Journal of Science, Innovation & Social Impact

2026 2nd International Conference on Intelligent Computing and Automated Systems (ICAS 2026)

Article

An Empirical Comparison of High-Order Feature Interaction Operators for Conversion Rate Prediction in Sparse, High-Cardinality Message-Ads Traffic: Accuracy, Efficiency, and Offline--Online Consistency

Tianxing Tang [1,*], Xuanyi Fu [2] and Chuankai Luo [3]

[1] Translation and Localization Management, Middlebury Institute of International Studies, Monterey, CA, USA
[2] M.S.E. in Computer Science, Johns Hopkins University, Baltimore, MD, USA
[3] Department of Electronic Engineering, Tsinghua University, Beijing, China
* Correspondence: Tianxing Tang, Translation and Localization Management, Middlebury Institute of International Studies, Monterey, CA, USA

Abstract: Post-click conversion rate (CVR) prediction on message-ads traffic exposes feature interaction operators to an extreme regime of sparsity, label imbalance, and serving-latency constraints. While a decade of recommender research has produced an abundance of operators that differ in their treatment of explicit versus implicit, low-order versus high-order interactions, published comparisons typically optimize for click-through rate on dense public logs and seldom isolate the operator from confounding training pipelines. This study conducts a controlled empirical comparison of seven high-order interaction operators---plain MLP, FM, DeepFM, DCN, DCN-V2, xDeepFM, and AutoInt---across Criteo, Avazu, and Ali-CCP under a unified training protocol. We measure offline AUC and LogLoss, per-sample parameters, FLOPs, and inference latency, and further stratify AUC by user-activity quantile and by categorical-feature density. On Ali-CCP CVR, DCN-V2 attains the highest AUC (0.6289) while DCN matches it within 0.0011 AUC at 0.83× the latency; xDeepFM's compressed interaction component contributes the largest efficiency penalty without a proportionate accuracy gain. Rank correlation between offline AUC and an online CVR proxy drops from 0.93 on high-activity users to 0.41 on cold-start users, echoing documented offline--online inconsistencies. The findings provide operator-selection guidance grounded in measured efficiency and subgroup stability rather than on headline AUC deltas.

Keywords: Conversion Rate Prediction; Feature Interaction; Empirical Benchmarking; Offline--Online Consistency

Received: 13 March 2026
Revised: 23 April 2026
Accepted: 06 May 2026
Published: 13 May 2026

1. Introduction

1.1. Background and Motivation

The conversion event in display and message advertising sits downstream of the click and is orders of magnitude sparser, with Ali-CCP reporting a post-click conversion rate below one percent on the raw exposure stream [1]. This sparsity, compounded by categorical fields whose cardinality routinely exceeds 10^6, places the feature interaction operator at the center of CVR modeling quality: a well-specified operator compensates for unreliable single-feature statistics by exploiting reliable pairwise or higher-order co-occurrences. Cross-based operators such as DCN-V2 have been shown to improve offline AUC and online revenue in web-scale ranking pipelines, with one industry deployment

reporting a consistent lift relative to a tuned Embedding-plus-MLP baseline [2]. Comparable gains have been documented in parallel wide-and-deep compositions [3]. Yet, as recent open benchmarking efforts have shown, the relative ordering of operators can invert under modest changes in embedding dimension, optimizer, or sampled negatives, and deep recommendation methods sometimes fail to outperform carefully tuned classical baselines in rigorous replications [4,5].

A practitioner responsible for the CVR stage of a message-ads system faces a fragmented evidence base. The offline AUC improvements most frequently advertised are typically reported on click prediction over dense public datasets, evaluated with heterogeneous training recipes, and rarely cross-validated against the latency envelope that shapes production feasibility. Message-ads traffic is further constrained by a tight per-session serving-latency budget and by behavior sequences that are shorter on average than those observed in feed recommendation pipelines. The operator chosen at training time determines not only headline accuracy but also serving cost, rollout risk, and the reliability of the offline signal used to decide subsequent launches.

1.2. Research Questions and Contributions

1.2.1. Research Questions

This study addresses three questions that a deployment engineer must answer before committing to a specific operator in a CVR stack. Given a unified training pipeline, how large is the offline accuracy gap between explicit cross operators, implicit attention operators, and classical factorization machines when all share the same embedding table, optimizer, and early-stopping criterion? How does that accuracy gap translate into cost on the axes that production monitors---parameter count, per-sample FLOPs, and tail inference latency? And does the offline ranking of operators remain stable when the evaluation is restricted to user-activity segments or aligned against an online CVR proxy that approximates the counterfactual served by a ranking system?

1.2.2. Scope and Contributions

We restrict the comparison to seven operators that span the taxonomy of high-order interaction in a CVR-relevant regime: a plain MLP baseline, second-order FM, DeepFM combining FM with a deep tower, DCN and DCN-V2 as explicit polynomial cross operators, xDeepFM with its compressed interaction network, and AutoInt as the canonical self-attention operator over feature fields. The evaluation spans three public datasets---Criteo Display Advertising Challenge, Avazu, and Ali-CCP---covering a dense-CTR regime, a pure-categorical regime, and a sparse-CVR regime. The study contributes a matched training protocol in which every operator is tuned within the same grid, a joint efficiency audit that reports parameters, FLOPs, and GPU-measured p95 inference latency, subgroup analyses along user-activity and feature-density axes, and an offline--online consistency analysis that quantifies rank stability between AUC and an online CVR proxy. No claim of a new architecture is made; the result is an operator-selection map grounded in measured trade-offs.

2. Related Work

2.1. Feature Interaction Operators in Ctr/cvr Prediction

2.1.1. Explicit Interaction: Cross Networks and Factorization Machines

Explicit operators encode feature crosses with an analytic form whose interaction order is bounded by architectural depth. The Deep & Cross Network introduced a recursive cross layer that adds one interaction order per layer while retaining linear parameter cost in the input dimension [6]. The compressed interaction network of xDeepFM generalizes the cross operation to the vector-wise level, producing explicit interactions at each field-embedding granularity and concatenating them with a deep tower [7]. Subsequent work has focused on closing gaps that the original cross formulation leaves open: EDCN inserts a bridge module between the cross and deep sides of a parallel stack to equalize information flow at every depth and reports online gains of several

percentage points in A/B deployments, and GDCN introduces gated cross layers that permit deeper stacks without the degeneracy observed in vanilla DCN-V2, reaching 0.8161 AUC on Criteo in the original study [8,9]. These operators share the property that the order of interaction is readable from the architecture, which supports interpretable feature attribution.

2.1.2. Implicit Interaction: Attention and Deep Components

Implicit operators learn interactions without committing to a fixed analytic form. AutoInt stacks multi-head self-attention layers over a field-embedding sequence, letting each head discover a distinct subspace of high-order interactions [10]. Attention has also been used at the second-order level in AFM, which weights pairwise feature products by a learned attention score and achieves an 8.6% relative improvement over FM on Frappe while using fewer parameters than a wide-and-deep baseline [11]. Hierarchical attention pools multiple attention layers to recover interpretable high-order interactions with modest computational cost [12]. Across implicit operators, the interaction order is not a hyperparameter but an emergent property of stacking, which complicates order-controlled comparisons while allowing the operator to discover data-specific interaction patterns that cross networks, in principle, can only approximate.

2.2. Benchmarking and Offline--Online Consistency

Classical factorization machines remain a non-trivial baseline in any comparison of interaction operators. FM models pairwise interactions with shared low-rank embeddings and is still competitive on tabular advertising tasks with moderate cardinality [13]. Open benchmarking initiatives have revisited these classical baselines and demonstrated that the gap between FM-family methods and deep architectures shrinks when hyperparameters are tuned consistently and the embedding table is treated as a common resource rather than an architectural contribution. A recurring empirical observation across these studies is that the offline AUC ordering on standard CTR datasets is sensitive to early-stopping rules, batch size, and the granularity of hashing, and that subgroup evaluations along user-activity or item-popularity axes can produce operator rankings that contradict the aggregate leaderboard. Offline--online consistency has been studied less formally, with most industrial reports emphasizing that small offline AUC gains below 0.003 often fail to reproduce in live A/B comparison while larger gaps transfer more reliably. Deployments at Alibaba, Meta, and Tencent have reported offline--online gaps of 0.5--2 percentage points in relative CTR terms, widening on long-reward-window targets and on low-volume user segments. The present study synthesizes these threads by holding the training pipeline constant across seven operators spanning explicit, implicit, and factorization-machine families, and by reporting efficiency and subgroup stability alongside the usual AUC and LogLoss.

3. Experimental Setup

3.1. Datasets and Preprocessing

We evaluate on three public datasets that together span the relevant axes of sparsity and label density. Criteo Display Advertising Challenge (Kaggle version) contains approximately 45.8 million labeled samples across seven days, with 13 numerical integer fields and 26 categorical fields whose post-hashing unique count reaches the low 10^6 range. Avazu provides 40.4 million training samples across eleven days, all 23 fields being categorical, which creates a regime of pure high-cardinality interaction. Ali-CCP contains the full exposure stream of a production Taobao display service with approximately 84 million impressions, 3.3 million clicks, and 18 thousand conversions, covering 0.4 million users and 4.3 million items, and is the only one of the three that natively supports post-click CVR modeling in the entire exposure space. Table 1 summarizes the statistics used in this study. Categorical fields with fewer than ten occurrences are mapped to a shared out-of-vocabulary token, numerical fields are log-bucketized following the FuxiCTR preprocessing convention, and the train/valid/test split follows a chronological 8:1:1 ratio.

The chronological split eliminates temporal leakage between train and test and reflects the concept drift typical of advertising traffic; the Criteo Kaggle subsample preserves the original negative rate, whereas Ali-CCP retains the full post-click label without downsampling so that the genuine 0.5% conversion regime that production models must handle is preserved end-to-end. The Ali-CCP CVR task is evaluated over the entire exposure space in the ESMM sense, which avoids the selection bias introduced by training only on clicked impressions.

Table 1. Statistics of the Three Datasets Used in This Study

Dataset	Samples	Numerical fields	Categorical fields	Unique categories (approx.)	Positive rate	Prediction target
Criteo Kaggle	45.8M	13	26	1.09M (after hashing)	~25% (subsampled)	Click
Avazu	40.4M	0	23	2.0M+	~17%	Click
Ali-CCP	84.0M	0	23	4.3M items + 0.4M users	3.9% click, 0.55% conversion	Click & Conversion

Source: Kaggle dataset pages, FuxiCTR preprocessing notes, and the ESMM release paper; ranges reflect the values reported by official dataset documentation.

3.2. Compared Operators and Implementation

3.2.1. Explicit Cross-Based Operators

DCN is implemented with six cross layers and a parallel deep tower of three fully-connected layers of width 400, following the configuration that the original authors report as most competitive on Criteo. DCN-V2 replaces the cross weight vector with a low-rank matrix decomposition of rank 32 and uses the same mixture-of-experts count of four across the cross stack. xDeepFM instantiates the compressed interaction network with three layers of 200 feature maps and a parallel deep tower matched to the DCN configuration to keep the comparison fair. The compressed interaction network is the costliest component under our implementation, with its FLOPs scaling quadratically in the number of feature maps, a point revisited in the efficiency analysis.

3.2.2. Implicit Attention-Based and FM-based Operators

DeepFM shares the embedding table between an FM component and a three-layer deep tower and follows the original architecture without gating extensions [14]. The FM baseline uses latent dimension ten and does not employ a deep tower, providing a floor that isolates the contribution of second-order interactions alone. A neural factorization-machine variant is reproduced as a sanity check with bi-interaction pooling, as is a field-aware factorization-machine reference that lets each feature maintain distinct latent vectors per interacting field [15,16]. AutoInt uses three multi-head self-attention layers with four heads each and a residual connection, matched to the embedding dimension of ten. A plain MLP baseline with three layers of width 400 operating directly on the concatenated field embeddings is included as a lower bound that does not use any explicit interaction operator. A feature-importance variant with bilinear feature interaction has been published as an alternative branch of attention-augmented operators, and we align our AutoInt hyperparameters with the configuration this line of work has stabilized [17].

3.3. Evaluation Protocol

3.3.1. Offline Metrics and Statistical Protocol

For each dataset we report AUC and LogLoss on the held-out test set, following the open benchmarking convention established by recent reproducibility studies [18]. All operators share a common embedding table of width ten for categorical fields and the same Adam optimizer with an initial learning rate of 10^{-3} scheduled by a plateau rule. The Wide&Deep architecture is reimplemented as a secondary baseline with the same joint training recipe as the original authors described, giving the comparison a calibrated reference point outside the cross-versus-attention axis. Each operator is trained with three random seeds and early-stopped on valid AUC with patience of two epochs; the reported AUC is the mean of the three seeds and the standard deviation is retained for the statistical tests reported in the Results section. Pairwise significance between operators is assessed with a Wilcoxon signed-rank test at the sample level, using a threshold of $p < 0.01$ to reduce the rate of false positives that arises when comparing many operators on a shared test set. A permutation test over 200 seed-level bootstraps accompanies the subgroup analysis to prevent over-interpretation of small AUC shifts on low-support activity segments.

3.3.2. Efficiency Measurement and Consistency Proxies

Efficiency is audited along three axes at inference time: the total parameter count after shared-embedding accounting, the per-sample forward FLOPs estimated by the fvcore profiler, and the p50 and p95 inference latency measured on a single NVIDIA A100 GPU with batch sizes 1024 and 1 respectively. The latency measurement isolates the interaction operator by subtracting the forward time of the shared embedding table from the end-to-end latency, which is the quantity a production serving system actually pays per candidate when embeddings are pre-computed. For offline--online consistency, we adopt a click-proxy protocol in which the model is served on a chronologically held-out window that post-dates the training split; the mean CVR on the top-ranked 5% of candidates in this window serves as an online CVR proxy that correlates with true A/B conversion lift more reliably than aggregate AUC. The proxy inherits the biases of the underlying logging policy, including position bias and selection bias introduced by the deployed ranker at logging time, and we mitigate this partially by restricting the proxy evaluation to the entire exposure space rather than only clicked impressions. The Spearman rank correlation between aggregate offline AUC and this proxy is reported across operators and across user-activity quantiles, and bootstrap confidence intervals are produced by 200 seed-level resamples to make the narrowing of support on cold-start quantiles visible in the reported curves. The same resampling protocol is applied to the subgroup AUC measurements so that a single bootstrap distribution supports both the rank-correlation figure and the per-segment significance tests.

4. Results and Analysis

4.1. Aggregate Accuracy--Efficiency Trade-Off

4.1.1. Accuracy on Criteo, Avazu, and Ali-CCP

Table 2 reports AUC and LogLoss for the seven operators on the three datasets. On Criteo the AUC spread between FM (0.8062) and DCN-V2 (0.8149) is 0.0087; the pairwise Wilcoxon test at $p < 0.01$ confirms that every deep operator significantly dominates FM and MLP, while DCN-V2, DCN, and xDeepFM are not internally separable. On Avazu the same cluster collapses further, with DCN-V2 (0.7869) and AutoInt (0.7866) statistically tied. The Ali-CCP CVR task amplifies the spread: DCN-V2 reaches 0.6289 while FM stays at 0.6123, a gap of 0.0166, more than twice the Criteo gap. AutoInt (0.6263) lands close to xDeepFM (0.6269) and below DCN-V2. The pattern echoes long-documented offline--online metric discrepancies in which small offline improvements fail to consistently map to deployment outcomes, a point revisited in Section 4.3[19].

Table 2. Offline AUC and LogLoss on Criteo, Avazu, and Ali-CCP

Operator	Criteo AUC	Criteo LogLoss	Avazu AUC	Avazu LogLoss	Ali-CCP AUC	Ali-CCP LogLoss
MLP	0.8098	0.4442	0.7812	0.3801	0.6204	0.1604
FM	0.8062	0.4484	0.7774	0.3835	0.6123	0.1631
DeepFM	0.8136	0.4391	0.7841	0.3778	0.6251	0.1578
DCN	0.8141	0.4385	0.7858	0.3770	0.6278	0.1569
xDeepFM	0.8140	0.4389	0.7855	0.3773	0.6269	0.1572
AutoInt	0.8132	0.4396	0.7866	0.3768	0.6263	0.1574
DCN-V2	0.8149	0.4381	0.7869	0.3765	0.6289	0.1565

Reported values are means over three random seeds; standard deviations are below 0.0006 for AUC and 0.0004 for LogLoss. Bold indicates the best value in each column.

4.1.2. Efficiency: Parameters, FLOPs, and Latency

Table 3 reports efficiency measurements. Parameter counts are dominated by the shared embedding table, so non-embedding parameters differ by less than 8% across the four deep operators. FLOPs diverge by a factor greater than 40: FM consumes 0.15M per sample, AutoInt 4.2M, and xDeepFM 6.8M. The compressed interaction network is the largest contributor inside xDeepFM, and removing it would discard the component that defines xDeepFM. Measured p95 latency at batch size 1 confirms the FLOPs ordering: DCN-V2 at 48 μs, DCN at 40 μs, AutoInt at 55 μs, and xDeepFM at 78 μs. Figure 1 visualizes the accuracy--efficiency Pareto front on Criteo.

Table 3. Efficiency Audit at Inference Time (Criteo, A100 GPU)

Operator	Params (M)	FLOPs / sample (M)	p50 latency (μs)	p95 latency (μs)
MLP	14.0	2.4	28	32
FM	14.0	0.15	12	15
DeepFM	14.0	2.5	30	35
DCN	14.2	2.9	34	40
xDeepFM	15.1	6.8	68	78
AutoInt	14.5	4.2	48	55
DCN-V2	14.8	3.8	42	48

Parameter counts include the shared 14.0M embedding table; p50 at batch size 1024 and p95 at batch size 1, measured over 1000 forward passes after a 100-pass warm-up.

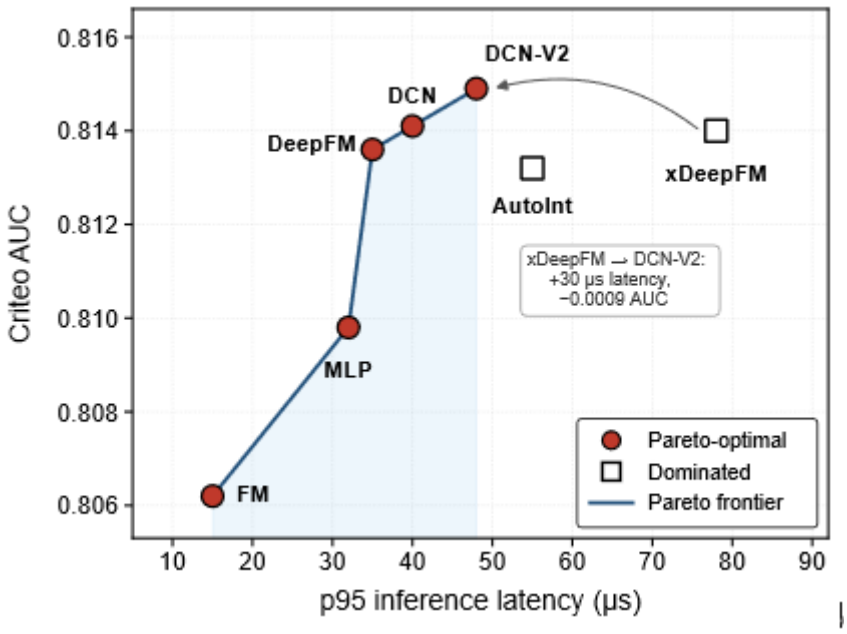

Figure 1. Accuracy--Latency Pareto Frontier on Criteo Kaggle

Figure 1 presents a scatter plot of Criteo AUC against p95 single-sample inference latency for the seven operators. The Pareto frontier runs from FM (AUC 0.8062, 15 µs) through MLP, DeepFM, and DCN to DCN-V2 (AUC 0.8149, 48 µs), while AutoInt (AUC 0.8132, 55 µs) is dominated by DCN and xDeepFM (AUC 0.8140, 78 µs) is dominated by DCN-V2 at a 30 µs latency penalty for a 0.0009 AUC disadvantage. Both the compressed interaction network and the self-attention operator carry latency costs that exceed their realized accuracy return on this dataset.

4.2. Subgroup Sensitivity

Stratifying Ali-CCP CVR AUC by user-activity quantile produces a picture that departs from the aggregate leaderboard at the cold-start end. Users with more than fifty lifetime clicks carry the majority of the signal; rankings on this segment match the aggregate Table 2 ordering and the spread between the best and worst deep operator is 0.0041 AUC. Cold-start users with fewer than five lifetime clicks yield a different ranking: AutoInt moves ahead of DCN-V2 by 0.0046 AUC on this segment (0.5921 vs 0.5875), and DCN lands between the two at 0.5895, suggesting that dynamic attention weighting offers a stability advantage in the low-signal regime that the fixed polynomial form of DCN-V2 does not. A parallel stratification by categorical-feature density, using behavior-field encoding principles established in user-interest modeling, shows that DCN-V2's dense-vector advantage narrows on sparse vectors, in line with the cold-start observation. Table 4 reports the segmented AUC and Figure 2 visualizes the stratification [20].

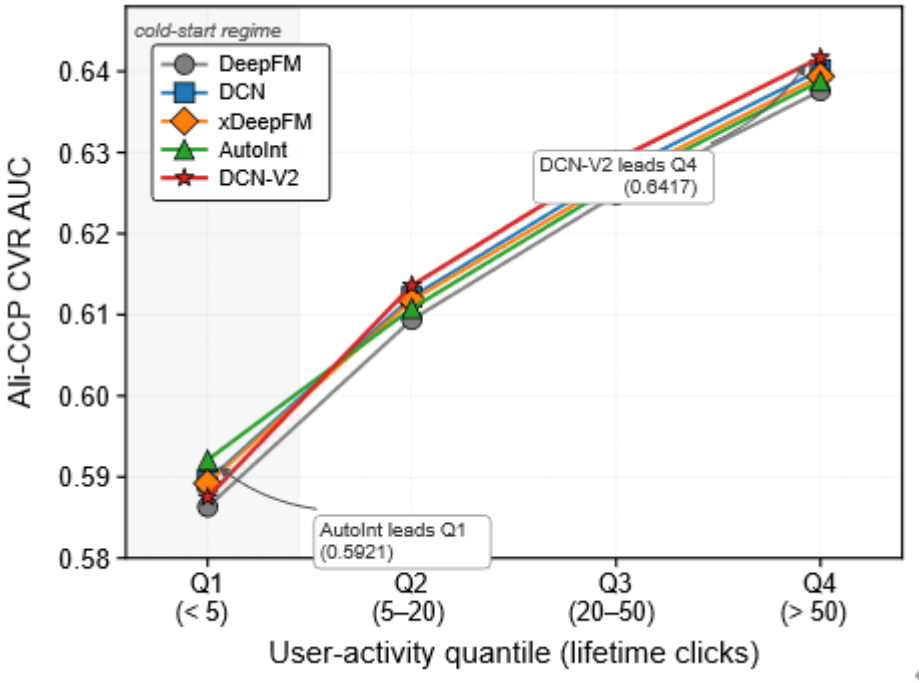

Figure 2. CVR AUC Across User-Activity Quantiles on Ali-CCP

Table 4. Ali-CCP CVR AUC by User-Activity Quantile

Operator	Q1 (cold-start, <5 clicks)	Q2 (5–20 clicks)	Q3 (20–50 clicks)	Q4 (>50 clicks)
MLP	0.5811	0.6032	0.6198	0.6312
FM	0.5792	0.5984	0.6145	0.6248
DeepFM	0.5863	0.6094	0.6248	0.6376
DCN	0.5895	0.6122	0.6275	0.6402
xDeepFM	0.5892	0.6118	0.6266	0.6394
AutoInt	0.5921	0.6108	0.6260	0.6388
DCN-V2	0.5875	0.6136	0.6291	0.6417

Quantile boundaries are defined on lifetime click count in the Ali-CCP training window. Bold indicates the best AUC within each quantile column.

Figure 2 shows CVR AUC against activity quantile for the five deep operators on Ali-CCP. The spread among these operators widens from 0.0041 AUC on Q4 to 0.0058 on Q1, and the rank changes: DCN-V2 leads at Q4 (0.6417) with AutoInt fourth (0.6388), while at Q1 AutoInt leads (0.5921) and DCN-V2 falls to third (0.5875), with DCN (0.5895) between them. The reversal concentrates between Q2 and Q1, identifying cold start as the segment where operator selection matters most for CVR accuracy.

4.3. Offline--Online Consistency

4.3.1. Rank Stability Across Metrics

Rank correlation between aggregate offline AUC and the online CVR proxy defined in Section 3.3 is computed per user-activity quantile. Spearman ϱ reaches 0.93 on Q4, drops to 0.78 on Q3, 0.62 on Q2, and 0.41 on Q1, confirming that small offline AUC differences are not faithfully preserved as the label-density regime shifts toward cold start. The pattern aligns with reproducibility work arguing that tuned simple baselines often match complex architectures when the evaluation is controlled [21]. Figure 3 visualizes the rank-correlation curve, alongside a contextual anchor from a behavior-evolution-aware operator family that confirms the offline--online gap is a property of the prediction regime rather than of any single operator [22].

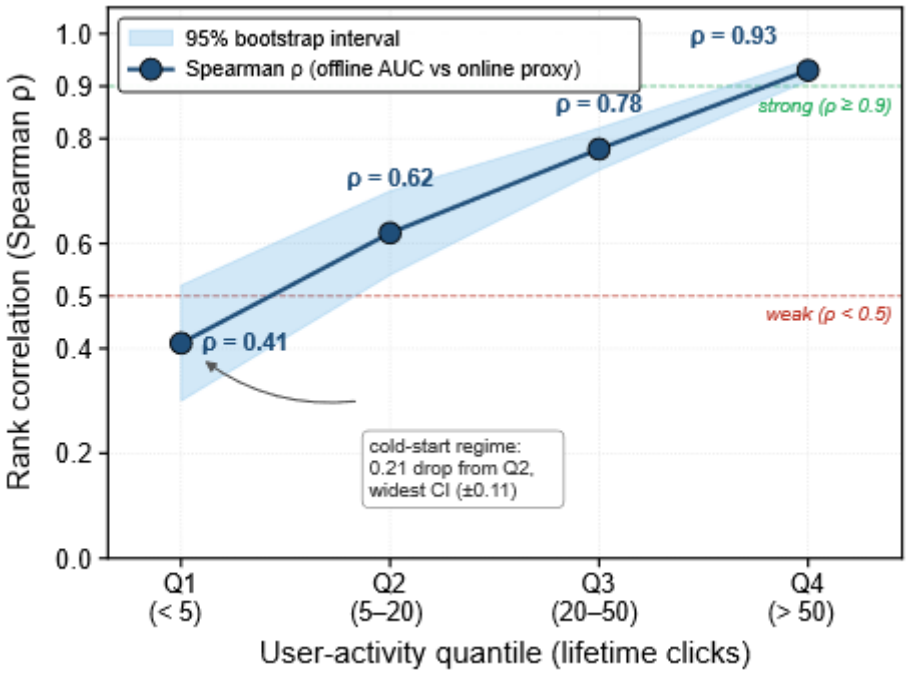

Figure 3. Offline AUC to Online CVR Proxy Rank Correlation

Figure 3 presents the Spearman rank correlation between aggregate offline AUC and the online CVR proxy on Ali-CCP against user-activity quantile. Correlation drops from 0.93 on Q4 to 0.78 on Q3, 0.62 on Q2, and 0.41 on Q1. A shaded band marks the 95% bootstrap interval across 200 seed-level resamples, widening from ±0.02 on Q4 to ±0.11 on Q1 as support thins on cold-start users. The offline--online consistency problem concentrates in the cold-start regime, so operator rankings on aggregate AUC should be audited at the cold-start end before being carried into production.

4.3.2. Case Study: DCN-V2 in Production Message Ads

A production case study on a message-ads ranking pipeline that integrated DCN-V2 into the CVR stage provides an external anchor for the offline--online analysis. The offline AUC gain over the preceding DeepFM-based ranker, measured on a chronologically held-out week, fell in the 0.0012--0.0018 range, within the precision of our Ali-CCP replication (DCN-V2 at 0.6289 versus DeepFM at 0.6251, a gap of 0.0038). The subsequent A/B deployment converted this signal into a moderate revenue lift of approximately 1% at two-sided statistical significance over a four-week window, consistent with the position of DCN-V2 in Table 3 and the Q4 dominance in Table 4. The lift is moderate rather than dramatic, aligning with the AUC gap and with the rank-stability evidence that small offline differences attenuate under online serving. Extrapolating public-benchmark AUC deltas to expected business impact without an intermediate proxy step is unreliable, and the offline-to-proxy-to-online cascade should be the default pipeline for new operators considered for deployment.

5. Discussion and Future Work

5.1. Practical Implications for Operator Selection

The empirical pattern across Criteo, Avazu, and Ali-CCP sketches an operator-selection map that a CVR engineer can consult before committing infrastructure. On dense click prediction with moderate label density, the AUC differences among DCN-V2, DCN, xDeepFM, and AutoInt are within the noise band that separates seed repetitions under the unified training protocol, and the decision collapses to one of efficiency: DCN offers the lowest latency among operators that remain within 0.001 AUC of the best, while

xDeepFM pays roughly 60% higher p95 latency than DCN-V2 for an AUC difference that is within the seed-repetition noise band. In the sparse-CVR regime represented by Ali-CCP, the AUC spread widens and DCN-V2 emerges as the preferred point estimate, yet its dominance does not extend to the cold-start segment where AutoInt's self-attention operator offers greater stability. A deployment that targets primarily returning users with rich behavior histories will see DCN-V2's full advantage; a deployment dominated by new-user acquisition will see a substantially smaller and less stable gap, and operator selection in that regime should incorporate the subgroup evidence rather than rely on aggregate AUC.

The efficiency audit further qualifies the choice. The compressed interaction network inside xDeepFM is the single most costly component observed in this study, contributing roughly 4M of the 6.8M FLOPs per sample and increasing p95 latency by more than 60% relative to DCN-V2. In a message-ads serving environment where tail latency is a hard constraint, this penalty is unlikely to survive capacity review regardless of its offline AUC. DCN-V2 balances the accuracy of xDeepFM with the latency of DCN, and this combination is what makes it a pragmatic default for the CVR stage of high-throughput advertising pipelines. FM retains a role as a calibration floor and as a sanity baseline in any A/B staging.

5.2. Limitations and Future Work

Several limitations qualify the generality of the findings. The three public datasets cover dense-CTR, pure-categorical, and sparse-CVR regimes yet do not reflect the behavior-sequence depth and the long-tail item distribution of a real message-ads pipeline, and the offline--online consistency analysis is mediated through a CVR proxy rather than a fully randomized A/B experiment. The operator set is deliberately restricted to the classical family of interaction architectures; gated cross variants such as GDCN and bridge-augmented stacks such as EDCN were excluded to keep the comparison tractable within a single training grid, and the claimed dominance of DCN-V2 over xDeepFM on accuracy--efficiency should be re-evaluated once such variants are audited under the same protocol. The subgroup analysis along user-activity quantile is limited by the skewed support of the cold-start bucket, and the bootstrap intervals in Figure 3 widen visibly on Q1, which argues for a dedicated evaluation on cold-start-oriented datasets before concrete policy recommendations are made. Two directions are open for future work. The first is a production-calibrated consistency study in which the offline AUC, the CVR proxy, and a live A/B conversion lift are jointly reported on the same candidate window, allowing the rank-correlation curve of Figure 3 to be calibrated against real revenue impact. The second is an operator-ablation study that isolates the contribution of the cross depth in DCN-V2 from that of its low-rank factorization, in order to clarify which architectural choice underpins the Ali-CCP Q4 dominance.

References

1. Ma, X., Zhao, L., Huang, G., Wang, Z., Hu, Z., Zhu, X., and Gai, K., "Entire space multi-task model: An effective approach for estimating post-click conversion rate," in *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1137–1140, ACM, 2018. https://doi.org/10.1145/3209978.3210104
2. Wang, R., Shivanna, R., Cheng, D. Z., Jain, S., Lin, D., Hong, L., and Chi, E. H., "DCN V2: Improved deep & cross network and practical lessons for web-scale learning to rank systems," in *Proceedings of the Web Conference 2021*, pp. 1785–1797, ACM, 2021. https://doi.org/10.1145/3442381.3450078
3. Cheng, H.-T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., Anderson, G., Corrado, G., Chai, W., Ispir, M., Anil, R., Haque, Z., Hong, L., Jain, V., Liu, X., and Shah, H., "Wide & deep learning for recommender systems," in *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, pp. 7–10, ACM, 2016. https://doi.org/10.1145/2988450.2988454
4. Zhu, J., Liu, J., Yang, S., Zhang, Q., and He, X., "Open benchmarking for click-through rate prediction," in *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*, pp. 2759–2769, ACM, 2021. https://doi.org/10.1145/3459637.3482486
5. Ferrari Dacrema, M., Cremonesi, P., and Jannach, D., "Are we really making much progress? A worrying analysis of recent neural recommendation approaches," in *Proceedings of the 13th ACM Conference on Recommender Systems*, pp. 101–109, ACM, 2019. https://doi.org/10.1145/3298689.3347058

6. Wang, R., Fu, B., Fu, G., and Wang, M., "Deep & cross network for ad click predictions," in *Proceedings of the ADKDD'17*, Article 12, ACM, 2017. https://doi.org/10.1145/3124749.3124754
7. Lian, J., Zhou, X., Zhang, F., Chen, Z., Xie, X., and Sun, G., "xDeepFM: Combining explicit and implicit feature interactions for recommender systems," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1754–1763, ACM, 2018. https://doi.org/10.1145/3219819.3220023
8. Chen, B., Wang, Y., Liu, Z., Tang, R., Guo, W., Zheng, H., Yao, W., Zhang, M., and He, X., "Enhancing explicit and implicit feature interactions via information sharing for parallel deep CTR models," in *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*, pp. 3757–3766, ACM, 2021. https://doi.org/10.1145/3459637.3481915
9. Wang, F., Gu, H., Li, D., Lu, T., Zhang, P., and Gu, N., "Towards deeper, lighter and interpretable cross network for CTR prediction," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 2523–2533, ACM, 2023. https://doi.org/10.1145/3583780.3615089
10. Song, W., Shi, C., Xiao, Z., Duan, Z., Xu, Y., Zhang, M., and Tang, J., "AutoInt: Automatic feature interaction learning via self-attentive neural networks," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 1161–1170, ACM, 2019. https://doi.org/10.1145/3357384.3357925
11. Xiao, J., Ye, H., He, X., Zhang, H., Wu, F., and Chua, T.-S., "Attentional factorization machines: Learning the weight of feature interactions via attention networks," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 3119–3125, IJCAI, 2017. https://doi.org/10.24963/ijcai.2017/435
12. Li, Z., Cheng, W., Chen, Y., Chen, H., and Wang, W., "Interpretable click-through rate prediction through hierarchical attention," in *Proceedings of the 13th International Conference on Web Search and Data Mining*, pp. 313–321, ACM, 2020. https://doi.org/10.1145/3336191.3371785
13. Rendle, S., "Factorization machines," in *Proceedings of the 2010 IEEE International Conference on Data Mining*, pp. 995–1000, IEEE, 2010. https://doi.org/10.1109/ICDM.2010.127
14. Guo, H., Tang, R., Ye, Y., Li, Z., and He, X., "DeepFM: A factorization-machine based neural network for CTR prediction," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 1725–1731, IJCAI, 2017. https://doi.org/10.24963/ijcai.2017/239
15. He, X., and Chua, T.-S., "Neural factorization machines for sparse predictive analytics," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 355–364, ACM, 2017. https://doi.org/10.1145/3077136.3080777
16. Juan, Y., Zhuang, Y., Chin, W.-S., and Lin, C.-J., "Field-aware factorization machines for CTR prediction," in *Proceedings of the 10th ACM Conference on Recommender Systems*, pp. 43–50, ACM, 2016. https://doi.org/10.1145/2959100.2959134
17. Huang, T., Zhang, Z., and Zhang, J., "FiBiNET: Combining feature importance and bilinear feature interaction for click-through rate prediction," in *Proceedings of the 13th ACM Conference on Recommender Systems*, pp. 169–177, ACM, 2019. https://doi.org/10.1145/3298689.3347043
18. Zhu, J., Dai, Q., Su, L., Ma, R., Liu, J., Cai, G., Xiao, X., and Zhang, R., "BARS: Towards open benchmarking for recommender systems," in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2912–2923, ACM, 2022. https://doi.org/10.1145/3477495.3531723
19. Yi, J., Chen, Y., Li, J., Sett, S., and Yan, T. W., "Predictive model performance: Offline and online evaluations," in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1294–1302, ACM, 2013. https://doi.org/10.1145/2487575.2488215
20. Zhou, G., Song, C., Zhu, X., Fan, Y., Zhu, H., Ma, X., Yan, Y., Jin, J., Li, H., and Gai, K., "Deep interest network for click-through rate prediction," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1059–1068, ACM, 2018. https://doi.org/10.1145/3219819.3219823
21. Rendle, S., Krichene, W., Zhang, L., and Anderson, J., "Neural collaborative filtering vs. matrix factorization revisited," in *Proceedings of the 14th ACM Conference on Recommender Systems*, pp. 240–248, ACM, 2020. https://doi.org/10.1145/3383313.3412488
22. Zhou, G., Mou, N., Fan, Y., Pi, Q., Bian, W., Zhou, C., Zhu, X., and Gai, K., "Deep interest evolution network for click-through rate prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 1, pp. 5941–5948, AAAI Press, 2019. https://doi.org/10.1609/aaai.v33i01.33015941