A simple training-free approach for recommendations using large language models. *arXiv preprint arXiv:2410.16458* (2024).

[10] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. 2022. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747* (2022).

[11] Xiang-Rong Sheng, Feifan Yang, Litong Gong, Biao Wang, Zhangming Chan, Yujing Zhang, Yueyao Cheng, Yong-Nan Zhu, Tiezheng Ge, Han Zhu, et al. 2024. Enhancing taobao display advertising with multimodal representations: Challenges, approaches and insights. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. 4858–4865.

[12] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388* (2025).

[13] Yuhao Yang, Zhi Ji, Zhaopeng Li, Yi Li, Zhonglin Mo, Yue Ding, Kai Chen, Zijian Zhang, Jie Li, Shuanglong Li, et al. 2025. Sparse meets dense: Unified generative recommendations with cascaded sparse-dense representations. *arXiv preprint arXiv:2503.02453* (2025).

[14] Liren Yu, Wenming Zhang, Silu Zhou, Tao Zhang, Zhixuan Zhang, and Dan Ou. 2025. HHFT: Hierarchical Heterogeneous Feature Transformer for Recommendation Systems. *arXiv preprint arXiv:2511.20235* (2025).

[15] Chao Zhang, Shiwei Wu, Haoxin Zhang, Tong Xu, Yan Gao, Yao Hu, and Enhong Chen. 2024. Notellm: A retrievable large language model for note recommendation. In *Companion Proceedings of the ACM Web Conference 2024*. 170–179.

[16] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1059–1068.