

SOLARIS: Speculative Offloading of Latent-bAsed Representation for Inference Scaling

Zikun Liu, Liang Luo, Qianru Li, Zhengyu Zhang, Wei Ling, Jingyi Shen, Zeliang Chen, Yaning Huang, Jingxian Huang, Abdallah Aboelela, Chonglin Sun, Feifan Gu, Fenggang Wu, Hang Qu, Huayu Li, Jill Pan, Kaidi Pei, Laming Chen, Longhao Jin, Qin Huang, Tongyi Tang, Varna Puvvada, Wenlin Chen, Xiaohan Wei, Xu Cao, Yantao Yao, Yuan Jin, Yunchen Pu, Yuxin Chen, Zijian Shen, Zhengkai Zhang, Jing Zhu, Dong Liang, Ellie Wen
Meta AI
Menlo Park, California, USA

Abstract

Recent advances in recommendation scaling laws have led to foundation models of unprecedented complexity. While these models offer superior performance, their computational demands make real-time serving impractical, often forcing practitioners to rely on knowledge distillation—compromising serving quality for efficiency.

To address this challenge, we present SOLARIS (Speculative Offloading of Latent-bAsed Representation for Inference Scaling), a novel framework inspired by speculative decoding. SOLARIS proactively precomputes user-item interaction embeddings by predicting which user-item pairs are likely to appear in future requests, and asynchronously generating their foundation model representations ahead of time. This approach decouples the costly foundation model inference from the latency-critical serving path, enabling real-time knowledge transfer from models previously considered too expensive for online use.

Deployed across Meta’s advertising system serving billions of daily requests, SOLARIS achieves 0.67% revenue-driving top-line metrics gain, demonstrating its effectiveness at scale.

1 Introduction

In recent years, foundation models (FMs) have revolutionized large-scale recommendation systems (RecSys), unlocking unprecedented modeling capacity, richer data utilization, and enhanced personalization. Modern FMs [20, 21, 36] excel at learning complex user-item interactions from massive, heterogeneous datasets, and consistently achieved performance gains in offline evaluations.

Despite these advances, deploying FMs in production remains a major challenge. Their computational complexity makes real-time serving impractical, leading practitioners to rely on knowledge transfer to smaller, vertical models (VMs).

Unfortunately, the industry standard, knowledge distillation, faces two bottlenecks:

- (1) **Limited Transfer Ratio and Generalizability:** soft-label based [2, 11, 17, 30] distillation typically achieves only a 20–25% transfer ratio and the resulting prediction labels are tightly coupled to specific tasks.
- (2) **Limited Availability and Coverage:** Knowledge transfer occurs exclusively during training when labels are available [7, 9, 13], while missing opportunities for knowledge sharing at inference time.

To address these challenges, we propose SOLARIS (Speculative Offloading of Latent-bAsed Representation for Inference Scaling)—a scalable, high-transfer-ratio, inference-time ready and production-friendly framework for large scale knowledge sharing in RecSys. SOLARIS introduces the following key innovations:

- (1) **Direct Embedding-based Transfer:** SOLARIS moves beyond soft-label distillation by directly transferring user-item interaction embeddings from the foundation model to VMs. This approach yields richer, more generalizable representations and significantly better knowledge transfer ratio.
- (2) **Speculative Embedding Precomputation:** Inspired by speculative decoding in LLMs [18], SOLARIS anticipates which user-item interactions are likely to occur in the near future and precomputes their embeddings asynchronously. This enables inference-time knowledge distillation, allowing VMs to leverage foundation model representations during serving.
- (3) **Hierarchical Feature Enrichment:** To maximize coverage without incurring linear distillation costs, SOLARIS employs hierarchical feature enrichment. When a direct user-item match is unavailable, it aggregates user and item embeddings and similarity-based representations to continue providing knowledge transfer benefits.

By decoupling the expensive knowledge transfer process from the latency-critical serving path, SOLARIS delivers richer user-item representations at inference time with no additional overheads. SOLARIS seamlessly integrates with existing multi-stage ranking pipelines and scales to billions of daily requests, all while maintaining strict latency requirements in Meta’s advertising infrastructure.

Our extensive evaluation of SOLARIS in real world production environment demonstrates up to 0.2% relative log loss (RLL) improvements on 10+ production models, lifting coverage by 30%, and resulting in 0.67% global ads revenue gain (on the order of \$100M).

2 Background

2.1 Multi-stage items Ranking

Modern RecSys employs a multi-stage pipeline [25, 33, 39] to efficiently select relevant items. The process consists of three stages: 1) Retrieval stage [8, 14, 19], where it filters items using broad, lightweight criteria such as user interests and ad targeting. 2) Early-stage ranking [22, 40], where it applies a lightweight deep learning

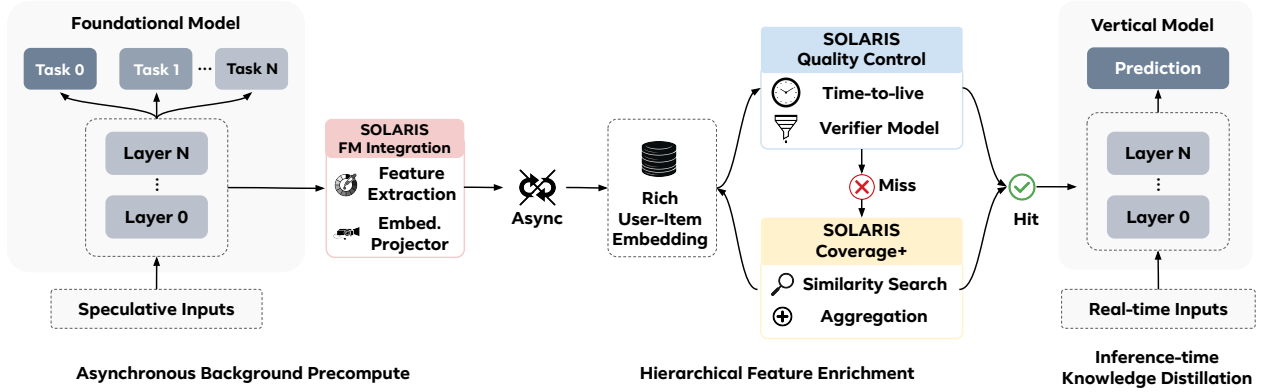


Figure 1: SOLARIS overview

model with user and ad features to narrow the selection to hundreds of items. 3) Final stage ranking [3, 10, 23], where it uses resource-intensive models that analyze thousands of signals, including real-time user activity, to select the top items for auction and delivery. In our system, SOLARIS serves the final stage ranking models.

2.2 Knowledge Transfer

Knowledge transfer is a fundamental technique in modern recommendation systems, enabling models to leverage information from data-rich domains or large teacher models to enhance performance in downstream or data-scarce scenarios.

Knowledge Distillation: The teacher model provides soft labels—probability distributions over possible classes—to enable the student to learn not only the correct answers but also the relative similarities among classes [2, 11, 17, 30]. However, this method is inherently limited by the amount of information that can be conveyed through soft labels alone in ranking systems, typically lower than 25% [30].

Embedding Sharing: Another widely used technique, where representations learned by upstream models are reused as features in downstream tasks [4, 5, 15, 16, 28, 38]. While effective, embedding sharing can introduce significant computational and storage overhead.

3 SOLARIS Design

3.1 System Overview

Figure 1 shows the overall SOLARIS architecture. The framework is built on the insight that foundation models (FMs) learn rich user-item interaction representations in their intermediate layers, which capture holistic knowledge from massive datasets, diverse features, and multi-objective optimization.

In SOLARIS, we specifically extract embeddings from the last shared layer of the FM architecture, where concentrates the most relevant knowledge about user-item interactions. Such embedding computation happens asynchronously in the background and we designed the hierarchical feature enrichment module to obtain embeddings with boosted coverage and guaranteed freshness. These

embeddings are used as input for downstream models to enhance performance.

3.2 SOLARIS Knowledge Sharing

Traditional soft-label distillation [1, 6, 24] transfers knowledge only during training with limited efficacy. We design SOLARIS to address this fundamental limitation by enabling continuous knowledge transfer using a unified, FM-agnostic paradigm.

3.2.1 Foundational Model Compatibility. SOLARIS applies to all multi-task multi-label style FMs as shown in Figure 1. As an example, many SOTA FMs process heterogeneous input features through multiple parallel pathways [21, 34]: (1) a dense architecture that transforms continuous features into low-dimensional embeddings via multi-layer perceptrons, (2) a sparse architecture that converts categorical features into pooled embeddings through embedding bag collections, and (3) a variable-length embedding architecture that captures sequential user behavior patterns from event-based features [12, 26, 29, 32]. The outcomes of the three paths are then processed via a shared stack of interaction layers [20, 31, 36, 37], producing the shared network output.

3.2.2 SOLARIS Feature Extraction. The user-item embeddings are derived from the output of our shared feature interaction network, which processes heterogeneous input features—including user demographics, behavioral sequences, and item attributes. This embedding is then compressed via an autoencoder [27, 35] to produce a compact, fixed-dimensional vector suitable for downstream consumption. This embedding serves as additional input features to vertical models. This knowledge sharing mechanism enables vertical models to leverage the rich user-item interaction patterns learned from the FM trained across different domains.

3.2.3 Knowledge Sharing During Inference Time. The rich embeddings produced by FMs are stored alongside other input features, and are consumed just like other features by the VMs, including during serving. And online training is conducted on FM to adapt to shifts in distribution.

3.3 Asynchronous Precompute

However, it is impractical to generate FM user-item interaction for each user-item pair from the user and items pool, due to prohibitive

infrastructure compute and storage cost. Therefore, we propose to generate the embeddings selectively via asynchronous precompute. At a high level, SOLARIS maintains a background service to compute user-item embeddings using a FM. It takes user-level requests, evaluates item candidates in those requests, and cache the generated user-item embeddings to be consumed by future requests. The async precompute process relaxed the latency budget by bringing embedding generation to the background. As a trade-off, this approach compromises feature freshness as well as comprehensive coverage at user-item level.

Background Teacher Computation: When a user request comes in, multi-stage ranking phases narrow down the items candidates to hundreds. The FMs maintain background jobs that continuously generates the user-item embedding from top 20% most relevant items to the user (decided by verifier model mechanism) from these candidates. The computed embeddings are stored in a distributed cache, keyed by $\langle \text{user}, \text{item} \rangle$ pairs.

Verifier Model Mechanism: To further improve efficiency and cache quality, we employ a lightweight verifier model—specifically, the prediction score from the current vertical model—to decide whether a user-item pair should be retained or discarded from the cache. Only user-item pairs with sufficiently high predicted relevance are selected for embedding precomputation and caching. This mechanism acts as a real-time filter, ensuring that computational resources are focused on the most promising candidates, and aligns with the speculative decoding paradigm: we “speculate” which user-item pairs will be needed and discard the rest.

TTL-based Cache Invalidation: The success of asynchronous precomputation is contingent upon the stability of a user’s preferences and interests within a short time frame. In early-stage ranking, since user preferences typically remain consistent within a few hours (it has been observed that over 60% of item candidates in the current request were ranked within the past six hours), teacher embeddings retain their relevance, enabling effective vertical model enhancement during inference without requiring real-time teacher computation. During both training and inference, vertical models query the cache for each user-item pair to retrieve precomputed embeddings. The system enforces a cache reading TTL (typically a few hours) to ensure embedding freshness while balancing computational efficiency.

Vertical Model Serving: As described in Section 3.2, vertical models takes the user-item embeddings as additional input features to improve predictions. When a $\langle \text{user}, \text{item} \rangle$ query is missing or expired from the cache, the user-item pair is queued for the SOLARIS model’s next refresh cycle while zero tensors serve as placeholders in the vertical models input.

3.4 Hierarchical Feature Enrichment

While asynchronous precomputation enables efficient knowledge transfer, achieving complete coverage for all user-item pairs remains challenging due to the vast scale of the user and item space, and the infrastructure concerns. As a result, the vanilla user-item embedding coverage is approximately 50% in our system. This coverage gap is particularly pronounced for long-tail items, new users, or emerging user-item interactions that haven’t been prioritized in recent precomputation cycles. To address the coverage limitations,

SOLARIS implements 2 complementary strategies that improves embedding availability while maintaining system efficiency.

Aggregated user-item Embedding: SOLARIS introduces an aggregated user-only embedding, computed by averaging all available user-item embeddings for a given user across different items within the past 24 hours, explicitly excluding the embedding for the current user-item pair. This approach leverages the assumption that user preferences exhibit short-term consistency, allowing recent interaction embeddings to inform current predictions. The aggregated user-item embedding serves as an additional input feature, supplementing the standard user-item embedding, and increases effective feature coverage to approximately 85–90%, ensuring that most users have some form of foundation model representation available during inference.

Similarity-Based Embedding: SOLARIS maintains a precomputed user clustering table that captures user similarity relationships. The clustering process operates in two phases: first, a dedicated FM generates comprehensive user embeddings that encode rich behavioral representations; second, a k-nearest neighbors (KNN) algorithm identifies similar users for each target user based on cosine similarity in the embedding space. For each user-item pair lacking a precomputed embedding, SOLARIS queries the clustering table to identify the top similar users to the target user. Among these similar users, SOLARIS searches for available user-item embeddings of form $\langle \text{neighbor}, \text{items} \rangle$. SOLARIS then employs one of two strategies for leveraging neighbor embeddings: (1) aggregating multiple neighbor embeddings using distance-weighted averaging, where closer users receive proportionally higher weights, or (2) selecting the embedding from the single most similar neighbor based on proximity in the user feature space. This similarity-based approach extends the foundation model’s effective coverage by 30%. The method leverages collaborative filtering principles, assuming that users with similar behavioral patterns will exhibit comparable responses to identical advertisements, thereby providing meaningful signal for previously unseen interactions.

4 Experiments

We now comprehensively evaluate SOLARIS with real-world serving scenarios, demonstrating its effectiveness in lifting serving quality without latency overhead. We then carefully ablate the contribution from individual SOLARIS components.

4.1 Deployment Baseline and Methodology

While our framework is applicable to general user-item scenarios, we demonstrate its effectiveness on Meta ads in our experiments. We have deployed SOLARIS across a wide range of advertising model types serving global users across platform. Our baseline is prior production system without SOLARIS, where each portfolio was served by a separate model trained on isolated datasets. During testing, both SOLARIS and the baseline predict CTR(Click-Through Rate)/CVR(Conversion Rate) metrics for live traffic in the form of (user, ad) pairs, with the highest-ranked pairs presented to users based on predictions and other factors.

Task Categories	Sub-tasks	Relative BCE Loss ↓ (%)
CTR	Facebook Feed	0.09
	Facebook Reels	0.08
	Instagram	0.05
	Instagram Link Click	0.05
CVR	Facebook Feed + Reels	0.1
	Offsite Conversion	0.05

Table 1: SOLARIS Embedding Improves Vertical Model Performance on Varieties of Tasks

SOLARIS Feature Coverage	20%	50%	60%	100%
BCE ↓ on Instagram CTR	0.05%	0.1%	0.13%	0.3%

Table 2: SOLARIS Feature Coverage Study

4.2 Results on Vertical Models

Table 1 summarizes the impact of integrating SOLARIS user-ad embeddings on a diverse set of CTR and CVR tasks across Meta’s major ad products. Here, we report the relative reduction in binary cross-entropy (BCE) loss (%) for each sub-task, reflecting improvements in model prediction quality. Sub-tasks are defined as distinct prediction objectives within each task category, corresponding to specific surfaces or user actions (e.g., Facebook Feed CTR, Instagram Link Click CTR, Offsite Conversion).

CTR Gains: SOLARIS user-ad embeddings consistently improved CTR prediction across all evaluated surfaces. Specifically, we observed a 0.09% reduction in BCE loss for Facebook Feed, 0.08% for Facebook Reels, and 0.05% for both Instagram Feed and Instagram Link Click sub-tasks. These improvements indicate that embedding sharing enables the model to better capture user-ad interaction patterns, leading to more accurate click probability estimates. Notably, the gains are present not only in primary CTR tasks (e.g., Feed, Reels) but also in auxiliary tasks such as Instagram Link Click, demonstrating the generality of the approach.

CVR Gains: For conversion rate tasks, SOLARIS led to clear reductions in prediction loss and higher conversion rates across both on-platform and offsite events. The model achieved a 0.10% reduction in BCE loss for the combined Facebook Feed + Reels conversion task, and a 0.05% reduction for offsite conversion (e.g., purchases or sign-ups occurring outside Meta platforms). These results highlight the effectiveness of embedding-based knowledge transfer in improving conversion prediction, especially in scenarios where conversion signals are sparse or user intent is more difficult to model, showing the value of SOLARIS for downstream business objectives.

Transfer Ratio: SOLARIS achieves a transfer ratio of 42% on Instagram tasks and 44% on Facebook products, representing a 2X increase over previous distillation-based approaches. This shows the effectiveness of SOLARIS’s unique knowledge sharing.

4.3 SOLARIS Embedding Coverage Results

Feature Coverage VS Performance: As described in Sec 3.3, SOLARIS adopts asynchronous pre-computation to generate user-ad embeddings during serving time. We deployed SOLARIS in production environment and user-ad embedding shows coverage of 40%. The coverage is calculated on the training data where the labels (click, conversion, etc) are generated. We conducted the offline experiment on user-ad embedding coverage’s effect on downstream models’ performance boost. The offline user-ad embeddings are

generated by the same SOLARIS model but during offline training and evaluation so that the coverage achieves nearly 100%. We down-sample the user-ad embeddings to achieve different coverages and add to the downstream model. As shown in Table 2, when the feature coverage increases by 30%, the log loss is improved by 0.05% which is considered significant on our production models. This shows great potential for unblocking more ads revenue when the SOLARIS feature coverage increases.

Aggregated User-ad Performance: As described in Sec 3.4, we introduce an aggregated version of user-ad embedding as user-only embedding, leading to the coverage increases to 90%. During production deployment, we deploy user-only embedding as a separate feature to achieve 0.03% additional BCE loss improvement on Instagram CTR tasks. This user-only feature has been deployed to 10 downstream models in real-world production pipelines and increases ads revenue by 0.07%.

Similarity-based Embedding Performance: We deployed an KNN operator to obtain 100 similar neighbors for each user and impute the missing <user, ad> embedding from the most similar neighbor’s user-ad embedding, leading to the coverage of 70% compared to 40% before. Offline experiments on Instagram CTR task shows the BCE loss improving of 0.02%. Compared to our feature coverage study, this achieves 40% of the headroom (0.02%/0.05%). This is as expected since the imputed user-ad embeddings from neighbors have lower quality.

5 Discussion and Limitation

Further Enhancing Coverage: The current clustering method for improving coverage yields only limited performance gains on vertical models. Our ongoing efforts will expand beyond the U2U (user-to-user) approach described in this paper to also explore A2A (ad-to-ad) and hybrid strategies.

Early-stage Roll Out: SOLARIS is only operational and beneficial for vertical models in the final ranking stage. This limitation arises because earlier ranking stages involve 100X more ads, making coverage and resource constraints significant challenges. Given the vast model space in these early stages, we are actively working to increase embedding coverage, aiming to unlock greater revenue potential.

Limitations: While we aim to share broadly applicable deployment insights, some findings are inherently company-specific. For instance, business requirements and partner relationships at Meta shape the problem spaces addressed by SOLARIS, which may limit the generalizability of certain results.

References

- [1] Jimmy Ba and Rich Caruana. 2014. Do deep nets really need to be deep? *Advances in neural information processing systems* 27 (2014).
- [2] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. 2017. Learning efficient object detection models with knowledge distillation. *Advances in neural information processing systems* 30 (2017).
- [3] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishikesh Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ipsir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*. 7–10.
- [4] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishikesh Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ipsir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*. 7–10.

- [5] 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.
- [6] Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. 2018. Born again neural networks. In *International conference on machine learning*. PMLR, 1607–1616.
- [7] Demi Guo, Alexander M Rush, and Yoon Kim. 2021. Parameter-efficient transfer learning with diff pruning. In *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (volume 1: Long papers)*. 4884–4896.
- [8] Jiafeng Guo, Yinqiong Cai, Yixing Fan, Fei Sun, Ruqing Zhang, and Xueqi Cheng. 2022. Semantic models for the first-stage retrieval: A comprehensive review. *ACM Transactions on Information Systems (TOIS)* 40, 4 (2022), 1–42.
- [9] Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. 2021. Towards a unified view of parameter-efficient transfer learning. *arXiv preprint arXiv:2110.04366* (2021).
- [10] Xinran He, Junfeng Pan, Ou Jin, Tianbing Xu, Bo Liu, Tao Xu, Yanxin Shi, Antoine Atallah, Ralf Herbrich, Stuart Bowers, et al. 2014. Practical lessons from predicting clicks on ads at facebook. In *Proceedings of the eighth international workshop on data mining for online advertising*. 1–9.
- [11] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015).
- [12] Yupeng Hou, Shanlei Mu, Wayne Xin Zhao, Yaliang Li, Bolin Ding, and Ji-Rong Wen. 2022. Towards universal sequence representation learning for recommender systems. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*. 585–593.
- [13] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *International conference on machine learning*. PMLR, 2790–2799.
- [14] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. 2333–2338.
- [15] Muhammad Bilal Islam, Usman Habib, and Muhammad Usman. 2024. Knowledge Transfer In Cross Domain Recommender Systems Using Deep Latent Embeddings. In *2024 International Conference on IT and Industrial Technologies (ICIT)*. IEEE, 1–6.
- [16] Syed Irteza Hussain Jafri, Rozaida Ghazali, Irfan Javid, Zahid Mahmood, and Abdullahi Abdi Abubakar Hassan. 2022. Deep transfer learning with multimodal embedding to tackle cold-start and sparsity issues in recommendation system. *Plos one* 17, 8 (2022), e0273486.
- [17] Yoon Kim and Alexander M Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of the 2016 conference on empirical methods in natural language processing*. 1317–1327.
- [18] Yaniv Leviathan, Matan Kalman, and Yossi Matias. 2023. Fast Inference from Transformers via Speculative Decoding. *arXiv:2211.17192 [cs.LG]* <https://arxiv.org/abs/2211.17192>
- [19] Lei Li, Dingding Wang, Tao Li, Daniel Knox, and Balaji Padmanabhan. 2011. Scene: a scalable two-stage personalized news recommendation system. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. 125–134.
- [20] Mingfu Liang, Xi Liu, Rong Jin, Boyang Liu, Qiuling Suo, Qinghai Zhou, Song Zhou, Laming Chen, Hua Zheng, Zhiyuan Li, Shali Jiang, Jijian Yang, Xiaozhen Xia, Fan Yang, Yasmine Badr, Ellie Wen, Shuyu Xu, Hansey Chen, Zhengyu Zhang, Jade Nie, Chunzhi Yang, Zhichen Zeng, Weilin Zhang, Xingliang Huang, Qianru Li, Shiquan Wang, Evelyn Lyu, Wenjing Lu, Rui Zhang, Wenjun Wang, Jason Rudy, Mengyue Hang, Kai Wang, Yinbin Ma, Shuaiwen Wang, Sihan Zeng, Tongyi Tang, Xiaohan Wei, Longhao Jin, Jamey Zhang, Marcus Chen, Jiayi Xu, Angie Huang, Xihuan Zeng, Chi Zhang, Zhengli Zhao, Jared Yang, Qiang Jin, Xian Chen, Amit Anand Amlasahwaram, Lexi Song, Liang Luo, Yuchen Hao, Nan Xiao, Yavuz Yetim, Luoshang Pan, Gaoxiang Liu, Yuxi Hu, Yuzhen Huang, Jackie Xu, Rich Zhu, Xin Zhang, Yiqun Liu, Hang Yin, Yuxin Chen, Buyun Zhang, Xiaoyi Liu, Xingyuan Wang, Wenguang Mao, Zhijing Li, Zhehui Zhou, Feifan Gu, Qin Huang, Chonglin Sun, Nancy Yu, Shuo Gu, Shupin Mao, Benjamin Au, Jingzheng Qin, Peggy Yao, Jae-Woo Choi, Bin Gao, Ernest Wang, Lei Zhang, Wen-Yen Chen, Ted Lee, Yujie Zha, Yi Meng, Alex Gong, Edison Gao, Jack Hsueh, Jie Zheng, Alireza Vahdatpour, Yiping Han, Yantao Yao, Toshinari Kureha, Shuo Chang, Musharaf Sultan, John Bocharov, Sagar Chordia, Xiaorui Gan, Peng Sun, Rocky Liu, Bo Long, Wenlin Chen, Santanu Kolay, and Huayu Li. 2025. External Large Foundation Model: How to Efficiently Serve Trillions of Parameters for Online Ads Recommendation. *arXiv:2502.17494 [cs.LR]* <https://arxiv.org/abs/2502.17494>
- [21] Liang Luo, Yuxin Chen, Zhengyu Zhang, Mengyue Hang, Andrew Gu, Buyun Zhang, Boyang Liu, Chen Chen, Chengze Fan, Dong Liang, Fan Yang, Feifan Gu, Huayu Li, Jade Nie, Jiayi Xu, Jijian Yang, Jongsoo Park, Laming Chen, Longhao Jin, Qianru Li, Qin Huang, Shali Jiang, Shiwen Shen, Shuaiwen Wang, Sihan Zeng, Siyang Yuan, Tongyi Tang, Weilin Zhang, Wenjun Wang, Xi Liu, Xiaohan Wei, Xiaozhen Xia, Yuchen Hao, Yunlong He, Yasmine Badr, Zeliang Chen, Maxim Naumov, Yantao Yao, Wenlin Chen, Santanu Kolay, GP Musumeci, and Ellie Dingqiao Wen. 2025. Meta Lattice: Model Space Redesign for Cost-Effective Industry-Scale Ads Recommendations. *arXiv:2512.09200 [cs.LR]* <https://arxiv.org/abs/2512.09200>
- [22] H Brendan McMahan, Gary Holt, David Sculley, Michael Young, Dietmar Ebner, Julian Grady, Lan Nie, Todd Phillips, Eugene Davydov, Daniel Golovin, et al. 2013. Ad click prediction: a view from the trenches. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1222–1230.
- [23] H Brendan McMahan, Gary Holt, David Sculley, Michael Young, Dietmar Ebner, Julian Grady, Lan Nie, Todd Phillips, Eugene Davydov, Daniel Golovin, et al. 2013. Ad click prediction: a view from the trenches. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1222–1230.
- [24] Ravi Teja Mullapudi, Steven Chen, Keyi Zhang, Deva Ramanan, and Kayvon Fatahalian. 2019. Online model distillation for efficient video inference. In *Proceedings of the IEEE/CVF International conference on computer vision*. 3573–3582.
- [25] Maxim Naumov, Dheevatsa Mudigere, Hao-Jun Michael Shi, Jianyu Huang, Narayanan Sundaraman, Jongsoo Park, Xiaodong Wang, Udit Gupta, Carole-Jean Wu, Allison G Azzolini, et al. 2019. Deep learning recommendation model for personalization and recommendation systems. *arXiv preprint arXiv:1906.00091* (2019).
- [26] Li-Wei Pan, Wei-Ke Pan, Mei-Yan Wei, Hong-Zhi Yin, and Zhong Ming. 2026. A survey on sequential recommendation. *Frontiers of Computer Science* 20, 3 (2026), 2003606.
- [27] Lucas Pinheiro Cinelli, Matheus Araújo Marins, Eduardo Antônio Barros da Silva, and Sérgio Lima Netto. 2021. Variational autoencoder. In *Variational methods for machine learning with applications to deep networks*. Springer, 111–149.
- [28] Bongun Shin, Hao Yang, and Jinho D Choi. 2019. The pupil has become the master: teacher-student model-based word embedding distillation with ensemble learning. *arXiv preprint arXiv:1906.00095* (2019).
- [29] Jiakai Tang, Sunhao Dai, Teng Shi, Jun Xu, Xu Chen, Wen Chen, Jian Wu, and Yuning Jiang. 2025. Think before recommend: Unleashing the latent reasoning power for sequential recommendation. *arXiv preprint arXiv:2503.22675* (2025).
- [30] Jiayi Tang and Ke Wang. 2018. Ranking distillation: Learning compact ranking models with high performance for recommender system. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 2289–2298.
- [31] Ruoxi Wang, Rakesh Shivanna, Derek Cheng, Sagar Jain, Dong Lin, Lichan Hong, and Ed Chi. 2021. Dcn v2: Improved deep & cross network and practical lessons for web-scale learning to rank systems. In *Proceedings of the web conference 2021*. 1785–1797.
- [32] Shoujin Wang, Qi Zhang, Liang Hu, Xiuzhen Zhang, Yan Wang, and Charu Aggarwal. 2022. Sequential/session-based recommendations: Challenges, approaches, applications and opportunities. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*. 3425–3428.
- [33] Biqing Zeng, Hao Shi, Yangyu Li, and Huimin Deng. 2025. TARAREc: Two-Stage Augmented Retrieval and Alignment for Recommendation Leveraging Large Language Models. In *International Conference on Advanced Data Mining and Applications*. Springer, 336–349.
- [34] Zhichen Zeng, Xiaolong Liu, Mengyue Hang, Xiaoyi Liu, Qinghai Zhou, Chaofei Yang, Yiqun Liu, Yichen Ruan, Laming Chen, Yuxin Chen, Yujia Hao, Jiaqi Xu, Jade Nie, Xi Liu, Buyun Zhang, Wei Wen, Siyang Yuan, Hang Yin, Xin Zhang, Kai Wang, Wen-Yen Chen, Yiping Han, Huayu Li, Chunzhi Yang, Bo Long, Philip S. Yu, Hanghang Tong, and Jijian Yang. 2025. InterFormer: Effective Heterogeneous Interaction Learning for Click-Through Rate Prediction. *arXiv:2411.09852 [cs.LR]* <https://arxiv.org/abs/2411.09852>
- [35] Junhai Zhai, Sufang Zhang, Junfen Chen, and Qiang He. 2018. Autoencoder and its various variants. In *2018 IEEE international conference on systems, man, and cybernetics (SMC)*. IEEE, 415–419.
- [36] Buyun Zhang, Liang Luo, Yuxin Chen, Jade Nie, Xi Liu, Daifeng Guo, Yanli Zhao, Shen Li, Yuchen Hao, Yantao Yao, Guna Lakshminarayanan, Ellie Dingqiao Wen, Jongsoo Park, Maxim Naumov, and Wenlin Chen. 2024. Wukong: Towards a Scaling Law for Large-Scale Recommendation. *arXiv:2403.02545 [cs.LG]* <https://arxiv.org/abs/2403.02545>
- [37] Buyun Zhang, Liang Luo, Xi Liu, Jay Li, Zeliang Chen, Weilin Zhang, Xiaohan Wei, Yuchen Hao, Michael Tsang, Wenjun Wang, et al. 2022. DHEN: A deep and hierarchical ensemble network for large-scale click-through rate prediction. *arXiv preprint arXiv:2203.11014* (2022).
- [38] Yongfeng Zhang, Qingyao Ai, Xu Chen, and Pengfei Wang. 2018. Learning over knowledge-base embeddings for recommendation. *arXiv preprint arXiv:1803.06540* (2018).
- [39] Zhe Zhao, Lichan Hong, Li Wei, Jilin Chen, Aniruddh Nath, Shawn Andrews, Aditee Kuntekar, Maheswaran Sathiamoorthy, Xinyang Yi, and Ed Chi. 2019. Recommending what video to watch next: a multitask ranking system. In *Proceedings of the 13th ACM conference on recommender systems*. 43–51.

Zikun Liu, Liang Luo, Qianru Li, Zhengyu Zhang, Wei Ling, Jingyi Shen, Zeliang Chen, Yanning Huang, Jingxian Huang, Abdallah Aboelela, Chonglin Sun, Feifan Gu, Fenggang Wu, Hang Qu, Huayu Li, Jill Pan, Kaidi Pei, Laming Chen, Longhao Jin, Qin Huang, Tongyi Tang, Varna Puvvada, Wenlin Chen, Xiaohan Wei, Xu Cao, Yantao Yao, Yuan Jin, Yunchen Pu, Yuxin Chen, Zijian Shen, Zhengkai Zhang, Jing Zhu, Dong Liang, Ellie Wen

[40] Zhe Zhao, Lichan Hong, Li Wei, Jilin Chen, Aniruddh Nath, Shawn Andrews, Aditee Kumthekar, Maheswaran Sathiamoorthy, Xinyang Yi, and Ed Chi. 2019.

Recommending what video to watch next: a multitask ranking system. In *Proceedings of the 13th ACM conference on recommender systems*. 43–51.