MAKING DEPTH MATTER: COMPLEMENTARY REPRESENTATION LEARNING FOR DEEP RECOMMENDER SYSTEMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Making a recommender deeper does not necessarily make it better. We study this problem through the joint design of representation learning and reuse, introducing DeCORA (Depthwise Correlation-Regularized Aggregation), a wrapper that couples token-conditional cross-depth correlation control with depth-prefix aggregation. A training-only objective regularizes source correlations with learned, normalized pair weights, while depth-specific maps shared across sources aggregate the embedding and all available hidden representations, retaining the backbone's interaction operators. Our conditional analysis connects correlation control to coordinate-function overlap and its transport through affine maps, clarifying the roles of target alignment and access after aggregation in making distinct responses useful for prediction. Across four structurally different backbones on Avazu and Criteo, DeCORA improves AUC and Logloss over vanilla in all 40 matched Base–$8\times$ configurations, with a mean absolute AUC gain of 0.00156 at $8\times$. Among the component and parameter-matched variants, only the full method improves mean AUC in all eight backbone–dataset pairs from $8\times$ to $16\times$; representation diagnostics further show that the full method consistently exhibits the geometric patterns motivated by our analysis. The code is available in an anonymous repository [1].

1 INTRODUCTION

Deep recommender systems improve feature-interaction capacity by stacking increasingly expressive modules. WuKong and RankMixer illustrate the potential of scaling these backbones (Zhang et al., 2024; Zhu et al., 2025b). Yet the practical value of depth remains configuration dependent: in our Avazu and Criteo experiments, vanilla models reach their highest AUC between Base and $6\times$ scaling, then decline at larger depths. This raises a design question beyond choosing a more expressive interaction operator: *how should representations at successive depths be organized so that the added computation improves recommendation?*

Depth-connection methods offer several answers. DenseFormer learns to combine preceding representations (Pagliardini et al., 2024); Hyper-Connections and its manifold-constrained variant enrich residual connectivity (Zhu et al., 2025a; Xie et al., 2026); and Attention Residuals make cross-depth aggregation input dependent (Kimi Team et al., 2026). Orthogonal Residual Update controls activation directions by projecting updates away from the current stream (Oh et al., 2025). These approaches address information flow, optimization, and diversity through the connections between layers. We investigate a complementary intervention: jointly shaping the *correlations across depth-indexed representations* and how those representations are reused by later computation.

Our hypothesis is that source reuse and correlation control serve complementary purposes. A *depth prefix* (*prefix* for short) comprises the embedding and hidden representations up to a given depth. Reusing a prefix need not prevent successive layers from learning redundant information, while reducing correlation alone need not make the resulting differences accessible to later computation. This motivates pairing an objective that encourages complementary responses across depth with aggregation that routes earlier sources to later computation. Here, *complementarity* refers to useful

[1] https://anonymous.4open.science/r/MakingDepthMatter-ABCD/

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

differences learned under the recommendation objective, with representation diagnostics used to characterize these differences and predictive performance used to assess their utility.

We realize this principle with DeCORA (Depthwise Correlation-Regularized Aggregation), coupling cross-depth correlation control with depth-prefix aggregation. For sequential sources, changing an earlier representation also changes the inputs to later layers. We therefore study correlation control and reuse jointly, using conditional analysis to interpret their roles and controlled experiments to assess their combination. This perspective builds on redundancy reduction as a view of representation diversity, as explored for augmented views (Zbontar et al., 2021) and recommendation experts (Wang et al., 2025). Our contributions are threefold:

- **Coordinated learning and reuse across depth.** We jointly address how representations differ across depth and how later computation accesses them. A token-conditional objective penalizes cross-depth correlations with learned, normalized pair weights, while depth-prefix aggregation routes sources through depth-specific shared maps. The wrapper retains the backbone's interaction operators; its affine instance requires only a running sum and one map application per depth.
- **An analytical view of representation and access across depth.** We connect correlation control to coordinate-function overlap and characterize its transport through shared affine maps. A fixed-representation linear-readout analysis relates the value of new responses to target alignment and their accessibility after aggregation. This analysis distinguishes representational novelty from predictive utility and motivates complementary diagnostics of source overlap, kernel representational novelty, and aggregate–prefix alignment.
- **Controlled evidence for the paired design.** Across two datasets and four structurally different backbones, DeCORA improves AUC and Logloss over vanilla in all 40 matched configurations from Base to $8\times$ depth. From $8\times$ to $16\times$, only the full method improves mean AUC in all eight backbone–dataset pairs across the component and capacity controls, including aggregation only, loss only, and parameter-matched widening. Comparisons with alternative depth mechanisms further support coordinating correlation control with aggregation.

2 Related Work

Scaling recommendation backbones. DHEN hierarchically combines heterogeneous feature-interaction modules (Zhang et al., 2022); WuKong stacks factorization-machine-based blocks (Zhang et al., 2024); RankMixer uses token mixing and per-token feed-forward networks in a mixer-type architecture (Zhu et al., 2025b; Yu et al., 2022); and the Field-Aware Transformer (FAT) adds field-specific structure to attention (Yan et al., 2026). These methods strengthen feature interactions; we instead organize representations across depth on the original backbones, studying depth-scaling behavior using CTR prediction as the evaluation setting.

Cross-depth connections. DenseFormer reuses prior representations through learned depth averaging (Pagliardini et al., 2024), whereas Attention Residuals (AttnRes) use input-dependent selection (Kimi Team et al., 2026); subsequent works further refine residual-stream routing (Luo et al., 2026; Yu et al., 2026). Hyper-Connections (HC) learn residual routing maps, with mHC imposing manifold constraints on these maps (Zhu et al., 2025a; Xie et al., 2026); Orthogonal Residual Update (ORU) instead controls the geometry of per-example residual updates (Oh et al., 2025). While these methods improve access to earlier representations, they do not necessarily make deeper representations complementary. We regularize cross-example feature correlations across depth pairs and couple this objective with a depth-specific prefix summary, comparing against all four implemented alternatives.

Redundancy reduction and representation diagnostics. Barlow Twins regularizes cross-correlation between augmented views toward the identity (Zbontar et al., 2021); D-MoE reduces correlations between parallel recommendation experts (Wang et al., 2025). Our distinction is the unit of regularization: sequential sources evaluated across examples. We learn softmax-normalized weights over depth pairs and couple this objective to depth-prefix aggregation. Our interpretation links squared cross-covariance to linear-kernel HSIC (Gretton et al., 2005). The kernel diagnostics use normalized linear-kernel comparisons (Kornblith et al., 2019) to distinguish overlap with a preceding kernel span from alignment between an aggregate and its prefix. This complements connection designs that determine how earlier representations remain accessible to subsequent computation.

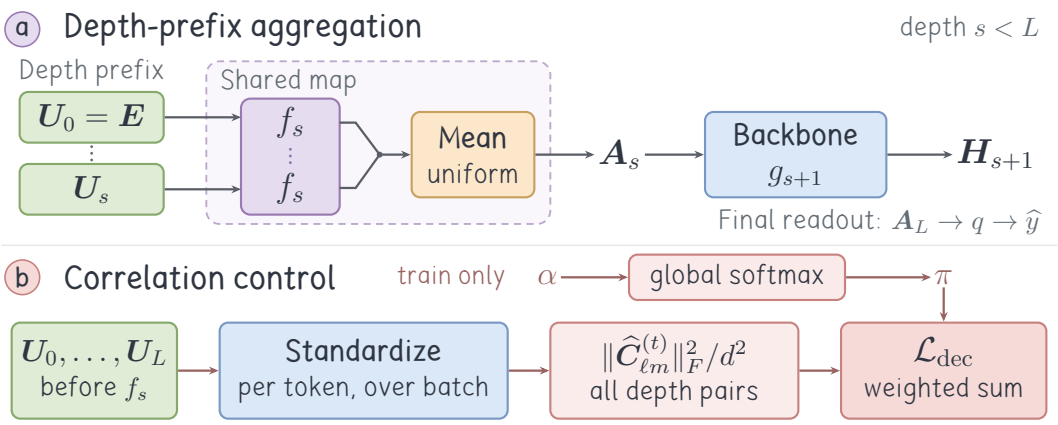


Figure 1: DeCORA overview. (a) A depth-specific f_s is shared across prefix sources before uniform averaging. (b) Regularization term $\mathcal{L}_{\text{dec}}$ acts on sources, with learned softmax weights over depth pairs and averaging over tokens.

3 Method

3.1 DeCORA: Framework Overview

Figure 1 outlines the two paths. For input X, let $\boldsymbol{E} \in \mathbb{R}^{T\times d}$ be the embedding representation and $\boldsymbol{H}_\ell \in \mathbb{R}^{T\times d}$ the hidden representation at layer ℓ. The source representations $\boldsymbol{U}_0 = \boldsymbol{E}$ and $\boldsymbol{U}_\ell = \boldsymbol{H}_\ell$ $(1 \le \ell \le L)$ require aligned tokens and compatible widths. At aggregation depth s, the depth prefix is $(\boldsymbol{U}_0, \ldots, \boldsymbol{U}_s)$. Starting from $\boldsymbol{H}_1 = g_1(\boldsymbol{E})$, define for $s = 1, \ldots, L$:

$$\boldsymbol{V}_{s\leftarrow k} = f_s(\boldsymbol{U}_k), \quad 0 \le k \le s, \qquad \boldsymbol{A}_s = \frac{1}{s+1}\sum_{k=0}^{s} \boldsymbol{V}_{s\leftarrow k}, \tag{1}$$

$$\boldsymbol{H}_{s+1} = g_{s+1}(\boldsymbol{A}_s), \quad s < L, \qquad \hat{y} = q(\boldsymbol{A}_L). \tag{2}$$

The tokenwise adapter $f_s : \mathbb{R}^d \to \mathbb{R}^p$ is specific to aggregation depth s and shared across all source depths $k \le s$. Its output width p matches the receiving block or prediction head. Each g_s retains the backbone's operator; q reads the final aggregate. Earlier sources thus reach later computation through a fixed-width summary.

Our implemented example of f_s sets $p = d$ and composes two linear layers of intermediate width $h \ge d$, with a first-layer bias and no intervening activation:

$$f_s(\boldsymbol{u}) = \boldsymbol{W}_{s,2}(\boldsymbol{W}_{s,1}\boldsymbol{u} + \boldsymbol{b}_{s,1}), \qquad h \ge d, \tag{3}$$

where $\boldsymbol{W}_{s,1} \in \mathbb{R}^{h\times d}$, $\boldsymbol{W}_{s,2} \in \mathbb{R}^{d\times h}$, and $\boldsymbol{b}_{s,1} \in \mathbb{R}^h$. This composition is affine, $f_s(\boldsymbol{u}) = \boldsymbol{W}_s\boldsymbol{u} + \boldsymbol{b}_s$, with $\boldsymbol{W}_s = \boldsymbol{W}_{s,2}\boldsymbol{W}_{s,1}$ and $\boldsymbol{b}_s = \boldsymbol{W}_{s,2}\boldsymbol{b}_{s,1}$. The experimental width is specified in Appendix A.1.

The correlation objective shapes sources before this map (Section 3.2). For the affine instance above, Section 3.3 explains how their geometry passes through it, and Section 3.4 relates aggregation to predictive access. Adding a nonlinear activation would extend expressivity while retaining the wrapper and source-level objective; our analysis develops the affine instance in Equation (3).

3.2 Token-Conditional Cross-Depth Correlation Control

We regularize sources $\boldsymbol{U}_\ell$ before depth-specific transformation. At each retained token t, standardize features over the same $n_t \ge 2$ valid examples at every compared depth:

$$\boldsymbol{z}^{(t)}_{\ell,b} = (\boldsymbol{u}^{(t)}_{\ell,b} - \hat{\boldsymbol{\mu}}^{(t)}_\ell) \oslash (\hat{\boldsymbol{s}}^{(t)}_\ell + \varepsilon), \qquad \widehat{\boldsymbol{C}}^{(t)}_{\ell m} = \frac{(\boldsymbol{Z}^{(t)}_\ell)^\top \boldsymbol{Z}^{(t)}_m}{n_t - 1}, \tag{4}$$

where $\hat{\boldsymbol{s}}$ is the sample standard deviation, $\varepsilon > 0$, and $\boldsymbol{Z}^{(t)}_\ell \in \mathbb{R}^{n_t\times d}$. Without masking, $n_t = B$, the batch size. The feature axis is preserved: token positions are neither concatenated into a single feature vector nor pooled before standardization.

The regularizer can use a star graph, $\mathcal{E}_{\text{star}} = \{(0, m) : 1 \le m \le L\}$, or a full pair graph, $\mathcal{E}_{\text{pair}} = \{(\ell, m) : 0 \le \ell < m \le L\}$. We use the full pair graph, assigning each pair a trainable logit $\alpha_{\ell m}$. A global softmax gives the effective weights; the initialization favors embedding–hidden pairs:

$$\pi_{\ell m} = \frac{\exp(\alpha_{\ell m})}{\sum_{(i,j)\in\mathcal{E}_{\text{pair}}} \exp(\alpha_{ij})}, \qquad \alpha^{(0)}_{\ell m} = \begin{cases} 1, & \ell = 0, \\ -1, & \ell \ge 1. \end{cases} \tag{5}$$

Each embedding–hidden pair initially receives e^2 times the weight of each hidden–hidden pair. All weights are positive and sum to one. We minimize

$$\mathcal{L}_{\text{dec}} = \frac{1}{T}\sum_{t=1}^{T} \sum_{(\ell,m)\in\mathcal{E}_{\text{pair}}} \pi_{\ell m} \frac{\|\widehat{\boldsymbol{C}}^{(t)}_{\ell m}\|_F^2}{d^2}, \qquad \mathcal{L} = \mathcal{L}_{\text{task}} + \mathcal{L}_{\text{dec}}, \tag{6}$$

where $\mathcal{L}_{\text{task}}$ is binary cross-entropy. This nonnegative penalty learns relative pair weights with fixed total mass and a unit loss coefficient. Softmax is recomputed at each training step; its gradients update the logits jointly with the backbone (Appendix A.1). The logits add no inference parameters. Direct evaluation costs $O(TBd^2L^2)$ per batch; a fixed star uses L pairs.

At a fixed token, let $\boldsymbol{Z}_\ell$ now denote the centered, standardized population vector and let $\mathcal{S}_\ell = \{\boldsymbol{a}^\top \boldsymbol{Z}_\ell(\cdot) : \boldsymbol{a} \in \mathbb{R}^d\}$. The cross-covariance $\boldsymbol{C}_{\ell m} = \mathbb{E}[\boldsymbol{Z}_\ell \boldsymbol{Z}_m^\top]$ satisfies

$$\|\boldsymbol{C}_{\ell m}\|_F^2 = \sum_{i,j} \langle Z_{\ell i}, Z_{mj}\rangle_{L_2}^2, \qquad \boldsymbol{C}_{\ell m} = 0 \iff \mathcal{S}_\ell \perp \mathcal{S}_m. \tag{7}$$

Theorem 1 derives this identity by expanding the covariance entries as coordinate-function inner products. Each term in Equation (6) therefore penalizes repeated linear responses. Corollary 1 relates small overlap to limited linear predictability from a well-conditioned prefix. Since the weights also adapt, we measure unweighted overlap to assess whether the sources actually become less redundant; Appendix B.1 makes this distinction precise.

3.3 Depth-Prefix Aggregation

The correlation objective shapes source relationships, while Equation (1) exposes their summary to later layers. For the affine instance in Equation (3), write $f_s(\boldsymbol{u}) = \boldsymbol{W}_s\boldsymbol{u} + \boldsymbol{b}_s$. At a fixed token,

$$\boldsymbol{A}_s = f_s\left(\frac{1}{s+1}\sum_{k=0}^{s} \boldsymbol{U}_k\right), \qquad \operatorname{Cov}(\boldsymbol{V}_{s\leftarrow k}, \boldsymbol{V}_{s\leftarrow m}) = \boldsymbol{W}_s \boldsymbol{D}_k \boldsymbol{C}_{km} \boldsymbol{D}_m \boldsymbol{W}_s^\top, \tag{8}$$

where $\boldsymbol{U}_k - \mathbb{E}[\boldsymbol{U}_k] = \boldsymbol{D}_k\boldsymbol{Z}_k$ and $\boldsymbol{D}_k$ contains positive population coordinate scales. Centering removes the bias. Theorem 2 shows that this map preserves zero overlap and bounds approximate overlap through the scales and $\|\boldsymbol{W}_s\|_2$. Maintaining $\boldsymbol{S}_s = \boldsymbol{S}_{s-1} + \boldsymbol{U}_s$, $\boldsymbol{S}_0 = \boldsymbol{E}$, gives $\boldsymbol{A}_s = f_s(\boldsymbol{S}_s/(s+1))$: inference needs a running sum and one map application per depth. Training still compares individual sources. This shortcut relies on the affine form in Equation (3).

Theorem 3 explains non-cancellation by expanding the sum's energy and bounding its cross-terms using transformed-space coherence μ_s. The average retains at least $(1 - s\mu_s)/(s+1)^2$ times the summed source energy. This bound supports stable averaging when $\mu_s < 1/s$, a condition that small empirical DCO alone does not establish.

The remaining issue is *access*. A full-prefix readout can assign different coefficients to each source; $\boldsymbol{a}^\top\boldsymbol{A}_s$ uses $\boldsymbol{W}_s^\top\boldsymbol{a}/(s+1)$ for every source. Corollary 2 quantifies the resulting loss of target-relevant access. Source separation and aggregation must therefore be learned under the task objective. Appendix C proves these results; Section 4.4 tests both components and their combination.

3.4 Design Rationale and Representation Diagnostics

A new response helps only if it contributes to prediction. For a fixed source sequence, let $\mathcal{V}_{-1} = \{0\}$, $\mathcal{V}_\ell = \sum_{k=0}^{\ell} \mathcal{S}_k$, and $\mathcal{I}_\ell = (I - P_{\mathcal{V}_{\ell-1}})\mathcal{S}_\ell$, where $P_{\mathcal{V}}$ is the L_2-orthogonal projector onto $\mathcal{V}$. For label Y, define the optimal population squared-loss risk as $\mathcal{R}^\star_\ell = \inf_{f\in\mathcal{V}_\ell} \mathbb{E}[(Y - f(X))^2]$. Theorem 4 separates the preceding prefix from the orthogonal innovation and gives

$$\mathcal{R}^\star_{\ell-1} - \mathcal{R}^\star_\ell = \|P_{\mathcal{I}_\ell}\eta\|_{L_2}^2, \qquad \eta(X) = \mathbb{E}[Y \mid X]. \tag{9}$$

Pythagoras yields this identity: representational novelty without target alignment gives zero gain. To account for the actual summary, let $\mathcal{A}_s$ be the linear span of its centered coordinates. For the affine instance, $\mathcal{A}_s \subseteq \mathcal{V}_s$, so Corollary 2 gives

$$\mathcal{R}^\star_{A_s} - \mathcal{R}^\star_s = \|P_{\mathcal{V}_s}\eta - P_{\mathcal{A}_s}\eta\|^2_{L_2}, \qquad \mathcal{R}^\star_{A_s} = \inf_{f\in\mathcal{A}_s} \mathbb{E}[(Y - f(X))^2]. \tag{10}$$

Equations (9) and (10) separate two requirements for useful depth: later sources must add target-aligned responses, and the aggregate must make those responses accessible. Corollary 3 bounds cumulative gain along a fixed nested sequence. These readout identities explain the paired design; the experiments evaluate the actual CTR loss and prediction head. The following label-free diagnostics complement task evaluation by measuring overlap, new sample geometry, and aggregate alignment.

Depthwise Correlation Overlap (DCO) measures repeated responses:

$$\mathrm{DCO}_{\ell m} = \frac{1}{T}\sum_t \frac{\|\widehat{\boldsymbol{C}}^{(t)}_{\ell m}\|^2_F}{d^2}. \tag{11}$$

Theorem 1 identifies this coordinate overlap, and $\mathcal{L}_{\mathrm{dec}} = \sum_{(\ell,m)} \pi_{\ell m}\,\mathrm{DCO}_{\ell m}$. Unweighted DCO separates changes in representation overlap from redistribution of the normalized pair weights. Corollary 1 supplies the conditioning assumptions for a stronger subspace interpretation.

For each token, form the centered sample matrix $\boldsymbol{X}^{(t)}_\ell$, its linear kernel $\boldsymbol{K}^{(t)}_\ell = \boldsymbol{X}^{(t)}_\ell \boldsymbol{X}^{(t)\top}_\ell$, and its unit-Frobenius normalization $\boldsymbol{G}^{(t)}_\ell$. Let $\Pi^{(t)}_{\ell-1}$ project onto the Frobenius span of the preceding normalized kernels. The *Incremental Kernel Residual (IKR)* is

$$\mathrm{IKR}_\ell = \frac{1}{T}\sum_t \|(I - \Pi^{(t)}_{\ell-1})\boldsymbol{G}^{(t)}_\ell\|^2_F. \tag{12}$$

By Pythagoras, Proposition 1 bounds IKR in $[0, 1]$: zero indicates membership in the prefix span; one indicates orthogonality. We display $\mathrm{IKR}_0 = 1$ for the empty prefix. IKR measures kernel representational novelty, distinct from the target-aligned gain in Equation (9).

Finally, set $\boldsymbol{Q}^{(t)}_s = \sum_{k=0}^s \boldsymbol{G}^{(t)}_k$ and let $\boldsymbol{G}^{(t)}_{A_s}$ be the normalized centered kernel of the aggregate. *Prefix Aggregate Coverage (PAC)* measures depth-prefix alignment:

$$\mathrm{PAC}_s = \frac{1}{T}\sum_t \left\langle \boldsymbol{G}^{(t)}_{A_s}, \frac{\boldsymbol{Q}^{(t)}_s}{\|\boldsymbol{Q}^{(t)}_s\|_F} \right\rangle_F. \tag{13}$$

Proposition 2 bounds PAC in $[0, 1]$; one indicates proportional aggregate and reference kernels. Redundant sources can also yield high PAC, which does not estimate Corollary 2's exposure gap. Section 4.5 examines all three diagnostics; Appendix B.2 gives proofs and zero-kernel conventions.

4 Experiments

4.1 Experimental Setup

We evaluate four **structurally diverse backbones**: RankMixer uses a mixer-type architecture (Zhu et al., 2025b; Yu et al., 2022); WuKong builds on factorization-machine-based interactions (Zhang et al., 2024); DHEN hierarchically combines heterogeneous interaction modules (Zhang et al., 2022); and FAT uses field-aware attention (Yan et al., 2026). This selection tests whether the benefits of cross-depth organization extend across distinct feature-interaction mechanisms. Experiments use Avazu and Criteo (Wang & Cukierski, 2014; Tien et al., 2014) through FuxiCTR (Zhu et al., 2021), with AUC (↑) and Logloss (LL; ↓) as evaluation metrics. Vanilla and wrapped models share data splits, optimizer policies, training budgets, and backbone settings. All four backbones use the global softmax and initialization in Equation (5).

We sweep Base, $2\times$, $4\times$, $6\times$, and $8\times$ depth, with a $16\times$ extension. Depth-method comparisons cover DenseFormer (Dense), manifold-constrained Hyper-Connections (mHC), Orthogonal Residual Update (ORU), and Attention Residuals (AttnRes) at $4\times$ and $16\times$; component and width controls use $8\times$ and $16\times$. All scores and diagnostic curves are ten-seed means. Depth-sweep and component tables include standard deviations for every method; depth-method tables consistently report means. Dataset statistics, protocols, and complete results are in Appendix A.

Table 1: Predictive performance at $8\times$. Ten-seed means appear above std ($\pm$). ΔAUC is the absolute gain in units of 10^{-3}; bold marks the better mean.

Dataset	Backbone	Vanilla		DeCORA		ΔAUC
		AUC ↑	Logloss ↓	AUC ↑	Logloss ↓	(10^{-3})
Avazu	RankMixer	0.79468 ± 8e−5	0.37091 ± 6e−5	**0.79636** ± 5e−5	**0.36991** ± 6e−5	+1.68
	WuKong	0.79354 ± 8e−5	0.37197 ± 7e−5	**0.79568** ± 7e−5	**0.37064** ± 5e−5	+2.14
	DHEN	0.79411 ± 7e−5	0.37274 ± 7e−5	**0.79619** ± 7e−5	**0.37002** ± 5e−5	+2.08
	FAT	0.79205 ± 1e−4	0.37272 ± 7e−5	**0.79464** ± 7e−5	**0.37087** ± 7e−5	+2.59
Criteo	RankMixer	0.81354 ± 1e−4	0.43895 ± 1e−4	**0.81481** ± 8e−5	**0.43749** ± 8e−5	+1.27
	WuKong	0.81367 ± 8e−5	0.43902 ± 1e−4	**0.81449** ± 8e−5	**0.43768** ± 7e−5	+0.82
	DHEN	0.81337 ± 1e−4	0.43907 ± 8e−5	**0.81419** ± 8e−5	**0.43791** ± 8e−5	+0.82
	FAT	0.81345 ± 9e−5	0.43915 ± 1e−4	**0.81456** ± 8e−5	**0.43736** ± 8e−5	+1.11

4.2 Predictive Performance and Useful Depth

Consistent gains at matched depth. Across the five primary depth settings, DeCORA improves both AUC and LL in all $2 \times 4 \times 5 = 40$ backbone–dataset–depth comparisons. At $8\times$ (Table 1), absolute AUC gains range from 0.00082 to 0.00259, averaging 0.00156 across the eight pairs, while relative LL reductions range from 0.26% to 0.73%, averaging 0.40%. The gains thus extend across diverse interaction architectures and consistently benefit both ranking quality and probabilistic prediction loss. Moreover, at $8\times$, the AUC gain is 8.2–29.7 times the larger of the two methods' run-level standard deviations in each matched comparison, with the corresponding LL reduction reaching 13.4–38.9 times that scale. Thus, descriptively, the mean improvements are substantially larger than the observed run-to-run variation.

Practical scale of the improvements. Absolute AUC changes of this size can be substantial in industrial CTR prediction. Zhou et al. (2018) report that a 0.001 gain in user-weighted AUC can justify deployment in a high-traffic commercial advertising system, providing useful industry context for the scale of our offline gains. Six of our eight $8\times$ comparisons exceed 0.001 absolute AUC improvement, as do all eight $16\times$ comparisons. Moreover, these gains consistently coincide with lower LL and become larger with depth, reaching a mean AUC improvement of 0.00199 at $16\times$. While the cited threshold provides context rather than a direct conversion to online impact, the breadth and increasing magnitude of these gains highlight their potential practical relevance.

The benefit increases as the backbone gets deeper. Figure 2 shows the complete AUC trajectory, including the $16\times$ extension. Vanilla reaches its highest observed AUC at Base–$6\times$, whereas DeCORA improves at every tested multiplier for every backbone–dataset pair. At $8\times$, it also exceeds the *best observed vanilla AUC anywhere in the sweep* by 0.00069–0.00171, showing that its advantage is not solely due to comparison with an over-deep baseline. At $16\times$, its mean AUC gain over matched-depth vanilla further grows to 0.00199. Overall, the mean trajectories show an increasingly pronounced benefit from DeCORA as the backbone becomes deeper, with adjacent-depth differences interpreted in light of their run-to-run variation. This trend describes the tested depth range rather than an unbounded scaling law. The exact AUC and LL sweeps are in Tables 7 and 8.

4.3 Alternative Depth Mechanisms and Training Cost

Table 2 compares RankMixer and WuKong at $16\times$, where their vanilla counterparts have already lost accuracy. DeCORA obtains the highest AUC and lowest LL in all four settings. Its AUC advantage over the strongest competing depth mechanism is 0.00110 and 0.00087 on Avazu, and 0.00078 and

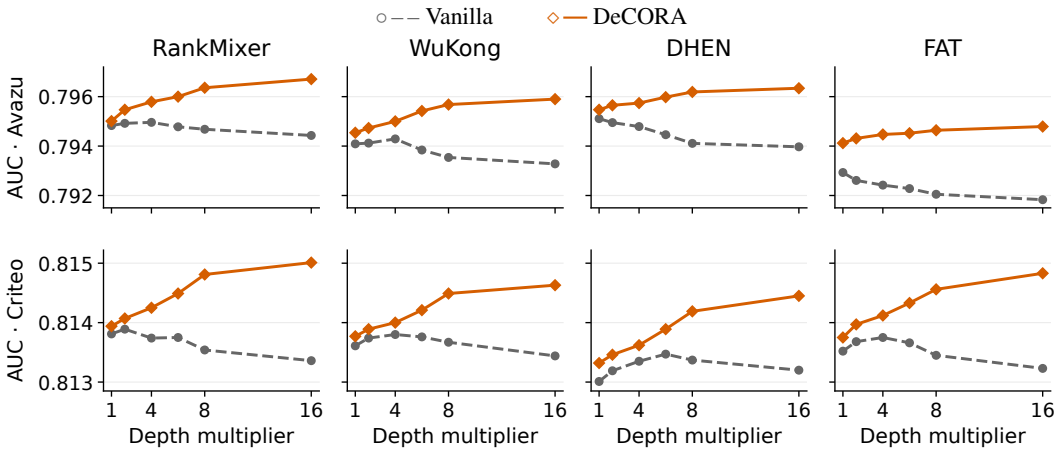


Figure 2: AUC versus depth multiplier. The full method improves at every tested depth in all eight settings; vanilla peaks earlier. Points are the reported ten-seed means, including the $16\times$ extension.

Table 2: Depth mechanisms at $16\times$. Mem.: GPU memory (GB); Time: minutes per training epoch. Best results are shown in bold. Results are ten-seed averages. See Appendix A.3 for full results.

Backbone	Method	Avazu				Criteo			
		AUC ↑	Logloss ↓	Mem.	Time	AUC ↑	Logloss ↓	Mem.	Time
RankMixer	Vanilla	0.79443	0.37118	3.9	39.8	0.81336	0.43917	4.1	47.2
	DenseFormer	0.79483	0.37101	4.6	41.5	0.81376	0.43873	4.8	49.2
	mHC	0.79561	0.37083	4.7	44.6	0.81423	0.43801	4.8	52.4
	ORU	0.79501	0.37099	4.5	42.1	0.81359	0.43834	4.7	50.5
	AttnRes	0.79552	0.37052	4.9	43.8	0.81386	0.43850	5.3	51.0
	DeCORA	**0.79671**	**0.36957**	4.9	44.0	**0.81501**	**0.43723**	5.3	51.3
WuKong	Vanilla	0.79328	0.37222	3.7	15.8	0.81344	0.43922	4.0	23.8
	DenseFormer	0.79387	0.37203	4.3	16.7	0.81352	0.43910	4.9	24.6
	mHC	0.79503	0.37107	4.4	17.1	0.81398	0.43832	5.0	27.9
	ORU	0.79442	0.37139	4.3	16.6	0.81370	0.43876	4.7	25.1
	AttnRes	0.79491	0.37098	4.6	17.3	0.81381	0.43828	5.2	27.3
	DeCORA	**0.79590**	**0.37019**	4.7	17.3	**0.81463**	**0.43739**	5.2	27.5

0.00065 on Criteo, for RankMixer and WuKong, respectively. While alternative depth mechanisms can improve upon vanilla, particularly at greater depth, DeCORA retains a consistent advantage over them. The same ordering holds for both metrics across all 16 settings when including DHEN, FAT, and $4\times$ (see Appendix A.3 for more results).

Accuracy comes with a measurable training overhead. Across 16 backbone–dataset–depth settings, DeCORA increases training time per epoch by 8.7%–18.5% and GPU memory by 0.6–1.2 GB relative to vanilla. DenseFormer and ORU are generally lighter. Compared with AttnRes, however, the training cost is similar: differences in Table 2 are at most 0.3 minutes per epoch and 0.1 GB, while DeCORA achieves better AUC and LL in every setting. Thus, within the measured training costs, DeCORA provides consistent accuracy gains at an overhead comparable to AttnRes.

4.4 Component Analysis and Parameter Overhead

Neither component alone reproduces the depth benefit. The complete method outperforms aggregation only, loss only, and Vanilla-W in both AUC and LL in all 16 component settings. Aggregation alone improves both metrics over vanilla in every setting, but its AUC falls from $8\times$ to $16\times$ in all eight backbone–dataset pairs. Over the same interval, the full method raises AUC by 0.00014–0.00035 in every pair. For RankMixer on Avazu, for example, aggregation only falls from 0.79554 to 0.79521, while the full method rises from 0.79636 to 0.79671. At $16\times$, the full method's advantage over the strongest component or width control is larger than the run-level variability

Table 3: Component and capacity controls at $16\times$. Agg. only denotes aggregation only; Vanilla-W widens FFNs to match the full method. ΔP is the dense-parameter increase over vanilla.

Backbone	Method	Avazu			Criteo		
		AUC ↑	Logloss ↓	ΔP	AUC ↑	Logloss ↓	ΔP
RankMixer	Vanilla	0.79443 ± 7e−5	0.37118 ± 1e−4	0.0%	0.81336 ± 8e−5	0.43917 ± 9e−5	0.0%
	Agg. only	0.79521 ± 9e−5	0.37072 ± 7e−5	3.7%	0.81402 ± 1e−4	0.43838 ± 1e−4	1.2%
	Loss only	0.79473 ± 7e−5	0.37135 ± 8e−5	0.0%	0.81351 ± 2e−4	0.43902 ± 9e−5	0.0%
	Vanilla-W	0.79455 ± 1e−4	0.37107 ± 8e−5	3.7%	0.81366 ± 9e−5	0.43885 ± 1e−4	1.2%
	DeCORA	**0.79671** ± 8e−5	**0.36957** ± 3e−5	3.7%	**0.81501** ± 7e−5	**0.43723** ± 6e−5	1.2%
WuKong	Vanilla	0.79328 ± 7e−5	0.37222 ± 9e−5	0.0%	0.81344 ± 2e−4	0.43922 ± 8e−5	0.0%
	Agg. only	0.79410 ± 7e−5	0.37173 ± 8e−5	3.9%	0.81381 ± 8e−5	0.43821 ± 1e−4	2.1%
	Loss only	0.79359 ± 8e−5	0.37146 ± 1e−4	0.0%	0.81350 ± 9e−5	0.43878 ± 8e−5	0.0%
	Vanilla-W	0.79390 ± 9e−5	0.37120 ± 7e−5	3.9%	0.81306 ± 1e−4	0.43975 ± 2e−4	2.1%
	DeCORA	**0.79590** ± 5e−5	**0.37019** ± 8e−5	3.9%	**0.81463** ± 6e−5	**0.43739** ± 8e−5	2.1%

Table 4: Change in mean AUC from $8\times$ to $16\times$ (10^{-3} units) for all four component variants. Differences use ten-seed means; source means and std are in Table 3 and Appendix A.4.

Backbone	Avazu				Criteo			
	Vanilla	Agg. only	Loss only	DeCORA	Vanilla	Agg. only	Loss only	DeCORA
RankMixer	−0.25	−0.33	−0.19	**+0.35**	−0.18	−0.15	+0.13	**+0.20**
WuKong	−0.26	−0.27	−0.14	**+0.22**	−0.23	−0.09	−0.07	**+0.14**
DHEN	−0.14	−0.15	−0.30	**+0.15**	−0.17	−0.02	−0.11	**+0.26**
FAT	−0.22	−0.16	−0.62	**+0.15**	−0.22	−0.06	−0.08	**+0.27**

(Table 3). Table 4 illustrates the distinction discussed in Section 3.4: access to earlier sources alone does not reproduce the depth benefit of the full method.

Loss only shows a limitation. It loses AUC in seven of the eight depth extensions, with its sole increase being 0.00013 for RankMixer on Criteo. At $8\times$, it lowers AUC for all four Criteo backbones despite improving LL in three of them, and it worsens RankMixer's Avazu LL at $16\times$. Thus, neither aggregation nor correlation control alone reproduces the gains of the full method; the depth benefit emerges when the two are used together (Table 3; Appendix A.4).

The gain is not explained by parameter count alone. The adapters add 0.1%–3.9% dense parameters, excluding embeddings. Aggregation only has the same adapter overhead, while loss only adds trainable pair weights but no inference parameters, with a reported dense-parameter increment of 0.0%. Vanilla-W matches the full method's reported dense parameter count by widening FFNs, yet trails the full method in both AUC and LL in every component setting. Although widening and changing connectivity are different interventions, this parameter-matched control shows that the reported parameter increase alone does not account for the consistent gains of the full method.

4.5 Representation Evolution Across Depth

Figure 3 shows the $8\times$ Avazu results. The full method reduces RankMixer's broad cross-depth overlap and retains more late-source kernel residual than vanilla. At source 8, IKR is approximately

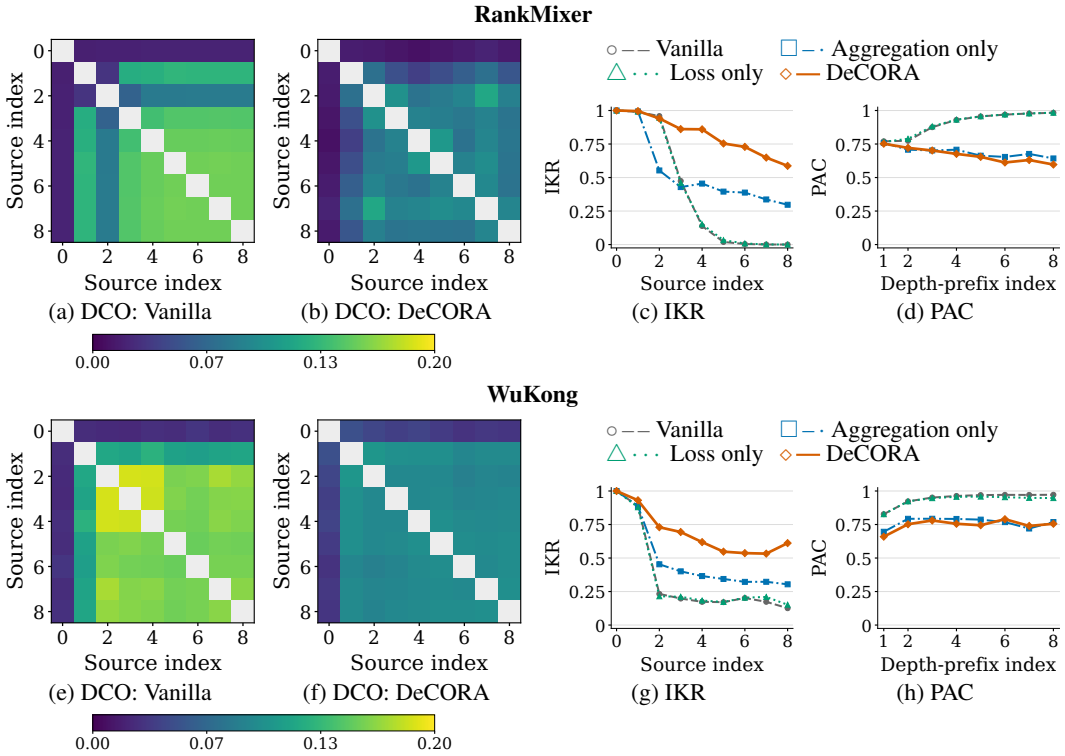


Figure 3: Representation diagnostics on Avazu at $8\times$. DCO compares vanilla and DeCORA; IKR and PAC compare all four variants. Each DCO row shares its color scale across variants and datasets; diagonal cells are masked. IKR starts at the empty-prefix reference of one.

0.59 versus nearly zero for RankMixer, and 0.61 versus 0.13 for WuKong. Across layers, the full method consistently exhibits lower DCO and higher IKR, matching the directions motivated by our analysis. In these results, aggregation only is intermediate, while loss only remains closer to vanilla.

PAC provides a complementary view of prefix alignment. At the final aggregation depth, the full method has PAC of about 0.60 and 0.76 for RankMixer and WuKong, versus 0.98 and 0.97 for vanilla. PAC is not a quantity to maximize: a redundant prefix can still be highly aligned with its aggregate. Together, these diagnostics characterize the representation changes associated with DeCORA and align with its improved task performance. This correspondence is clearest for the full method; individual ablations need not exhibit a one-to-one relation between each diagnostic and task performance (see Appendix A.5 for the full results).

DCO measures coordinate overlap; IKR measures kernel geometry beyond the prefix; and PAC measures aggregate–prefix alignment. Their roles follow Theorem 1 and Propositions 1–2, motivating their use to interpret the observed representation changes. Together with the component comparisons motivated by Equations (9) and (10), they provide an explanatory view of the empirical depth benefit.

5 Conclusion

DeCORA combines token-conditional correlation control with depth-prefix aggregation to coordinate what successive layers learn and how their representations are reused. Our analysis provides an explanatory view through coordinate-function overlap, representation geometry, and aggregate access. Across four structurally different backbones on Avazu and Criteo, DeCORA improves both AUC and Logloss in all 40 matched settings and, among the component and capacity variants, uniquely improves mean AUC in all eight backbone–dataset pairs from $8\times$ to $16\times$. Together with the connectivity comparisons, these results show consistent benefits from coordinating representation learning and reuse as depth increases. Future work can extend this study beyond offline CTR benchmarks and compatible token interfaces to online serving, heterogeneous representations, and nonlinear adapters.

REFERENCES

Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. Measuring statistical dependence with hilbert-schmidt norms. In Sanjay Jain, Hans Ulrich Simon, and Etsuji Tomita (eds.), *Algorithmic Learning Theory*, pp. 63–77, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. ISBN 978-3-540-31696-1.

Kimi Team, Guangyu Chen, Yu Zhang, Jianlin Su, Weixin Xu, Siyuan Pan, Yaoyu Wang, Yucheng Wang, Guanduo Chen, Bohong Yin, Yutian Chen, Junjie Yan, Ming Wei, Y. Zhang, Fanqing Meng, Chao Hong, Xiaotong Xie, Shaowei Liu, Enzhe Lu, Yunpeng Tai, Yanru Chen, Xin Men, Haiqing Guo, Y. Charles, Haoyu Lu, Lin Sui, Jinguo Zhu, Zaida Zhou, Weiran He, Weixiao Huang, Xinran Xu, Yuzhi Wang, Guokun Lai, Yulun Du, Yuxin Wu, Zhilin Yang, and Xinyu Zhou. Attention residuals, 2026. URL https://arxiv.org/abs/2603.15031.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. URL https://arxiv.org/abs/1412.6980.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3519–3529. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/kornblith19a.html.

Cheng Luo, Zefan Cai, and Junjie Hu. Delta attention residuals, 2026. URL https://arxiv.org/abs/2605.18855.

Giyeong Oh, Woohyun Cho, Siyeol Kim, Suhwan Choi, and Youngjae Yu. Revisiting residual connections: Orthogonal updates for stable and efficient deep networks. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 71725–71759. Curran Associates, Inc., 2025. doi: 10.52202/085713-2409. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/67c15da4a9340140c60783d9a175fd3f-Paper-Conference.pdf.

Matteo Pagliardini, Amirkeivan Mohtashami, Francois Fleuret, and Martin Jaggi. Denseformer: Enhancing information flow in transformers via depth weighted averaging. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 136479–136508. Curran Associates, Inc., 2024. doi: 10.52202/079017-4336. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/f67449c7ab72f441d3a713b046c6818c-Paper-Conference.pdf.

Jean-Baptiste Tien, joycenv, and Olivier Chapelle. Display advertising challenge. https://kaggle.com/competitions/criteo-display-ad-challenge, 2014. Kaggle.

Jiancheng Wang, Mingjia Yin, Hao Wang, and Enhong Chen. Enhancing ctr prediction with decorrelated expert networks, 2025. URL https://arxiv.org/abs/2505.17925.

Peng Wang, Huikang Liu, Druv Pai, Yaodong Yu, Zhihui Zhu, Qing Qu, and Yi Ma. A global geometric analysis of maximal coding rate reduction. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 51012–51040. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/wang24as.html.

Steve Wang and Will Cukierski. Click-through rate prediction. https://kaggle.com/competitions/avazu-ctr-prediction, 2014. Kaggle.

Zhenda Xie, Yixuan Wei, Huanqi Cao, Chenggang Zhao, Chengqi Deng, Jiashi Li, Damai Dai, Huazuo Gao, Mingyu Xu, Kuai Yu, Liang Zhao, Shangyan Zhou, Zhean Xu, Zhengyan Zhang, Wangding Zeng, Shengding Hu, Yuqing Wang, Jingyang Yuan, Lean Wang, and Wenfeng Liang. mHC: Manifold-constrained hyper-connections. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=mDhyxu8WRb.

Bencheng Yan, Yuejie Lei, Zhiyuan Zeng, Zheye Deng, Di Wang, Kaiyi Lin, Pengjie Wang, Chuan Yu, Jian Xu, and Bo Zheng. From scaling to structured expressivity: Rethinking transformers for ctr prediction. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, KDD '26, pp. 8358–8368, New York, NY, USA, 2026. Association for Computing Machinery. ISBN 9798400722592. doi: 10.1145/3770855.3818420. URL https://doi.org/10.1145/3770855.3818420.

Weihao Yu, Mi Luo, Pan Zhou, Chenyang Si, Yichen Zhou, Xinchao Wang, Jiashi Feng, and Shuicheng Yan. Metaformer is actually what you need for vision. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10809–10819, 2022. doi: 10.1109/CVPR52688.2022.01055.

Xingda Yu, Yining Li, Xinzhang Liu, Zhihao Yang, Haowei He, Chao Wang, Yongxiang Li, and Shuangyong Song. Dual attention residuals, 2026. URL https://arxiv.org/abs/2607.18730.

Yaodong Yu, Kwan Ho Ryan Chan, Chong You, Chaobing Song, and Yi Ma. Learning diverse and discriminative representations via the principle of maximal coding rate reduction. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 9422–9434. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/6ad4174eba19ecb5fed17411a34ff5e6-Paper.pdf.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stephane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12310–12320. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/zbontar21a.html.

Buyun Zhang, Liang Luo, Xi Liu, Jay Li, Zeliang Chen, Weilin Zhang, Xiaohan Wei, Yuchen Hao, Michael Tsang, Wenjun Wang, Yang Liu, Huayu Li, Yasmine Badr, Jongsoo Park, Jiyan Yang, Dheevatsa Mudigere, and Ellie Wen. Dhen: A deep and hierarchical ensemble network for large-scale click-through rate prediction, 2022. URL https://arxiv.org/abs/2203.11014.

Buyun Zhang, Liang Luo, Yuxin Chen, Jade Nie, Xi Liu, Shen Li, Yanli Zhao, Yuchen Hao, Yantao Yao, Ellie Dingqiao Wen, Jongsoo Park, Maxim Naumov, and Wenlin Chen. Wukong: Towards a scaling law for large-scale recommendation. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 59421–59434. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/zhang24ao.html.

Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18, pp. 1059–1068, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450355520. doi: 10.1145/3219819.3219823. URL https://doi.org/10.1145/3219819.3219823.

Defa Zhu, Hongzhi Huang, Zihao Huang, Yutao Zeng, Yunyao Mao, Banggu Wu, Qiyang Min, and Xun Zhou. Hyper-connections. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=9FqARW7dwB.

Jie Zhu, Zhifang Fan, Xiaoxie Zhu, Yuchen Jiang, Hangyu Wang, Xintian Han, Haoran Ding, Xinmin Wang, Wenlin Zhao, Zhen Gong, Huizhi Yang, Zheng Chai, Zhe Chen, Yuchao Zheng, Qiwei Chen, Feng Zhang, Xun Zhou, Peng Xu, Xiao Yang, Di Wu, and Zuotao Liu. Rankmixer: Scaling up ranking models in industrial recommenders. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, CIKM '25, pp. 6309–6316, New York, NY, USA, 2025b. Association for Computing Machinery. ISBN 9798400720406. doi: 10.1145/3746252.3761507. URL https://doi.org/10.1145/3746252.3761507.

Jieming Zhu, Jinyang Liu, Shuai Yang, Qi Zhang, and Xiuqiang He. Open benchmarking for click-through rate prediction. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, pp. 2759–2769, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450384469. doi: 10.1145/3459637.3482486. URL https://doi.org/10.1145/3459637.3482486.

AI USE STATEMENT

Large language models, including ChatGPT and Gemini, were used to assist with manuscript organization, language editing, mathematical exposition, and LaTeX preparation. The reported experimental means and diagnostic figures were provided by the authors.

A EXPERIMENTAL DETAILS AND COMPLEMENTARY RESULTS

This appendix supplies dataset statistics, implementation details, the remaining depths and backbones, and complementary diagnostics.

A.1 DATASETS, PROTOCOL, AND IMPLEMENTATION DETAILS

Evaluation and comparisons. Scores are reported ten-seed means under FuxiCTR, with matched data, optimization, training budgets, and backbone settings. Resource tables give GPU memory (GB) and training time (minutes/epoch). Depth sweeps and component comparisons report mean and std for every method. The depth-method comparisons use mean scores consistently across all rows. These standard deviations describe variation across runs.

Correlation weights and optimization. Adam updates the raw logits in Equation (5) jointly with the network, using the same learning rate and zero weight decay. Each loss evaluation computes a differentiable global softmax without overwriting the logits. All pairs remain active, with a unit coefficient on $\mathcal{L}_{\mathrm{dec}}$. Appendix B.1 relates these updates to the unweighted diagnostics.

Table 5: Dataset sizes under the experimental preprocessing. Counts are rounded to millions.

Dataset	Train	Validation	Test	Fields
Avazu	32.3M	4.0M	4.0M	24
Criteo	33.0M	8.3M	4.6M	39

Table 6: Implementation details.

Item	Configuration
Data processing	We use the standard preprocessing pipeline provided by FuxiCTR for both datasets. The embedding dimension is set to 24 for Avazu and 39 for Criteo.
Backbones	For each architecture, the base model contains a single backbone layer, while a $k\times$ variant contains k such layers. All non-depth architectural configurations follow the original papers; backbone layers are dimension-preserving, each base model has fewer than 5M parameters, and all weights use Kaiming-normal initialization.
Depth-specific adapter	Two linear layers of widths $d \to \lfloor 8d/3 \rfloor \to d$; bias in the first layer only; no intermediate activation. Weights use Kaiming-normal initialization.
Pair weights	Global softmax over all pairs. Raw logits initialized to 1 (embedding–hidden) and -1 (hidden–hidden), and optimized jointly with the backbone. Loss coefficient fixed at one.
Optimization	Adam (Kingma & Ba, 2017); learning rate 10^{-3}; weight decay 0; early-stopping patience of two epochs without validation improvement.
Depth baselines	Recommended configurations from the original papers, with expanded search ranges to tune each baseline.
Hard- and software	One GPU with 48 GB memory; Python 3.8; CUDA 11.8.

Component definitions. *Aggregation only* (originally *w/ wrapper*) adds the depth-prefix aggregation path and adapters without $\mathcal{L}_{\text{dec}}$. *Loss only* regularizes vanilla; its learned weights add no inference parameters. *Vanilla-W* widens FFNs to match the full method's reported dense parameter count. Increments exclude embeddings and are reported as equal at $8\times$ and $16\times$ for each backbone–dataset pair.

Implementation. Table 6 summarizes the shared settings; the code supplement's README documents the complete experiment configurations. The evaluated adapter is the affine example in Equation (3), with $p = d$, linking the reported results to the covariance and prefix-access analysis.

A.2 Complete Depth Sweeps

Tables 7 and 8 report Base–$8\times$ means. The depth-method comparisons include $16\times$ results. The full method's AUC increases at every multiplier in all eight backbone–dataset pairs. LL improves over matched vanilla in every primary setting but is not strictly monotone: for example, Avazu DHEN changes from 0.37031 at $2\times$ to 0.37038 at $4\times$, and Avazu FAT from 0.37167 at Base to 0.37179 at $2\times$. Our depth-scaling claim therefore concerns AUC.

Table 7: Avazu depth sweep. AUC (↑) and Logloss (↓) means over ten seeds appear above std (±). Bold marks the better mean at each depth.

Backbone	Metric	Method	Base	2×	4×	6×	8×
RankMixer	AUC ↑	Vanilla	0.79483 ± 8e−5	0.79492 ± 1e−4	0.79496 ± 5e−5	0.79478 ± 7e−5	0.79468 ± 8e−5
		DeCORA	**0.79501** ± 5e−5	**0.79547** ± 6e−5	**0.79579** ± 8e−5	**0.79600** ± 6e−5	**0.79636** ± 5e−5
	Logloss ↓	Vanilla	0.37143 ± 6e−5	0.37107 ± 7e−5	0.37086 ± 9e−5	0.37089 ± 8e−5	0.37091 ± 6e−5
		DeCORA	**0.37093** ± 7e−5	**0.37055** ± 6e−5	**0.37023** ± 4e−5	**0.37015** ± 5e−5	**0.36991** ± 6e−5
WuKong	AUC ↑	Vanilla	0.79409 ± 7e−5	0.79412 ± 8e−5	0.79429 ± 7e−5	0.79384 ± 9e−5	0.79354 ± 8e−5
		DeCORA	**0.79454** ± 7e−5	**0.79473** ± 5e−5	**0.79500** ± 3e−5	**0.79542** ± 4e−5	**0.79568** ± 7e−5
	Logloss ↓	Vanilla	0.37167 ± 9e−5	0.37148 ± 7e−5	0.37234 ± 1e−4	0.37174 ± 6e−5	0.37197 ± 7e−5
		DeCORA	**0.37122** ± 4e−5	**0.37113** ± 8e−5	**0.37088** ± 6e−5	**0.37084** ± 7e−5	**0.37064** ± 5e−5
DHEN	AUC ↑	Vanilla	0.79511 ± 1e−4	0.79495 ± 6e−5	0.79479 ± 9e−5	0.79446 ± 8e−5	0.79411 ± 7e−5
		DeCORA	**0.79547** ± 4e−5	**0.79565** ± 7e−5	**0.79574** ± 5e−5	**0.79598** ± 6e−5	**0.79619** ± 7e−5
	Logloss ↓	Vanilla	0.37062 ± 7e−5	0.37100 ± 8e−5	0.37097 ± 5e−5	0.37117 ± 9e−5	0.37274 ± 7e−5
		DeCORA	**0.37035** ± 6e−5	**0.37031** ± 3e−5	**0.37038** ± 4e−5	**0.37011** ± 4e−5	**0.37002** ± 5e−5
FAT	AUC ↑	Vanilla	0.79293 ± 7e−5	0.79261 ± 9e−5	0.79242 ± 7e−5	0.79228 ± 5e−5	0.79205 ± 1e−4
		DeCORA	**0.79412** ± 6e−5	**0.79431** ± 4e−5	**0.79447** ± 8e−5	**0.79452** ± 3e−5	**0.79464** ± 7e−5
	Logloss ↓	Vanilla	0.37203 ± 5e−5	0.37213 ± 1e−4	0.37229 ± 8e−5	0.37253 ± 7e−5	0.37272 ± 7e−5
		DeCORA	**0.37167** ± 3e−5	**0.37179** ± 7e−5	**0.37104** ± 5e−5	**0.37101** ± 6e−5	**0.37087** ± 7e−5

Table 8: Criteo depth sweep. AUC (↑) and Logloss (↓) means over ten seeds appear above std (±). Bold marks the better mean at each depth.

Backbone	Metric	Method	Base	2×	4×	6×	8×
RankMixer	AUC ↑	Vanilla	0.81381 ± 1e-4	0.81389 ± 1e-4	0.81374 ± 2e-4	0.81375 ± 9e-5	0.81354 ± 1e-4
		DeCORA	**0.81394** ± 8e-5	**0.81407** ± 1e-4	**0.81425** ± 9e-5	**0.81449** ± 6e-5	**0.81481** ± 8e-5
	Logloss ↓	Vanilla	0.43879 ± 9e-5	0.43881 ± 2e-4	0.43901 ± 1e-4	0.43865 ± 8e-5	0.43895 ± 1e-4
		DeCORA	**0.43833** ± 6e-5	**0.43815** ± 7e-5	**0.43792** ± 8e-5	**0.43777** ± 1e-4	**0.43749** ± 8e-5
WuKong	AUC ↑	Vanilla	0.81361 ± 9e-5	0.81374 ± 1e-4	0.81380 ± 8e-5	0.81376 ± 2e-4	0.81367 ± 8e-5
		DeCORA	**0.81377** ± 6e-5	**0.81389** ± 8e-5	**0.81400** ± 7e-5	**0.81421** ± 1e-4	**0.81449** ± 8e-5
	Logloss ↓	Vanilla	0.43869 ± 2e-4	0.43885 ± 8e-5	0.43901 ± 9e-5	0.43911 ± 1e-4	0.43902 ± 1e-4
		DeCORA	**0.43840** ± 9e-5	**0.43818** ± 8e-5	**0.43806** ± 6e-5	**0.43781** ± 8e-5	**0.43768** ± 7e-5
DHEN	AUC ↑	Vanilla	0.81301 ± 2e-4	0.81319 ± 9e-5	0.81335 ± 1e-4	0.81347 ± 8e-5	0.81337 ± 1e-4
		DeCORA	**0.81332** ± 9e-5	**0.81346** ± 6e-5	**0.81362** ± 1e-4	**0.81389** ± 7e-5	**0.81419** ± 8e-5
	Logloss ↓	Vanilla	0.43897 ± 8e-5	0.43908 ± 1e-4	0.43913 ± 2e-4	0.43920 ± 9e-5	0.43907 ± 8e-5
		DeCORA	**0.43883** ± 7e-5	**0.43857** ± 8e-5	**0.43830** ± 9e-5	**0.43811** ± 6e-5	**0.43791** ± 8e-5
FAT	AUC ↑	Vanilla	0.81352 ± 8e-5	0.81368 ± 2e-4	0.81375 ± 9e-5	0.81366 ± 1e-4	0.81345 ± 9e-5
		DeCORA	**0.81375** ± 7e-5	**0.81397** ± 9e-5	**0.81412** ± 8e-5	**0.81433** ± 6e-5	**0.81456** ± 8e-5
	Logloss ↓	Vanilla	0.43835 ± 1e-4	0.43877 ± 9e-5	0.43894 ± 8e-5	0.43909 ± 2e-4	0.43915 ± 1e-4
		DeCORA	**0.43799** ± 1e-4	**0.43790** ± 6e-5	**0.43780** ± 7e-5	**0.43754** ± 9e-5	**0.43736** ± 8e-5

A.3 Additional Depth-Method and Resource Comparisons

Tables 9 and 10 report the 4× comparisons for all four backbones. Table 11 supplements the RankMixer and WuKong means in Table 2 with their ten-seed standard deviations, while Table 12 reports the corresponding 16× results for DHEN and FAT. DeCORA achieves the best mean AUC and Logloss in all 16 backbone–dataset–depth configurations. At 16×, its AUC advantage over the strongest competing method ranges from 0.00065 to 0.00122, while its Logloss reduction ranges from 0.00078 to 0.00129. Each mean improvement is at least 8.1 times the larger across-seed standard deviation of DeCORA and the corresponding strongest competitor. This consistently favorable separation relative to run-to-run variability strengthens the evidence for DeCORA at greater depth.

The predictive gains come with moderate training overhead. Across the reported 4× and 16× configurations, DeCORA increases training time per epoch by 8.7%–18.5% and GPU memory by 0.6–1.2 GB relative to vanilla. Its cost remains close to the stronger depth mechanisms; for example, the differences from AttnRes are at most 0.3 minutes per epoch and 0.1 GB at 16×, while DeCORA achieves better mean AUC and Logloss in every corresponding setting.

Table 9: Depth mechanisms at $4\times$ for RankMixer and WuKong. AUC and Logloss are reported as ten-seed means with standard deviations ($\pm$). Mem. is GPU memory in GB, and Time is training time in minutes per epoch. Bold marks the best mean AUC and Logloss.

Backbone	Method	Avazu				Criteo			
		AUC ↑	Logloss ↓	Mem.	Time	AUC ↑	Logloss ↓	Mem.	Time
RankMixer	Vanilla	0.79496 ± 8e−5	0.37086 ± 7e−5	3.2	16.7	0.81374 ± 1e−4	0.43901 ± 7e−5	3.6	20.7
	DenseFormer	0.79475 ± 7e−5	0.37112 ± 6e−5	3.7	17.8	0.81377 ± 8e−5	0.43883 ± 1e−4	4.3	22.1
	mHC	0.79513 ± 8e−5	0.37077 ± 9e−5	3.7	19.1	0.81389 ± 1e−4	0.43839 ± 7e−5	4.3	23.8
	ORU	0.79491 ± 8e−5	0.37071 ± 1e−4	3.5	17.7	0.81365 ± 8e−5	0.43851 ± 9e−5	4.2	22.0
	AttnRes	0.79527 ± 8e−5	0.37069 ± 9e−5	3.8	18.2	0.81379 ± 8e−5	0.43858 ± 9e−5	4.4	22.7
	DeCORA	**0.79579** ± 6e−5	**0.37023** ± 7e−5	3.8	18.2	**0.81425** ± 7e−5	**0.43792** ± 7e−5	4.4	22.9
WuKong	Vanilla	0.79429 ± 8e−5	0.37234 ± 7e−5	3.1	8.9	0.81380 ± 7e−5	0.43901 ± 8e−5	3.4	10.5
	DenseFormer	0.79433 ± 7e−5	0.37242 ± 7e−5	3.7	9.3	0.81374 ± 8e−5	0.43906 ± 8e−5	4.1	11.2
	mHC	0.79457 ± 9e−5	0.37210 ± 8e−5	3.6	10.0	0.81385 ± 1e−4	0.43853 ± 8e−5	4.2	12.1
	ORU	0.79410 ± 6e−5	0.37219 ± 8e−5	3.5	9.4	0.81366 ± 8e−5	0.43889 ± 9e−5	3.9	11.4
	AttnRes	0.79441 ± 7e−5	0.37195 ± 7e−5	3.7	10.1	0.81377 ± 9e−5	0.43912 ± 7e−5	4.3	11.7
	DeCORA	**0.79500** ± 6e−5	**0.37088** ± 7e−5	3.8	10.1	**0.81400** ± 8e−5	**0.43806** ± 7e−5	4.3	11.8

Table 10: Depth mechanisms at $4\times$ for DHEN and FAT. AUC and Logloss are reported as ten-seed means with standard deviations ($\pm$). Mem. is GPU memory in GB, and Time is training time in minutes per epoch. Bold marks the best mean AUC and Logloss.

Backbone	Method	Avazu				Criteo			
		AUC ↑	Logloss ↓	Mem.	Time	AUC ↑	Logloss ↓	Mem.	Time
DHEN	Vanilla	0.79479 ± 8e−5	0.37097 ± 9e−5	3.3	10.4	0.81335 ± 7e−5	0.43913 ± 8e−5	3.6	14.6
	DenseFormer	0.79423 ± 7e−5	0.37145 ± 9e−5	3.7	11.6	0.81310 ± 9e−5	0.43926 ± 9e−5	3.7	15.5
	mHC	0.79527 ± 8e−5	0.37056 ± 9e−5	3.7	12.5	0.81325 ± 7e−5	0.43933 ± 8e−5	3.9	18.9
	ORU	0.79466 ± 8e−5	0.37112 ± 8e−5	3.6	11.6	0.81317 ± 1e−4	0.43917 ± 9e−5	3.7	15.9
	AttnRes	0.79503 ± 9e−5	0.37080 ± 9e−5	3.9	11.9	0.81344 ± 8e−5	0.43892 ± 8e−5	4.2	17.3
	DeCORA	**0.79574** ± 7e−5	**0.37038** ± 7e−5	3.9	12.0	**0.81362** ± 6e−5	**0.43830** ± 8e−5	4.2	17.3
FAT	Vanilla	0.79242 ± 8e−5	0.37229 ± 8e−5	3.4	15.3	0.81375 ± 7e−5	0.43894 ± 9e−5	3.7	19.1
	DenseFormer	0.79214 ± 8e−5	0.37266 ± 9e−5	3.8	16.4	0.81371 ± 9e−5	0.43922 ± 8e−5	4.2	20.2
	mHC	0.79331 ± 8e−5	0.37173 ± 1e−4	3.7	17.9	0.81394 ± 1e−4	0.43871 ± 8e−5	4.2	22.0
	ORU	0.79286 ± 1e−4	0.37210 ± 1e−4	3.6	16.5	0.81345 ± 1e−4	0.43855 ± 8e−5	4.0	20.4
	AttnRes	0.79340 ± 9e−5	0.37192 ± 9e−5	4.0	17.1	0.81360 ± 1e−4	0.43840 ± 9e−5	4.5	21.4
	DeCORA	**0.79447** ± 7e−5	**0.37104** ± 8e−5	4.0	17.2	**0.81412** ± 8e−5	**0.43780** ± 9e−5	4.5	21.6

Table 11: Depth mechanisms at 16× for RankMixer and WuKong. The means reproduce the results in Table 2, with their ten-seed standard deviations (±) added here. Mem. is GPU memory in GB, and Time is training time in minutes per epoch. Bold marks the best mean AUC and Logloss.

Backbone	Method	Avazu				Criteo			
		AUC ↑	Logloss ↓	Mem.	Time	AUC ↑	Logloss ↓	Mem.	Time
RankMixer	Vanilla	0.79443 ± 7e−5	0.37118 ± 8e−5	3.9	39.8	0.81336 ± 8e−5	0.43917 ± 9e−5	4.1	47.2
	DenseFormer	0.79483 ± 7e−5	0.37101 ± 9e−5	4.6	41.5	0.81376 ± 9e−5	0.43873 ± 8e−5	4.8	49.2
	mHC	0.79561 ± 8e−5	0.37083 ± 1e−4	4.7	44.6	0.81423 ± 9e−5	0.43801 ± 8e−5	4.8	52.4
	ORU	0.79501 ± 1e−4	0.37099 ± 9e−5	4.5	42.1	0.81359 ± 8e−5	0.43834 ± 8e−5	4.7	50.5
	AttnRes	0.79552 ± 9e−5	0.37052 ± 8e−5	4.9	43.8	0.81386 ± 8e−5	0.43850 ± 9e−5	5.3	51.0
	DeCORA	**0.79671** ± 7e−5	**0.36957** ± 7e−5	4.9	44.0	**0.81501** ± 7e−5	**0.43723** ± 8e−5	5.3	51.3
WuKong	Vanilla	0.79328 ± 7e−5	0.37222 ± 9e−5	3.7	15.8	0.81344 ± 7e−5	0.43922 ± 8e−5	4.0	23.8
	DenseFormer	0.79387 ± 8e−5	0.37203 ± 9e−5	4.3	16.7	0.81352 ± 8e−5	0.43910 ± 9e−5	4.9	24.6
	mHC	0.79503 ± 7e−5	0.37107 ± 8e−5	4.4	17.1	0.81398 ± 7e−5	0.43832 ± 7e−5	5.0	27.9
	ORU	0.79442 ± 6e−5	0.37139 ± 7e−5	4.3	16.6	0.81370 ± 9e−5	0.43876 ± 7e−5	4.7	25.1
	AttnRes	0.79491 ± 8e−5	0.37098 ± 8e−5	4.6	17.3	0.81381 ± 8e−5	0.43828 ± 9e−5	5.2	27.3
	DeCORA	**0.79590** ± 6e−5	**0.37019** ± 7e−5	4.7	17.3	**0.81463** ± 8e−5	**0.43739** ± 7e−5	5.2	27.5

Table 12: Depth mechanisms at 16× for DHEN and FAT. AUC and Logloss are reported as ten-seed means with standard deviations (±). Mem. is GPU memory in GB, and Time is training time in minutes per epoch. Bold marks the best mean AUC and Logloss.

Backbone	Method	Avazu				Criteo			
		AUC ↑	Logloss ↓	Mem.	Time	AUC ↑	Logloss ↓	Mem.	Time
DHEN	Vanilla	0.79397 ± 7e−5	0.37288 ± 9e−5	3.7	17.1	0.81320 ± 8e−5	0.43942 ± 9e−5	4.1	21.2
	DenseFormer	0.79422 ± 7e−5	0.37223 ± 7e−5	4.1	18.1	0.81309 ± 9e−5	0.43915 ± 8e−5	4.6	22.4
	mHC	0.79493 ± 9e−5	0.37192 ± 9e−5	4.2	19.6	0.81365 ± 8e−5	0.43920 ± 9e−5	4.8	25.7
	ORU	0.79441 ± 9e−5	0.37241 ± 8e−5	4.0	18.3	0.81329 ± 8e−5	0.43845 ± 9e−5	4.6	22.8
	AttnRes	0.79512 ± 9e−5	0.37097 ± 1e−4	4.4	18.9	0.81358 ± 1e−4	0.43872 ± 8e−5	5.0	23.3
	DeCORA	**0.79634** ± 8e−5	**0.36970** ± 7e−5	4.4	19.0	**0.81445** ± 7e−5	**0.43722** ± 8e−5	5.1	23.4
FAT	Vanilla	0.79183 ± 9e−5	0.37296 ± 9e−5	3.9	23.1	0.81323 ± 9e−5	0.43955 ± 8e−5	4.3	26.1
	DenseFormer	0.79231 ± 8e−5	0.37240 ± 9e−5	4.3	25.0	0.81329 ± 1e−4	0.43910 ± 8e−5	4.7	27.5
	mHC	0.79388 ± 8e−5	0.37151 ± 9e−5	4.3	26.7	0.81376 ± 7e−5	0.43819 ± 9e−5	4.7	29.8
	ORU	0.79273 ± 7e−5	0.37209 ± 9e−5	4.1	24.2	0.81343 ± 8e−5	0.43887 ± 8e−5	4.4	27.2
	AttnRes	0.79369 ± 8e−5	0.37138 ± 9e−5	4.5	25.8	0.81362 ± 1e−4	0.43806 ± 9e−5	4.9	28.6
	DeCORA	**0.79479** ± 6e−5	**0.37051** ± 8e−5	4.6	25.8	**0.81483** ± 9e−5	**0.43677** ± 8e−5	5.0	28.8

A.4 Additional Component and Capacity Controls

Tables 13–15 complete the 8× and 16× component results. The full method improves both metrics over every control in all 16 settings. Aggregation only is consistently stronger than vanilla, but does not maintain its AUC gain when extended from 8× to 16×. Loss-only behavior varies across metrics and datasets: its Avazu FAT AUC at 16× is slightly below vanilla, and its Criteo FAT LL at 16× is worse. These exceptions favor the paired design over correlation control alone.

The dense-parameter increases range from 0.1% for DHEN on Criteo to 3.9% for WuKong on Avazu. These relative increments provide a capacity control; matching parameter counts does not equate the computation or inductive bias of widening and depth-prefix aggregation.

Table 13: Component and capacity controls at $8\times$ for RankMixer and WuKong. Abbreviations and ΔP follow Table 3. Bold marks the best AUC and Logloss. Ten-seed std ($\pm$) appears below each mean.

Backbone	Method	Avazu			Criteo		
		AUC ↑	Logloss ↓	ΔP	AUC ↑	Logloss ↓	ΔP
RankMixer	Vanilla	0.79468 ± 8e−5	0.37091 ± 6e−5	0.0%	0.81354 ± 1e−4	0.43895 ± 1e−4	0.0%
	Agg. only	0.79554 ± 6e−5	0.37055 ± 8e−5	3.7%	0.81417 ± 2e−4	0.43805 ± 8e−5	1.2%
	Loss only	0.79492 ± 8e−5	0.37070 ± 7e−5	0.0%	0.81338 ± 9e−5	0.43845 ± 1e−4	0.0%
	Vanilla-W	0.79452 ± 7e−5	0.37081 ± 9e−5	3.7%	0.81366 ± 8e−5	0.43870 ± 2e−4	1.2%
	DeCORA	**0.79636** ± 5e−5	**0.36991** ± 6e−5	3.7%	**0.81481** ± 8e−5	**0.43749** ± 8e−5	1.2%
WuKong	Vanilla	0.79354 ± 8e−5	0.37197 ± 7e−5	0.0%	0.81367 ± 8e−5	0.43902 ± 1e−4	0.0%
	Agg. only	0.79437 ± 1e−4	0.37154 ± 6e−5	3.9%	0.81390 ± 1e−4	0.43850 ± 9e−5	2.1%
	Loss only	0.79373 ± 9e−5	0.37117 ± 5e−5	0.0%	0.81357 ± 2e−4	0.43893 ± 1e−4	0.0%
	Vanilla-W	0.79381 ± 6e−5	0.37179 ± 8e−5	3.9%	0.81317 ± 8e−5	0.43930 ± 8e−5	2.1%
	DeCORA	**0.79568** ± 7e−5	**0.37064** ± 5e−5	3.9%	**0.81449** ± 8e−5	**0.43768** ± 7e−5	2.1%

Table 14: Component and capacity controls at $8\times$ for DHEN and FAT. ΔP and highlighting follow Table 13. Ten-seed std ($\pm$) appears below each mean.

Backbone	Method	Avazu			Criteo		
		AUC ↑	Logloss ↓	ΔP	AUC ↑	Logloss ↓	ΔP
DHEN	Vanilla	0.79411 ± 7e−5	0.37274 ± 7e−5	0.0%	0.81337 ± 1e−4	0.43907 ± 8e−5	0.0%
	Agg. only	0.79505 ± 8e−5	0.37176 ± 5e−5	1.8%	0.81367 ± 8e−5	0.43875 ± 1e−4	0.1%
	Loss only	0.79451 ± 8e−5	0.37202 ± 8e−5	0.0%	0.81320 ± 1e−4	0.43933 ± 9e−5	0.0%
	Vanilla-W	0.79434 ± 5e−5	0.37210 ± 9e−5	1.8%	0.81305 ± 9e−5	0.43950 ± 1e−4	0.1%
	DeCORA	**0.79619** ± 7e−5	**0.37002** ± 5e−5	1.8%	**0.81419** ± 8e−5	**0.43791** ± 8e−5	0.1%
FAT	Vanilla	0.79205 ± 1e−4	0.37272 ± 7e−5	0.0%	0.81345 ± 9e−5	0.43915 ± 1e−4	0.0%
	Agg. only	0.79331 ± 7e−5	0.37182 ± 8e−5	2.4%	0.81388 ± 9e−5	0.43867 ± 8e−5	1.3%
	Loss only	0.79242 ± 7e−5	0.37190 ± 6e−5	0.0%	0.81322 ± 2e−4	0.43902 ± 9e−5	0.0%
	Vanilla-W	0.79183 ± 6e−5	0.37290 ± 7e−5	2.4%	0.81317 ± 1e−4	0.43910 ± 2e−4	1.3%
	DeCORA	**0.79464** ± 7e−5	**0.37087** ± 7e−5	2.4%	**0.81456** ± 8e−5	**0.43736** ± 8e−5	1.3%

Table 15: Component and capacity controls at $16\times$ for DHEN and FAT. ΔP and highlighting follow Table 13. Ten-seed std ($\pm$) appears below each mean.

Backbone	Method	Avazu			Criteo		
		AUC ↑	Logloss ↓	ΔP	AUC ↑	Logloss ↓	ΔP
DHEN	Vanilla	0.79397 ± 8e−5	0.37288 ± 6e−5	0.0%	0.81320 ± 9e−5	0.43942 ± 8e−5	0.0%
	Agg. only	0.79490 ± 6e−5	0.37137 ± 7e−5	1.8%	0.81365 ± 1e−4	0.43820 ± 1e−4	0.1%
	Loss only	0.79421 ± 7e−5	0.37202 ± 6e−5	0.0%	0.81309 ± 8e−5	0.43920 ± 2e−4	0.0%
	Vanilla-W	0.79355 ± 8e−5	0.37342 ± 7e−5	1.8%	0.81312 ± 1e−4	0.43925 ± 8e−5	0.1%
	DeCORA	**0.79634** ± 4e−5	**0.36970** ± 6e−5	1.8%	**0.81445** ± 8e−5	**0.43722** ± 9e−5	0.1%
FAT	Vanilla	0.79183 ± 6e−5	0.37296 ± 8e−5	0.0%	0.81323 ± 1e−4	0.43955 ± 9e−5	0.0%
	Agg. only	0.79315 ± 5e−5	0.37164 ± 8e−5	2.4%	0.81382 ± 1e−4	0.43813 ± 1e−4	1.3%
	Loss only	0.79180 ± 8e−5	0.37275 ± 5e−5	0.0%	0.81314 ± 1e−4	0.43980 ± 8e−5	0.0%
	Vanilla-W	0.79209 ± 9e−5	0.37270 ± 8e−5	2.4%	0.81303 ± 8e−5	0.43995 ± 1e−4	1.3%
	DeCORA	**0.79479** ± 7e−5	**0.37051** ± 5e−5	2.4%	**0.81483** ± 8e−5	**0.43677** ± 8e−5	1.3%

A.5 Complementary Representation Diagnostics

Figures 4–6 complete the three-statistic comparisons, and Figures 7–10 show DCO for all component variants. All diagnostics are at $8\times$ and are reported as means over ten seeds. Source zero denotes the embedding; sources 1–8 denote the displayed hidden representations. PAC is shown for prefixes 1–8. Every IKR curve starts at the corrected empty-prefix reference of one.

Across backbones and datasets, the full method consistently exhibits lower DCO and higher IKR than vanilla and the ablation variants, matching the geometric behavior motivated by our design and analysis. Its PAC remains comparable to the other variants and sufficiently high, consistent with its role in measuring aggregate–prefix alignment rather than a quantity that should be maximized. In contrast, the relative ordering among individual ablation variants can vary across sources and depths. For example, aggregation only has larger IKR than other ablations at several intermediate sources for DHEN on Criteo, while the relative PAC ordering among variants can change with aggregation depth for WuKong. Overall, the full method retains the expected geometric pattern despite these variations among its ablations.

For a given backbone, every DCO map uses the same color scale across datasets and variants, while different backbones may use different scales. Diagonal cells are masked because the comparison concerns cross-depth overlap. Legends identify IKR/PAC curves; captions identify heatmaps.

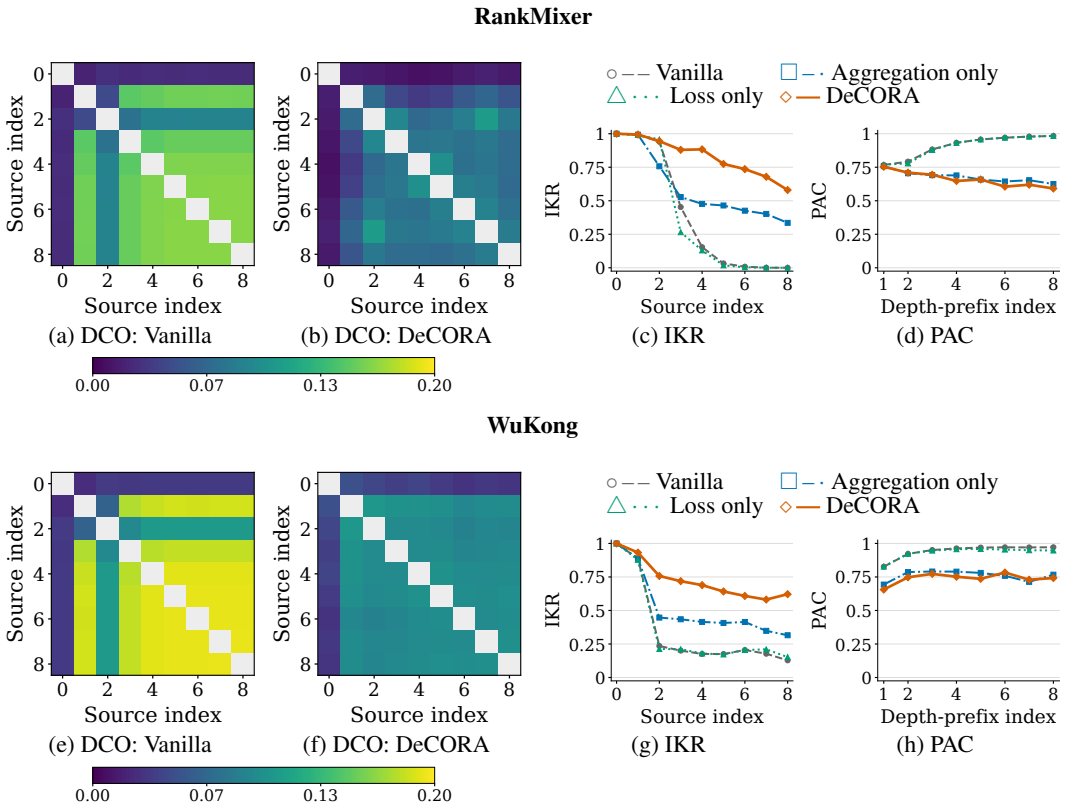


Figure 4: RankMixer and WuKong on Criteo at $8\times$. Layout and diagnostic conventions follow Figure 3.

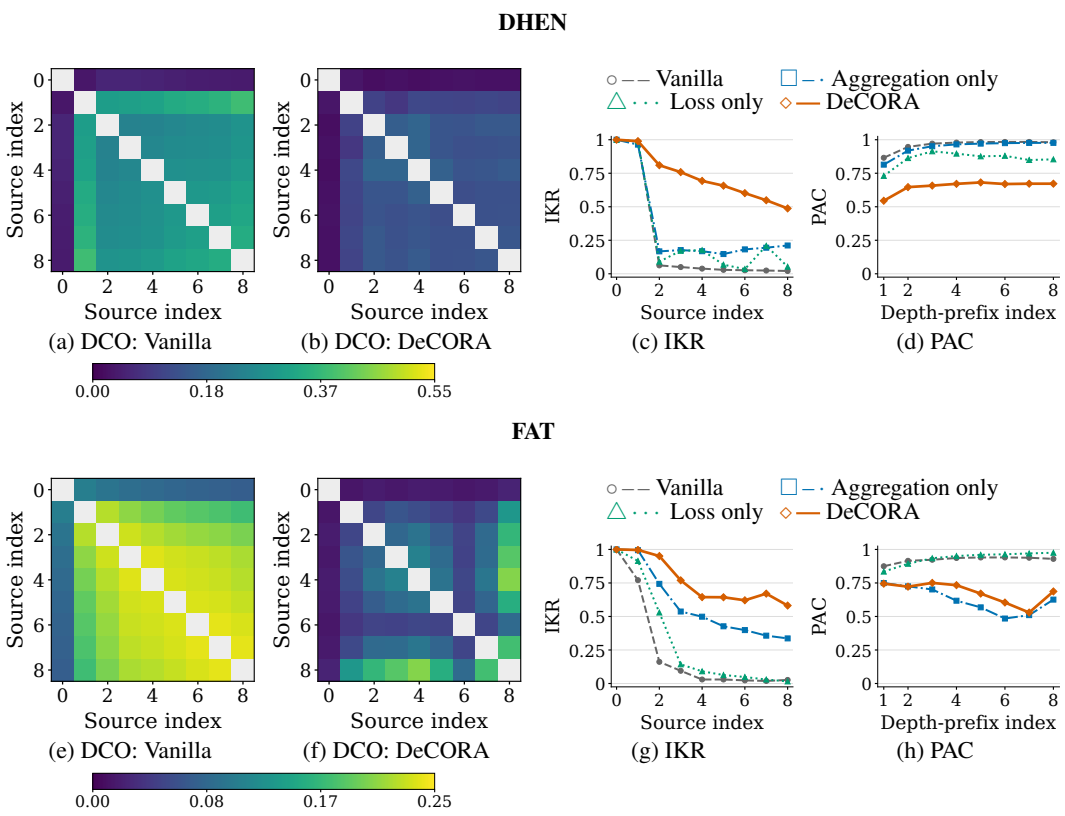


Figure 5: DHEN and FAT on Avazu at 8×. Layout and diagnostic conventions follow Figure 3.

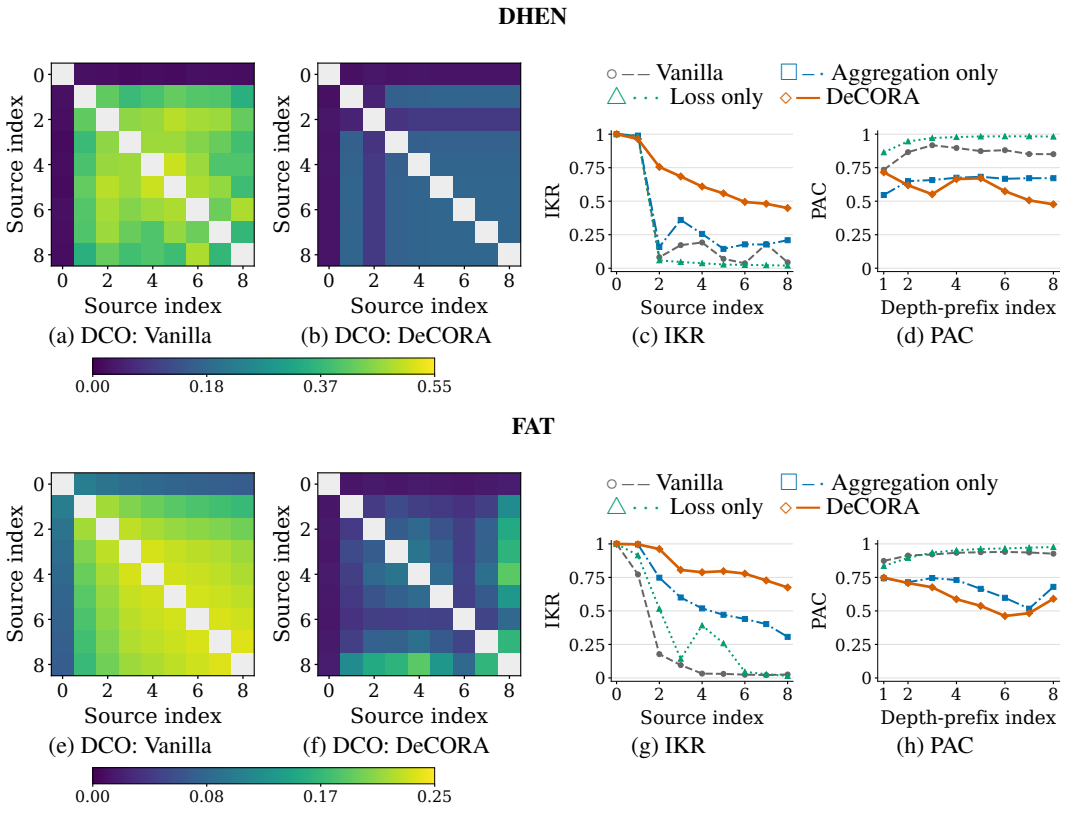


Figure 6: DHEN and FAT on Criteo at 8×. Layout and diagnostic conventions follow Figure 3.

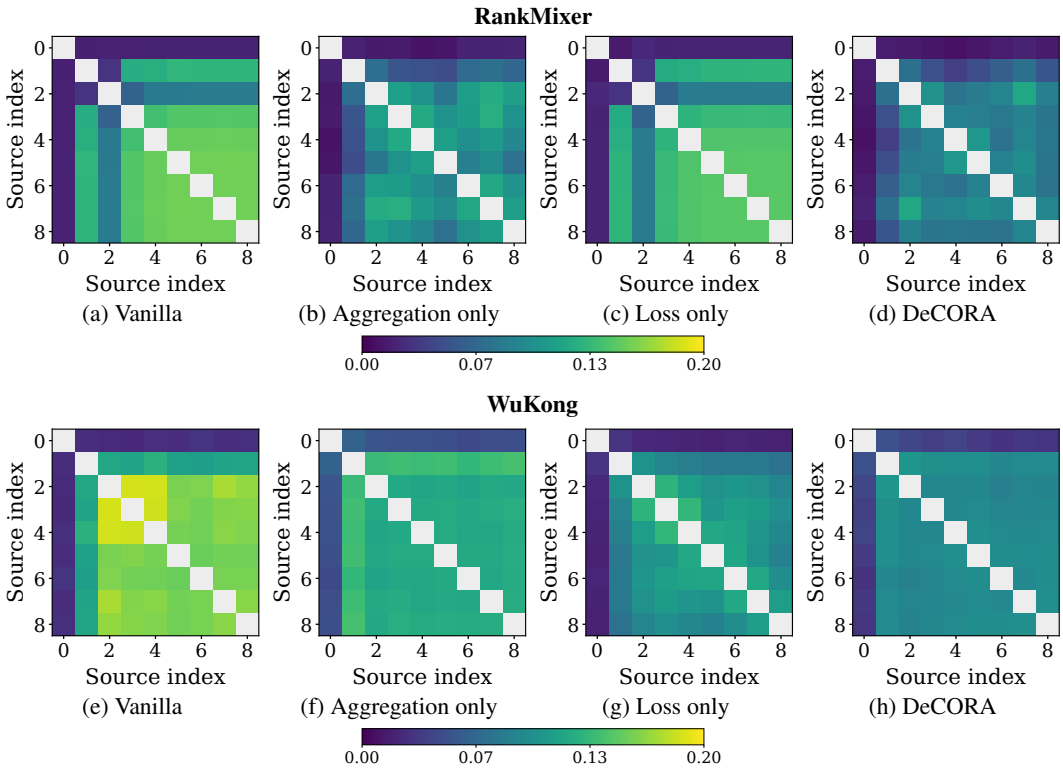


Figure 7: DCO for all variants of RankMixer and WuKong on Avazu at $8\times$. Each row uses a common color scale; gray diagonal cells are masked.

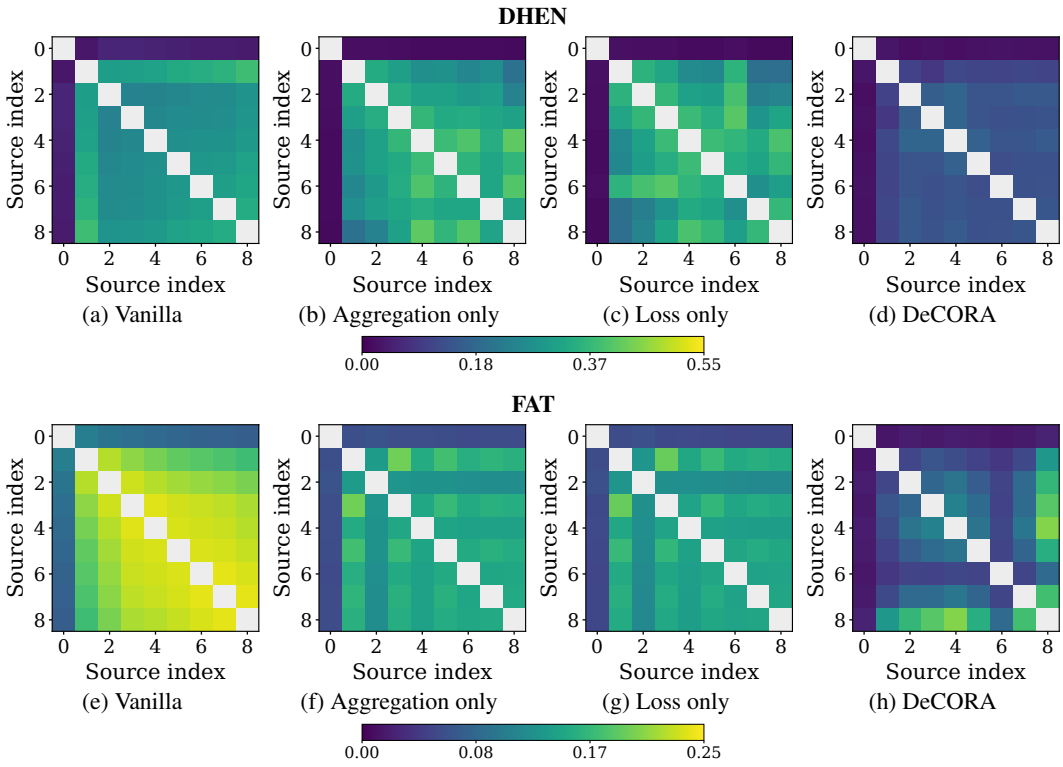


Figure 8: DCO for all variants of DHEN and FAT on Avazu at $8\times$. Each row uses a common color scale; gray diagonal cells are masked.

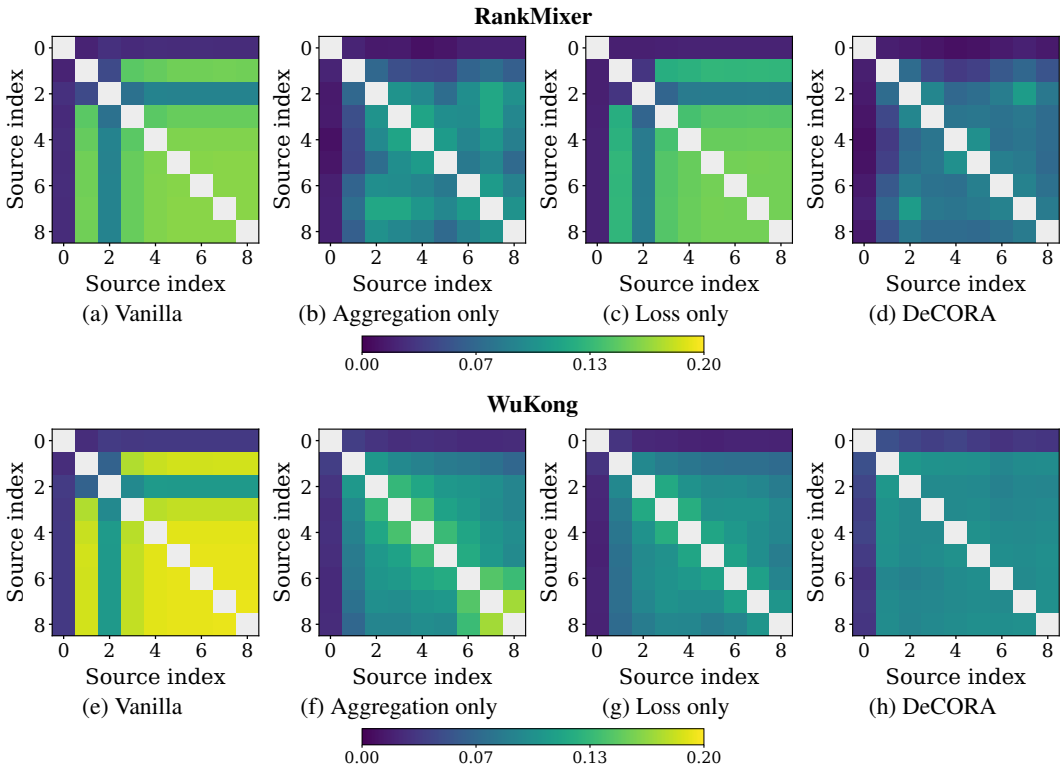


Figure 9: DCO for all variants of RankMixer and WuKong on Criteo at $8\times$. Scales match the corresponding Avazu maps.

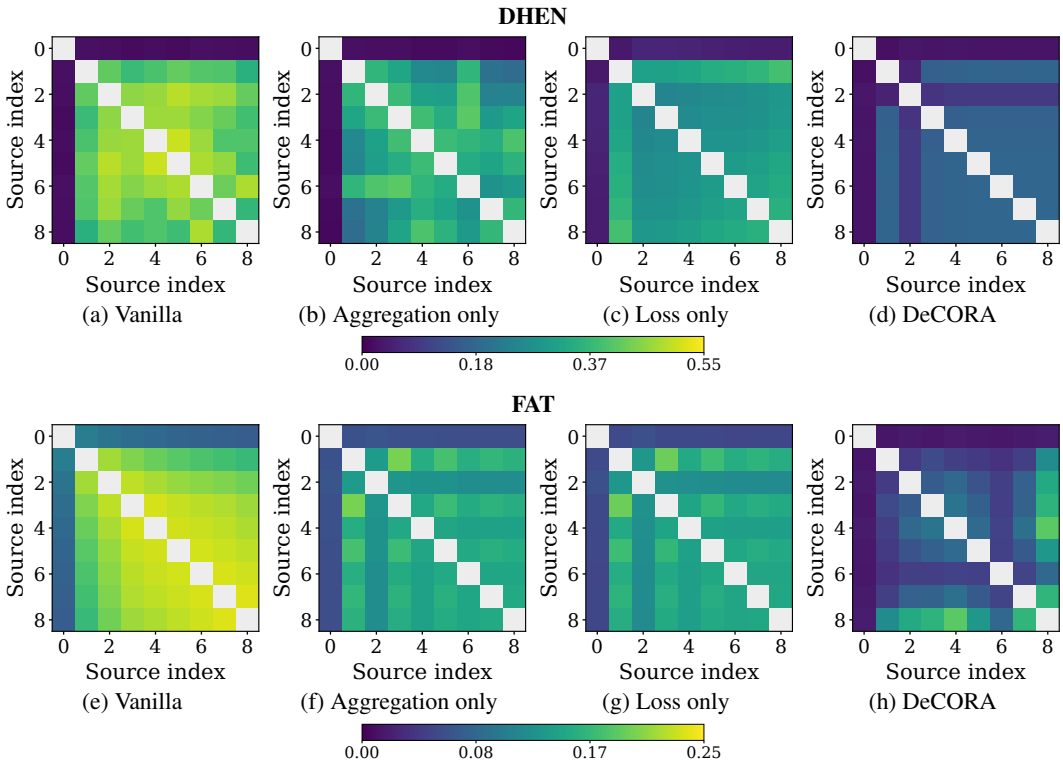


Figure 10: DCO for all variants of DHEN and FAT on Criteo at $8\times$. Scales match the corresponding Avazu maps.

B Mathematical Support for the Design and Diagnostics

The analysis makes the objective and diagnostics precise using covariance, orthogonal projection, and matrix identities. Population statements fix a token position and the distribution of examples valid at that position; we suppress the token index. Internal token mixing does not change this observation convention. Averaging tokenwise statistics does not require independence among tokens, but also does not establish a claim about arbitrary cross-token functions. The kernel propositions are finite-sample identities. Throughout, $\|\cdot\|_2$ denotes the Euclidean or operator norm as appropriate, and $\|\cdot\|_F$ the Frobenius norm.

B.1 Coordinate Correlation and Linear-Function Overlap

Let $\boldsymbol{U}_\ell$ be square integrable, $\boldsymbol{X}_\ell = \boldsymbol{U}_\ell - \mathbb{E}[\boldsymbol{U}_\ell]$, and $\boldsymbol{Z}_\ell = \boldsymbol{D}_\ell^{-1}\boldsymbol{X}_\ell$, where $\boldsymbol{D}_\ell$ is deterministic, positive diagonal, and invertible. Population standard deviations plus $\varepsilon > 0$ satisfy this convention, with constant coordinates contributing zero. Define

$$\boldsymbol{C}_{\ell m} = \mathbb{E}[\boldsymbol{Z}_\ell \boldsymbol{Z}_m^\top], \qquad \mathcal{S}_\ell = \{\boldsymbol{a}^\top \boldsymbol{Z}_\ell(\cdot) : \boldsymbol{a} \in \mathbb{R}^d\} = \{\boldsymbol{a}^\top \boldsymbol{X}_\ell(\cdot) : \boldsymbol{a} \in \mathbb{R}^d\} \subset L_2(P_X). \tag{14}$$

Coordinate standardization is not whitening: the within-source covariance need not be the identity.

Theorem 1 (Cross-correlation and functional overlap)**.** *For any two depths,*

$$\|\boldsymbol{C}_{\ell m}\|_F^2 = \sum_{i,j} \langle Z_{\ell i}, Z_{mj}\rangle_{L_2}^2, \qquad \boldsymbol{C}_{\ell m} = 0 \iff \mathcal{S}_\ell \perp \mathcal{S}_m. \tag{15}$$

For $f = \boldsymbol{a}^\top \boldsymbol{Z}_\ell$ and $g = \boldsymbol{b}^\top \boldsymbol{Z}_m$,

$$|\langle f, g\rangle_{L_2}| \le \|\boldsymbol{C}_{\ell m}\|_2 \|\boldsymbol{a}\|_2 \|\boldsymbol{b}\|_2 \le \|\boldsymbol{C}_{\ell m}\|_F \|\boldsymbol{a}\|_2 \|\boldsymbol{b}\|_2. \tag{16}$$

Proof. Centering gives $[\boldsymbol{C}_{\ell m}]_{ij} = \mathbb{E}[Z_{\ell i} Z_{mj}] = \langle Z_{\ell i}, Z_{mj}\rangle_{L_2}$. Squaring and summing proves the identity. Bilinearity gives $\langle f, g\rangle = \boldsymbol{a}^\top \boldsymbol{C}_{\ell m} \boldsymbol{b}$. If the matrix is zero, all pairs of functions are orthogonal; conversely, orthogonality applied to the coordinate functions makes every matrix entry zero. The norm bounds follow from Cauchy–Schwarz and $\|\boldsymbol{C}\|_2 \le \|\boldsymbol{C}\|_F$. □

For centered empirical matrices, the exact identity

$$\|\widehat{\boldsymbol{C}}_{\ell m}\|_F^2 = \frac{\mathrm{tr}[(\boldsymbol{Z}_\ell \boldsymbol{Z}_\ell^\top)(\boldsymbol{Z}_m \boldsymbol{Z}_m^\top)]}{(n_t - 1)^2} \tag{17}$$

connects the objective to linear-kernel HSIC (Gretton et al., 2005). The empirical squared statistic need not be an unbiased population estimator. DCO's diagonal records within-source coordinate correlations and need not equal one; the heatmaps mask it to emphasize cross-depth comparisons. Zero cross-correlation also admits constant representations, which is one reason that its interaction with the task objective matters.

Corollary 1 (Conditioning-dependent overlap bounds)**.** *Let $\boldsymbol{\Sigma}_{\ell\ell} = \mathbb{E}[\boldsymbol{Z}_\ell \boldsymbol{Z}_\ell^\top] = \boldsymbol{O}_\ell \boldsymbol{\Lambda}_\ell \boldsymbol{O}_\ell^\top$ be its positive-eigenvalue decomposition. If the positive eigenvalues are at least $\kappa_\ell > 0$, and likewise for m, then the cross-covariance of the support-whitened coordinates satisfies*

$$\widetilde{\boldsymbol{C}}_{\ell m} = \boldsymbol{\Lambda}_\ell^{-1/2} \boldsymbol{O}_\ell^\top \boldsymbol{C}_{\ell m} \boldsymbol{O}_m \boldsymbol{\Lambda}_m^{-1/2}, \qquad \|\widetilde{\boldsymbol{C}}_{\ell m}\|_F \le \frac{\|\boldsymbol{C}_{\ell m}\|_F}{\sqrt{\kappa_\ell \kappa_m}}. \tag{18}$$

Its singular values are the canonical correlations between the source function spaces. For the prefix stack $\boldsymbol{Z}_{<\ell}$, write $\boldsymbol{\Sigma}_{<\ell} = \mathrm{Cov}(\boldsymbol{Z}_{<\ell})$ and $\boldsymbol{C}_{\ell,<\ell} = \mathbb{E}[\boldsymbol{Z}_\ell \boldsymbol{Z}_{<\ell}^\top]$. If the positive prefix eigenvalues are at least $\gamma > 0$, the energy of the optimal linear prediction of $\boldsymbol{Z}_\ell$ from its prefix obeys

$$\mathrm{tr}(\boldsymbol{C}_{\ell,<\ell} \boldsymbol{\Sigma}_{<\ell}^\dagger \boldsymbol{C}_{\ell,<\ell}^\top) \le \gamma^{-1} \sum_{k<\ell} \|\boldsymbol{C}_{\ell k}\|_F^2. \tag{19}$$

Proof. Support whitening produces an orthonormal coordinate basis for each scalar function space; zero covariance directions have zero L_2 norm. The cross-Gram matrix between these bases is $\widetilde{\boldsymbol{C}}_{\ell m}$, whose singular values are their canonical correlations. The Frobenius bound follows from

$\|ABC\|_F \le \|A\|_2\|B\|_F\|C\|_2$. For the prefix statement, abbreviate $\boldsymbol{C} = \boldsymbol{C}_{\ell,<\ell}$ and $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}_{<\ell}$. Every $\boldsymbol{v} \in \ker \boldsymbol{\Sigma}$ satisfies $\boldsymbol{v}^\top \boldsymbol{Z}_{<\ell} = 0$ almost surely, hence $\boldsymbol{C}\boldsymbol{v} = 0$. The normal equations are therefore solved by $\boldsymbol{A}_* = \boldsymbol{C}\boldsymbol{\Sigma}^\dagger$. Completing the square in $\mathbb{E}\|\boldsymbol{Z}_\ell - \boldsymbol{A}\boldsymbol{Z}_{<\ell}\|_2^2$ shows that its explained energy is $\mathrm{tr}(\boldsymbol{C}\boldsymbol{\Sigma}^\dagger\boldsymbol{C}^\top)$. On the positive support, $\boldsymbol{\Sigma}^\dagger \preceq \gamma^{-1}\boldsymbol{I}$; the bound follows from $\|\boldsymbol{C}\|_F^2 = \sum_{k<\ell}\|\boldsymbol{C}_{\ell k}\|_F^2$. If the prefix covariance is zero, both sides vanish. □

The conditioning assumptions distinguish coordinate overlap from subspace overlap: ill-conditioned features can have small coordinate correlations but large correlations between normalized linear combinations. A permanently masked star controls only embedding–hidden pairs and cannot replace the full prefix sum in Equation (19). Our full graph gives every pair positive weight at finite logits.

From the learned objective to measured overlap. Write $D_{\ell m} = \mathrm{DCO}_{\ell m}$ and $\overline{D} = \sum_{(\ell,m)\in\mathcal{E}_{\mathrm{pair}}} \pi_{\ell m} D_{\ell m} = \mathcal{L}_{\mathrm{dec}}$. The global softmax makes $\overline{D}$ a convex combination of nonnegative overlaps, hence $0 \le \min_{(\ell,m)} D_{\ell m} \le \overline{D} \le \max_{(\ell,m)} D_{\ell m}$. For any fixed finite logits, $\pi_{\min} = \min_{(\ell,m)} \pi_{\ell m} > 0$ also gives

$$\sum_{(\ell,m)\in\mathcal{E}_{\mathrm{pair}}} D_{\ell m} \le \frac{\mathcal{L}_{\mathrm{dec}}}{\pi_{\min}}. \tag{20}$$

The smallest pair weight links total overlap to the loss. Initialization favors embedding–hidden pairs while retaining hidden–hidden pairs: with $N_h = L(L-1)/2$, $\pi_{0m}^{(0)} = e^2/(Le^2 + N_h)$ and $\pi_{\ell m}^{(0)} = 1/(Le^2 + N_h)$ for $1 \le \ell < m \le L$. These are relative priorities, not binary weights.

For fixed representations, differentiating through softmax yields

$$\frac{\partial \mathcal{L}}{\partial \alpha_{\ell m}} = \pi_{\ell m}(D_{\ell m} - \overline{D}). \tag{21}$$

The logits can therefore reduce the objective by shifting weight toward pairs with below-average overlap, alongside the network's changes to the representations. No fixed lower bound on $\pi_{\min}$ is imposed, so Equation (20) need not be tight. Reporting unweighted DCO directly tests source overlap independently of this redistribution. Corollary 1 gives the conditioning needed to interpret population overlaps as subspace overlap; task metrics and prefix-access analysis assess predictive utility.

B.2 Kernel Residual and Depth-Prefix Alignment

At each token, use the same ordered valid examples for every source and aggregate. Let $\boldsymbol{X}_\ell$ be the batch-centered data matrix, $\boldsymbol{K}_\ell = \boldsymbol{X}_\ell\boldsymbol{X}_\ell^\top$, and $\boldsymbol{G}_\ell = \boldsymbol{K}_\ell/\|\boldsymbol{K}_\ell\|_F$ for a nonzero kernel. A zero kernel is assigned $\boldsymbol{G}_\ell = 0$ and adds no direction to a prefix span. The following statements concern exact arithmetic; numerical pseudoinverse thresholds approximate the projection.

Proposition 1 (Incremental Kernel Residual)**.** *Let $\Pi_{\ell-1}$ project onto $\mathrm{span}\{\boldsymbol{G}_0, \ldots, \boldsymbol{G}_{\ell-1}\}$ in Frobenius inner product. For a nonzero current kernel,*

$$0 \le \|(I - \Pi_{\ell-1})\boldsymbol{G}_\ell\|_F^2 \le 1. \tag{22}$$

The residual is zero exactly when the current kernel belongs to the prefix span and one exactly when it is orthogonal to that span. If $M_{ij} = \langle \boldsymbol{G}_i, \boldsymbol{G}_j\rangle_F$ and $c_i = \langle \boldsymbol{G}_i, \boldsymbol{G}_\ell\rangle_F$ for $i, j < \ell$, then

$$\|(I - \Pi_{\ell-1})\boldsymbol{G}_\ell\|_F^2 = \|\boldsymbol{G}_\ell\|_F^2 - \boldsymbol{c}^\top\boldsymbol{M}^\dagger\boldsymbol{c}. \tag{23}$$

Proof. Pythagoras gives $\|\boldsymbol{G}_\ell\|_F^2 = \|\Pi_{\ell-1}\boldsymbol{G}_\ell\|_F^2 + \|(I - \Pi_{\ell-1})\boldsymbol{G}_\ell\|_F^2$. For a normalized nonzero kernel the left side is one, proving the bounds and equality conditions. Write $\mathcal{F}\boldsymbol{a} = \sum_{i<\ell} a_i \boldsymbol{G}_i$. Then $\mathcal{F}^*\mathcal{F} = \boldsymbol{M}$ and $\mathcal{F}^*\boldsymbol{G}_\ell = \boldsymbol{c}$. If $\boldsymbol{a} \in \ker \boldsymbol{M}$, then $\mathcal{F}\boldsymbol{a} = 0$ and $\boldsymbol{a}^\top\boldsymbol{c} = 0$, so $\boldsymbol{c}$ lies in the range of $\boldsymbol{M}$. Thus $\mathcal{F}\boldsymbol{M}^\dagger\boldsymbol{c}$ is the orthogonal projection and its squared norm is $\boldsymbol{c}^\top\boldsymbol{M}^\dagger\boldsymbol{c}$. □

For $\ell > 0$, a zero kernel has zero residual. At source zero, a nonzero kernel has residual one against the empty prefix. We plot $\mathrm{IKR}_0 = 1$ as a common reference, not evidence of a difference between variants. With degenerate source-zero kernels, this reference differs from the token average, which assigns zero to each zero kernel.

Proposition 2 (Prefix Aggregate Coverage). *Set $\boldsymbol{Q}_s = \sum_{k=0}^{s} \boldsymbol{G}_k$ and let $\boldsymbol{K}_{A_s}$ be the centered aggregate kernel. When both are nonzero, tokenwise PAC is*

$$p_s = \left\langle \frac{\boldsymbol{K}_{A_s}}{\|\boldsymbol{K}_{A_s}\|_F}, \frac{\boldsymbol{Q}_s}{\|\boldsymbol{Q}_s\|_F} \right\rangle_F \in [0, 1]. \tag{24}$$

It equals one exactly when the two unnormalized matrices are positive scalar multiples. With $a_k = \langle \boldsymbol{G}_{A_s}, \boldsymbol{G}_k \rangle_F$ and $M_{ij} = \langle \boldsymbol{G}_i, \boldsymbol{G}_j \rangle_F$ for $i, j \le s$,

$$p_s = \frac{\sum_{k=0}^{s} a_k}{\sqrt{\sum_{i,j=0}^{s} M_{ij}}}. \tag{25}$$

If the aggregate kernel or prefix sum is zero, assign PAC zero.

Proof. Centered linear kernels and their nonnegative sums are positive semidefinite. Consequently, $\mathrm{tr}(\boldsymbol{K}\boldsymbol{Q}) = \mathrm{tr}(\boldsymbol{K}^{1/2}\boldsymbol{Q}\boldsymbol{K}^{1/2}) \ge 0$. Cauchy–Schwarz bounds the inner product between unit-Frobenius matrices by one, with equality exactly when they coincide. Expanding $\|\boldsymbol{Q}_s\|_F^2$ and $\langle \boldsymbol{G}_{A_s}, \boldsymbol{Q}_s \rangle_F$ gives the formula. □

The identity $\langle \boldsymbol{X}\boldsymbol{X}^\top, \boldsymbol{Y}\boldsymbol{Y}^\top \rangle_F = \|\boldsymbol{X}^\top \boldsymbol{Y}\|_F^2$ permits both statistics to be computed without storing sample-by-sample kernels, and identifies normalized kernel inner products with linear CKA (Kornblith et al., 2019). An invertible non-orthogonal feature transformation can change a linear kernel without changing its scalar feature span. IKR therefore differs from the innovation space used below. Likewise, the kernel of an average contains cross-source terms, while PAC's reference sums separately normalized kernels. PAC neither measures source recovery nor bounds the aggregation gap.

C Conditional Analysis of Aggregation and Prediction

For fixed parameters and deterministic measurable evaluation maps, every hidden representation and aggregate remains a function of $\boldsymbol{E}$. This follows by induction in Equations (1) and (2). The analysis concerns the organization of functions of the input, not the creation of new information about the input. We use the same fixed-token population convention as Appendix B. The affine results allow a general output width p, specializing to $p = d$ for Equation (3). The coherence bound concerns the transformed sources; covariance transport and source-space inclusion additionally require an affine map. These are conditional properties of the representations, rather than guarantees for nonconvex training.

C.1 Shared Affine Maps

Theorem 2 (Affine covariance transport and aggregate access). *Fix an aggregation depth s and let $f_s(\boldsymbol{u}) = \boldsymbol{W}_s\boldsymbol{u} + \boldsymbol{b}_s$, with $\boldsymbol{W}_s \in \mathbb{R}^{p\times d}$. Define $\boldsymbol{Y}_{s\leftarrow k} = \boldsymbol{W}_s\boldsymbol{X}_k$ and $\mathcal{T}_{s\leftarrow k} = \{\boldsymbol{a}^\top \boldsymbol{Y}_{s\leftarrow k} : \boldsymbol{a} \in \mathbb{R}^p\}$. Then*

$$\boldsymbol{\Gamma}_{km}^{[s]} = \mathbb{E}[\boldsymbol{Y}_{s\leftarrow k}\boldsymbol{Y}_{s\leftarrow m}^\top] = \boldsymbol{W}_s\boldsymbol{D}_k\boldsymbol{C}_{km}\boldsymbol{D}_m\boldsymbol{W}_s^\top, \tag{26}$$

$$\|\boldsymbol{\Gamma}_{km}^{[s]}\|_F \le \|\boldsymbol{W}_s\|_2^2 \|\boldsymbol{D}_k\|_2 \|\boldsymbol{D}_m\|_2 \|\boldsymbol{C}_{km}\|_F. \tag{27}$$

Moreover, $\mathcal{T}_{s\leftarrow k} \subseteq \mathcal{S}_k$, with equality if $\boldsymbol{W}_s$ has full column rank. Exact source-space orthogonality is preserved. The centered aggregate satisfies

$$\overline{\boldsymbol{A}}_s = \frac{1}{s+1}\sum_{k=0}^{s} \boldsymbol{Y}_{s\leftarrow k}, \qquad \mathcal{A}_s := \{\boldsymbol{a}^\top \overline{\boldsymbol{A}}_s : \boldsymbol{a} \in \mathbb{R}^p\} \subseteq \sum_{k=0}^{s} \mathcal{T}_{s\leftarrow k} \subseteq \mathcal{V}_s. \tag{28}$$

Proof. Centering removes the shared bias, giving $\boldsymbol{Y}_{s\leftarrow k} = \boldsymbol{W}_s\boldsymbol{D}_k\boldsymbol{Z}_k$. Taking its cross-covariance proves Equation (26), and submultiplicativity proves the norm bound. Every transformed coordinate combination has the form $(\boldsymbol{W}_s^\top \boldsymbol{a})^\top \boldsymbol{X}_k$, so it belongs to $\mathcal{S}_k$. When $\boldsymbol{W}_s$ has full column rank, $\boldsymbol{W}_s^\top$ is surjective, which proves equality. Orthogonality is preserved by subspace inclusion. Finally, centering the source average and applying any common readout vector expresses the aggregate function as a sum of transformed source functions, proving both inclusions. □

Sharing an affine map also gives $\frac{1}{s+1}\sum_k f_s(\boldsymbol{U}_k) = f_s(\frac{1}{s+1}\sum_k \boldsymbol{U}_k)$: the bias appears once after averaging. A running sum therefore suffices to compute the aggregation path with one adapter application per depth. This implementation shortcut does not hold for a general nonlinear map, and the training regularizer can still require access to individual sources.

C.2 Conditional Non-Cancellation

Theorem 3 (A coherence bound for depth-prefix averaging)**.** *Let $\mathfrak{H}_p = L_2(P_X; \mathbb{R}^p)$. At aggregation depth s, let $\mathfrak{T}_k$ be the vector-valued space whose coordinates lie in the scalar span of the centered transformed source $\boldsymbol{V}_{s\leftarrow k} - \mathbb{E}[\boldsymbol{V}_{s\leftarrow k}]$. Define their maximal pairwise coherence as*

$$\mu_s = \max_{k\neq m} \sup_{\boldsymbol{u}\in\mathfrak{T}_k\setminus\{0\},\, \boldsymbol{v}\in\mathfrak{T}_m\setminus\{0\}} \frac{|\langle \boldsymbol{u}, \boldsymbol{v}\rangle_{\mathfrak{H}_p}|}{\|\boldsymbol{u}\|_{\mathfrak{H}_p}\|\boldsymbol{v}\|_{\mathfrak{H}_p}}, \tag{29}$$

assigning coherence zero when either space is zero. For arbitrary $\boldsymbol{v}_k \in \mathfrak{T}_k$,

$$\frac{1-s\mu_s}{(s+1)^2}\sum_{k=0}^{s}\|\boldsymbol{v}_k\|^2 \le \left\|\frac{1}{s+1}\sum_{k=0}^{s}\boldsymbol{v}_k\right\|^2 \le \frac{1+s\mu_s}{(s+1)^2}\sum_{k=0}^{s}\|\boldsymbol{v}_k\|^2. \tag{30}$$

If $\mu_s < 1/s$, the summation map on $\mathfrak{T}_0 \times \cdots \times \mathfrak{T}_s$ is injective.

Proof. Expanding the squared norm gives the sum of individual energies plus $2\sum_{k<m}\langle \boldsymbol{v}_k, \boldsymbol{v}_m\rangle$. Writing $b_k = \|\boldsymbol{v}_k\|$, the absolute cross-term is at most $2\mu_s\sum_{k<m} b_k b_m$. Since $(\sum_k b_k)^2 \le (s+1)\sum_k b_k^2$, $2\sum_{k<m} b_k b_m \le s\sum_k b_k^2$. Adding and subtracting this bound, then dividing by $(s+1)^2$, proves the inequalities. If the sum is zero and $1 - s\mu_s > 0$, the lower bound forces every $\boldsymbol{v}_k$ to be zero, proving injectivity. □

Exact source orthogonality under affine maps implies $\mu_s = 0$. More generally, normalized scalar overlap bounds can bound vector coherence by applying Cauchy–Schwarz across coordinates. Small averaged DCO alone does not establish $\mu_s < 1/s$. The injectivity statement concerns functions over the observation distribution; it does not give pointwise recovery of every source from one example's aggregate. PAC is not a measure of this coherence.

C.3 Population Readout over Fixed Representations

Fix the representations before optimizing the readout, and assume square-integrable sources throughout this analysis. Let $Y \in L_2$, $\eta(X) = \mathbb{E}[Y \mid X]$, $\mathcal{V}_\ell = \sum_{k=0}^{\ell}\mathcal{S}_k$, and $\mathcal{V}_{-1} = \{0\}$. These finite-dimensional subspaces are closed. Linear access to the complete prefix is an idealization of what the fixed-width aggregate makes available.

Theorem 4 (Predictive gain of depth-prefix innovation)**.** *Set $\mathcal{I}_\ell = (I - P_{\mathcal{V}_{\ell-1}})\mathcal{S}_\ell$ and $\mathcal{R}^\star_\ell = \inf_{f\in\mathcal{V}_\ell}\mathbb{E}[(Y - f(X))^2]$. Then $\mathcal{V}_\ell = \mathcal{V}_{\ell-1}\oplus\mathcal{I}_\ell$ and*

$$\mathcal{R}^\star_{\ell-1} - \mathcal{R}^\star_\ell = \|P_{\mathcal{I}_\ell}\eta\|^2_{L_2}. \tag{31}$$

If $\mathcal{S}_\ell \perp \mathcal{V}_{\ell-1}$, then $\mathcal{I}_\ell = \mathcal{S}_\ell$, but the gain may still vanish if the new space is orthogonal to η.

Proof. For any square-integrable $f(X)$, $\mathbb{E}[(Y - f(X))^2] = \mathbb{E}[\mathrm{Var}(Y \mid X)] + \|\eta - f\|^2_{L_2}$. Thus the optimal readout is $P_{\mathcal{V}_\ell}\eta$. Decompose every $u \in \mathcal{S}_\ell$ as $P_{\mathcal{V}_{\ell-1}}u + (I - P_{\mathcal{V}_{\ell-1}})u$. This proves that $\mathcal{V}_\ell$ equals the orthogonal sum in the statement. Consequently, $P_{\mathcal{V}_\ell}\eta = P_{\mathcal{V}_{\ell-1}}\eta + P_{\mathcal{I}_\ell}\eta$, and Pythagoras gives the risk difference. □

Corollary 2 (Aggregation exposure gap)**.** *For a fixed affine adapter f_s, let $\mathcal{A}_s$ be the linear-readout space in Equation* (28)*. Then*

$$\inf_{f\in\mathcal{A}_s}\mathbb{E}[(Y - f(X))^2] - \mathcal{R}^\star_s = \|P_{\mathcal{V}_s}\eta - P_{\mathcal{A}_s}\eta\|^2_{L_2} \ge 0. \tag{32}$$

Proof. Since $\mathcal{A}_s \subseteq \mathcal{V}_s$, the residual $\eta - P_{\mathcal{V}_s}\eta$ is orthogonal to $P_{\mathcal{V}_s}\eta - P_{\mathcal{A}_s}\eta$. Apply Pythagoras to the squared-loss projection decomposition. □

This gap describes target-relevant functions in the full prefix that are inaccessible to a linear readout of the fixed aggregate. The theorem does not assert that training makes the gap small, and PAC does not estimate or bound it. A nonlinear adapter need not satisfy the subspace inclusion used here.

Corollary 3 (Bounded gain for a fixed nested sequence)**.** *For a fixed infinite sequence of finite-dimensional source spaces on the same probability space, let* $\Delta_\ell = \mathcal{R}^\star_{\ell-1} - \mathcal{R}^\star_\ell$*. Then* $\Delta_\ell \geq 0$*,* $\sum_{\ell=0}^{\infty} \Delta_\ell \leq \|\eta\|^2_{L_2}$*, and* $\Delta_\ell \to 0$*.*

Proof. The innovations are pairwise orthogonal and $\sum_{\ell=0}^{L} \Delta_\ell = \|P_{\mathcal{V}_L}\eta\|^2_{L_2} \leq \|\eta\|^2_{L_2}$. The nonnegative increments of this bounded nondecreasing sequence converge to zero. □

This result neither orders finite-depth gains nor compares retrained models, whose source spaces can change. It does not predict IKR. For a local binary cross-entropy analogy, let $r \in (0,1)$ be the conditional label probability and define $\ell_{\mathrm{BCE}}(z;r) = \log(1+e^z) - rz$ and $z_* = \log(r/(1-r))$. If $z, z_* \in [-M, M]$ for $M > 0$, Taylor's theorem gives

$$\frac{\sigma(M)(1-\sigma(M))}{2}(z-z_*)^2 \leq \ell_{\mathrm{BCE}}(z;r) - \ell_{\mathrm{BCE}}(z_*;r) \leq \frac{1}{8}(z-z_*)^2, \tag{33}$$

where σ is the logistic sigmoid. This bounded-logit comparison does not transfer the prefix-readout identity to AUC, test error, or the trained nonlinear model.

D A Supplementary Covariance Coding-Rate Interpretation

Log-determinant coding rates provide a second interpretation of the unweighted covariance overlaps. MCR^2 uses whole-representation and class-conditioned coding rates (Yu et al., 2020; Wang et al., 2024); here we instead partition a covariance by source depth. The resulting quantity is an analytical functional, not an additional trained loss or reported empirical metric.

Stack the standardized sources at a fixed token and let $\boldsymbol{\Sigma}$ be their block covariance, with $\boldsymbol{\Sigma}_{km} = \mathbb{E}[\boldsymbol{Z}_k \boldsymbol{Z}_m^\top] = \boldsymbol{C}_{km}$. For $\tau > 0$, define

$$\mathrm{DCR}_\tau = \frac{1}{2}\left[\sum_{\ell=0}^{L} \log\det(\boldsymbol{I} + \tau^{-1}\boldsymbol{\Sigma}_{\ell\ell}) - \log\det(\boldsymbol{I} + \tau^{-1}\boldsymbol{\Sigma})\right]. \tag{34}$$

Theorem 5 (Nonnegativity and a local quadratic expansion)**.** *For* $\boldsymbol{\Sigma} \succeq 0$*,* $\mathrm{DCR}_\tau \geq 0$*, with equality exactly when every off-diagonal covariance block is zero. Write* $\boldsymbol{B}_\ell = \boldsymbol{I} + \tau^{-1}\boldsymbol{\Sigma}_{\ell\ell}$*,* $\boldsymbol{D} = \mathrm{blkdiag}(\boldsymbol{B}_0, \ldots, \boldsymbol{B}_L)$*, and*

$$\boldsymbol{M} = \boldsymbol{D}^{-1/2}\tau^{-1}[\boldsymbol{\Sigma} - \mathrm{blkdiag}(\boldsymbol{\Sigma}_{00}, \ldots, \boldsymbol{\Sigma}_{LL})]\boldsymbol{D}^{-1/2}. \tag{35}$$

If $\rho = \|\boldsymbol{M}\|_2 < 1$*, then*

$$\mathrm{DCR}_\tau = \frac{1}{2}\sum_{\ell<m} \|\boldsymbol{B}_\ell^{-1/2}(\tau^{-1}\boldsymbol{\Sigma}_{\ell m})\boldsymbol{B}_m^{-1/2}\|_F^2 + \mathcal{E}_3, \qquad |\mathcal{E}_3| \leq \frac{\rho}{6(1-\rho)}\|\boldsymbol{M}\|_F^2. \tag{36}$$

Proof. Let $\boldsymbol{A} = \boldsymbol{I} + \tau^{-1}\boldsymbol{\Sigma} \succ 0$. For two blocks, the Schur-complement determinant formula gives

$$\det \boldsymbol{A} = \det \boldsymbol{A}_{11} \det(\boldsymbol{A}_{22} - \boldsymbol{A}_{21}\boldsymbol{A}_{11}^{-1}\boldsymbol{A}_{12}) \leq \det \boldsymbol{A}_{11} \det \boldsymbol{A}_{22}.$$

The inequality follows because subtracting the positive semidefinite Schur term cannot increase the determinant of a positive definite matrix. Equality requires $\boldsymbol{A}_{21}\boldsymbol{A}_{11}^{-1}\boldsymbol{A}_{12} = 0$, hence $\boldsymbol{A}_{12} = 0$. Repeating the argument over the depth blocks proves nonnegativity and the equality condition.

Next, $\boldsymbol{A} = \boldsymbol{D}^{1/2}(\boldsymbol{I} + \boldsymbol{M})\boldsymbol{D}^{1/2}$, so $\mathrm{DCR}_\tau = -\frac{1}{2}\log\det(\boldsymbol{I} + \boldsymbol{M})$. The symmetric matrix $\boldsymbol{M}$ has zero diagonal blocks and hence zero trace. For $\rho < 1$, expand the matrix logarithm to obtain

$$\mathrm{DCR}_\tau = \frac{1}{4}\operatorname{tr}(\boldsymbol{M}^2) - \frac{1}{2}\sum_{k=3}^{\infty} \frac{(-1)^{k+1}}{k} \operatorname{tr}(\boldsymbol{M}^k).$$

For every $k \geq 3$, $|\operatorname{tr}(\boldsymbol{M}^k)| \leq \rho^{k-2}\|\boldsymbol{M}\|_F^2$. Bounding $1/k$ by $1/3$ and summing the geometric series gives the stated remainder. Finally, $\operatorname{tr}(\boldsymbol{M}^2) = 2\sum_{\ell<m} \|\boldsymbol{M}_{\ell m}\|_F^2$, which gives the blockwise quadratic term. □

If $\boldsymbol{\Sigma}_{\ell\ell} = \boldsymbol{I}$ for every source, the quadratic term is $[2(\tau+1)^2]^{-1}\sum_{\ell<m}\|\boldsymbol{\Sigma}_{\ell m}\|_F^2$. Under the same local spectral condition and equal source widths, this term is proportional to the population counterpart of $\mathcal{L}_{\mathrm{dec}}$ when the pair weights are uniform. Learned nonuniform weights change the relative contribution of each pair, so the expansion interprets the underlying covariance overlaps rather than identifying the trained objective with a coding rate. Coordinate standardization alone does not ensure the isotropic assumption.

For an affine depth-specific adapter, the transformed stack has covariance blocks $\boldsymbol{\Gamma}_{km}^{[s]} = \boldsymbol{W}_s\boldsymbol{D}_k\boldsymbol{\Sigma}_{km}\boldsymbol{D}_m\boldsymbol{W}_s^\top$. Applying the theorem to that covariance gives the same nonnegativity and local expansion. Zero source cross-covariance implies zero transformed coding redundancy. If $\boldsymbol{W}_s$ has full column rank and all $\boldsymbol{D}_k$ are invertible, a left inverse of $\boldsymbol{W}_s$ proves the converse. These covariance statements do not establish independence or transfer optimization guarantees for MCR^2 to the proposed wrapper.