On the Embedding Collapse When Scaling Up Recommendation Models

Xingzhuo Guo [1] * Junwei Pan [2] Ximei Wang [2] Baixu Chen [1] Jie Jiang [2] Mingsheng Long [1]

Abstract

Recent advances in foundation models have led to a promising trend of developing large recommendation models to leverage vast amounts of available data. Still, mainstream models remain embarrassingly small in size and naïve enlarging does not lead to sufficient performance gain, suggesting a deficiency in the *model scalability*. In this paper, we identify the *embedding collapse* phenomenon as the inhibition of scalability, wherein the embedding matrix tends to occupy a low-dimensional subspace. Through empirical and theoretical analysis, we demonstrate a *two-sided effect* of feature interaction specific to recommendation models. On the one hand, interacting with collapsed embeddings restricts embedding learning and exacerbates the collapse issue. On the other hand, interaction is crucial in mitigating the fitting of spurious features as a scalability guarantee. Based on our analysis, we propose a simple yet effective *multi-embedding* design incorporating embedding-set-specific interaction modules to learn embedding sets with large diversity and thus reduce collapse. Extensive experiments demonstrate that this proposed design provides consistent scalability and effective collapse mitigation for various recommendation models. Code is available at this repository: `https://github.com/thuml/Multi-Embedding`.

1. Introduction

Recommender systems are important machine learning scenarios that predict users' actions on items based on tremendous multi-field categorical data (Zhang et al., 2016), which play an indispensable role in our daily lives to help people discover information about their interests and have been adopted in a wide range of online applications, such as E-commerce, social media, news feeds, and music streaming. Researchers have developed deep-learning-based recommendation models to dig feature representations flexibly. These models have been successfully deployed across a multitude of application scenarios, thereby demonstrating their widespread adoption and effectiveness.

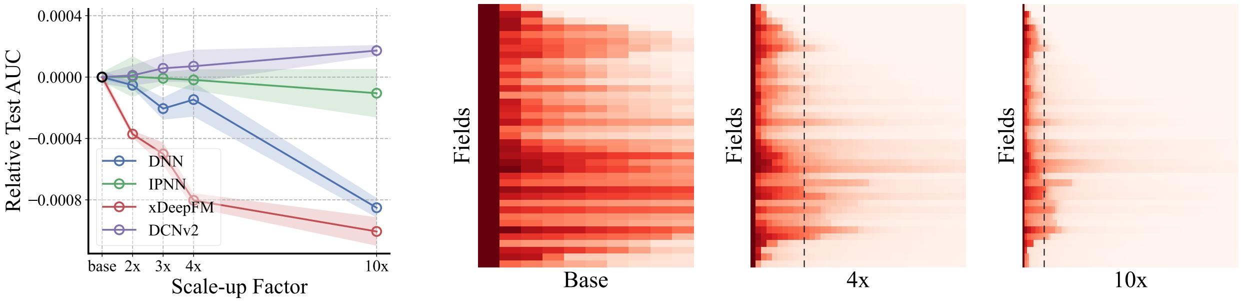
Motivated by the advancement of large foundation models (Kirillov et al., 2023; OpenAI, 2023; Radford et al., 2021; Rombach et al., 2022) that benefit from increasing parameters, it should have been a promising trend to scale up the recommendation model size to use the data amount fully. Yet counterintuitively, as the most-weighted component critical for performance (Qu et al., 2016; Lian et al., 2018; Wang et al., 2021), the embeddings of recommendation models are typically tuned too small such as a size of 10 (Zhu et al., 2022), and thus do not adequately capture the magnitude of data. Worsely, increasing the embedding size does not sufficiently improve the performance or even hurts the model, as shown in Figure 1a. This suggests a deficiency in the *model scalability* of existing architecture designs, constraining the potential upper bound for recommender systems.

To figure out the reason behind it, we take a spectral analysis on the learned embedding matrices based on singular value decomposition and exhibit the normalized singular values in Figure 1b. Surprisingly, most singular values are significantly small, *i.e.*, the learned embedding matrices are nearly low-rank, which we refer to as the *embedding collapse* phenomenon. With the enlarged model size, the model does not learn to capture a larger dimension of information, implying a learning process with ineffective parameter utilization, which restricts the scalability.

In this work, we study the mechanism behind the embedding collapse through empirical and theoretical analysis and shed light on the *two-sided effect* on model scalability of the feature interaction module, the cornerstone of recommendation models to model higher-order correlations. On the one hand, interaction with collapsed embeddings will constrain the embedding learning and, thus, in turn, aggravate the collapse issue. On the other hand, the feature interaction also plays a vital role in reducing overfitting when scaling up models, which cannot be restricted or removed.

[*] Work partially done while an intern at Tencent Inc. [1] School of Software, BNRist, Tsinghua University, China [2] Tencent Inc, China. Xingzhuo Guo <gxz23@mails.tsinghua.edu.cn>. Correspondence to: Mingsheng Long <mingsheng@tsinghua.edu.cn>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).



(a) Performance when scaling up recommendation models

(b) Singular values of DCNv2 under different model size, with the dashed lines corresponding to the base size.

Figure 1. Unsatisfactory scalability of existing recommendation models. **(a)**: Increasing the embedding size does not improve remarkably or even hurts the model performance. **(b)**: Most embedding matrices do not learn large singular values and tend to be low-rank.

Based on our analysis, we conclude the principle to mitigate collapse without suppressing feature interaction, so that scalable models can be approached. We propose *multi-embedding* as a simple yet efficient design for model scaling. Multi-embedding scales the number of independent embedding sets and incorporates embedding-set-specific interaction modules to jointly capture different patterns. Our experimental results demonstrate that multi-embedding provides scalability for extensive mainstream models and significantly mitigates embedding collapse, pointing to a methodology of breaking through the *size limit* of recommender systems.

Our contributions can be summarized as:

- To the best of our knowledge, we are the first to point out the *model scalability* issue in recommender systems and discover the *embedding collapse* phenomenon, an urgent problem to address to enhance scalability.
- Using empirical and theoretical analysis, we shed light on the *two-sided effect* of the feature interaction process on scalability based on the collapse phenomenon. We reveal that feature interaction leads to collapse while providing essential overfitting resistance.
- Following our concluded principle to mitigate collapse without suppressing feature interaction, we propose *multi-embedding* as a simple unified design, consistently improving scalability and effectively mitigating embedding collapse for extensive state-of-the-art recommendation models.

2. Preliminaries

Recommendation models aim to predict an action based on features from various fields. Throughout this paper, we consider the fundamental scenario of recommender systems, in which categorial features and binary outputs are involved. Formally, suppose there are N fields, with the i-th field denoted as $\mathcal{X}_i = \{1, 2, ..., D_i\}$, where D_i denotes the field cardinality. Let

$$\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times ... \times \mathcal{X}_N$$

and $\mathcal{Y} = \{0, 1\}$, then recommendation models aim to learn a mapping from $\mathcal{X}$ to $\mathcal{Y}$. In addition to considering individual features from diverse fields, there have been numerous studies (Koren et al., 2009; Rendle, 2010; Juan et al., 2016; Guo et al., 2017; Lian et al., 2018; Pan et al., 2018; Sun et al., 2021; Wang et al., 2021) within the area of recommender systems to model combined features using *feature interaction* modules. In this work, we investigate the following widely adopted architecture for mainstream models. A model comprises: (1) embedding layers $\boldsymbol{E}_i \in \mathbb{R}^{D_i \times K}$ for each field, with embedding size K; (2) an interaction module I responsible for integrating all embeddings into a combined feature scalar or vector; and (3) a subsequent postprocessing module F used for prediction purposes, such as MLP and MoE. The forward pass of such a model is formalized as

$$\begin{aligned} \boldsymbol{e}_i &= \boldsymbol{E}_i^\top \mathbf{1}_{x_i}, \ \forall i \in \{1, 2, ..., N\}, \\ h &= I(\boldsymbol{e}_1, \boldsymbol{e}_2, ..., \boldsymbol{e}_n), \\ \hat{y} &= F(h), \end{aligned}$$

where $\mathbf{1}_{x_i}$ indicates the one-hot encoding of $x_i \in \mathcal{X}_i$, in other words, $\boldsymbol{e}_i$ refers to (transposed) x_i-th row of the embedding table $\boldsymbol{E}_i$.

3. Embedding Collapse

Singular value decomposition has been widely used to measure the collapse phenomenon (Jing et al., 2021). In Figure 1b, we have shown that the learned embedding matrices of recommendation models are approximately low-rank with some extremely small singular values. To determine the degree of collapse for such matrices with low-rank tendencies, we propose *information abundance* as a generalized quantification.

Definition 3.1 (Information Abundance)**.** Consider a matrix $\boldsymbol{E} \in \mathbb{R}^{D\times K}$ and its singular value decomposition $\boldsymbol{E} = \boldsymbol{U\Sigma V} = \sum_{k=1}^{K} \sigma_k \boldsymbol{u}_k \boldsymbol{v}_k^\top$, then the *information abundance* of $\boldsymbol{E}$ is defined as

$$\mathrm{IA}(\boldsymbol{E}) = \frac{\|\boldsymbol{\sigma}\|_1}{\|\boldsymbol{\sigma}\|_\infty},$$

i.e., the sum of all singular values normalized by the maximum singular value.

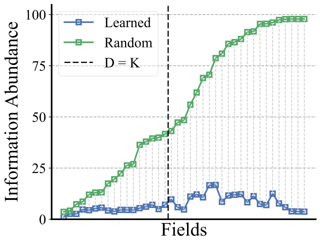


Figure 2. Visualization of information abundance on the Criteo dataset. Fields are sorted by their cardinalities.

Intuitively, a matrix with high information abundance demonstrates a balanced distribution in vector space since it has similar singular values. In contrast, a matrix with low information abundance suggests that the components corresponding to smaller singular values can be compressed without significantly impacting the result. Compared with matrix rank, information abundance can be regarded as a simple extension by noticing that $\mathrm{rank}(\boldsymbol{E}) = \|\boldsymbol{\sigma}\|_0$, yet it is applicable for non-strictly low-rank matrices, especially for fields with $D_i \gg K$ which is possibly of rank K. We calculate the information abundance of embedding matrices for the enlarged DCNv2 (Wang et al., 2021) and compare it with that of randomly initialized matrices, shown in Figure 2. It is observed that the information abundance of learned embedding matrices is extremely low, indicating the embedding collapse phenomenon.

4. Feature Interaction Revisited

In this section, we delve deeper into the embedding collapse phenomenon for recommendation models. Our investigation revisits feature interaction modules which are the key to recommendation models, and revolves around two questions: (1) How is embedding collapse caused? (2) How to properly mitigate embedding collapse and enhance scalability? Through empirical and theoretical studies, we shed light on the two-sided effect of feature interaction modules on model scalability.

4.1. Interaction-Collapse Theory

To determine how feature interaction leads to embedding collapse, it is inadequate to directly analyze the raw embedding matrices since the learned embedding matrix results from interactions with all other fields, making it difficult to isolate the impact of field-pair-level interaction on embedding learning. Under this obstacle, we propose empirical evidence on models with *sub-embeddings* and theoretical analysis on general models, and conclude that feature interaction causes embedding collapse, named the *interaction-collapse theory*.

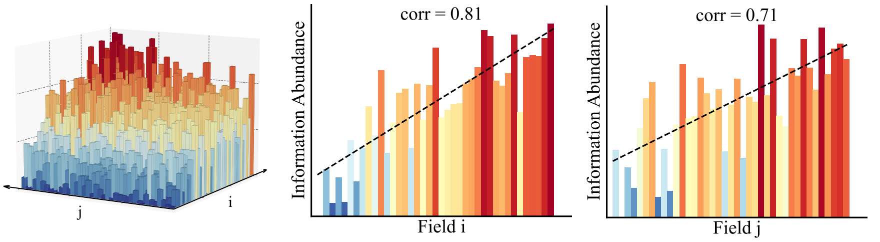
Evidence I: Empirical analysis on models with sub-embeddings. DCNv2 (Wang et al., 2021) incorporates a crossing network parameterized with transformation matrices $\boldsymbol{W}_{i\to j}$ (Sun et al., 2021) over each field pair to project an embedding vector from field i before interaction with field j. By collecting all projected embedding vectors, DCNv2 can be regarded to implicitly generate field-aware sub-embeddings $\boldsymbol{E}_i^{\to 1}, \boldsymbol{E}_i^{\to 2}, ..., \boldsymbol{E}_i^{\to N}$ to interact with all fields from embedding matrix $\boldsymbol{E}_i$, using

$$\boldsymbol{E}_i^{\to j} = \boldsymbol{E}_i \boldsymbol{W}_{i\to j}^\top.$$

DCNv2 consists of multiple stacked cross layers, and we only discuss the first layer as simplification. To determine the collapse of sub-embedding matrices, we calculate $\mathrm{IA}(\boldsymbol{E}_i^{\to j})$ for all i, j pair and show them in Figure 3a. For convenience, we pre-sort the field indices by the ascending order of information abundance, *i.e.*, i is ordered according to $\mathrm{IA}(\boldsymbol{E}_i)$, similar to j. We can observe that $\mathrm{IA}(\boldsymbol{E}_i^{\to j})$ is approximately increasing along i, which is trivial since $\boldsymbol{E}_i^{\to j}$ is simply a projection of $\boldsymbol{E}_i$. Interestingly, another correlation can be observed that the information abundance of sub-embeddings is co-influenced by the fields it interacts with, reflected by the increasing trend along j, especially for those with larger i. For instance, we further calculate the summation of $\mathrm{IA}(\boldsymbol{E}_i^{\to j})$ over j or i to study the effect of the other single variable, shown in Figure 3b and Figure 3c. The increasing trend and the correlation factor confirm the co-influence of i and j. We further analyze the IA of FFM (Juan et al., 2016) model which also consists of sub-embeddings as DCNv2, obtaining similar observations shown in Appendx H.

Evidence II: Theoretical analysis on general recommendation models. We now theoretically present how collapse is caused by feature interaction in general models, even without sub-embedding. For simplicity, we consider an FM-style (Rendle, 2010) feature interaction. Formally, the interaction process is defined by

$$h = \sum_{i=1}^{N}\sum_{j=1}^{i-1} \boldsymbol{e}_i^\top \boldsymbol{e}_j = \sum_{i=1}^{N}\sum_{j=1}^{i-1} \mathbf{1}_{x_i}^\top \boldsymbol{E}_i \boldsymbol{E}_j^\top \mathbf{1}_{x_j},$$



(a) $\mathrm{IA}(\boldsymbol{E}_i^{\to j})$. (b) $\sum_{j=1}^{N}\mathrm{IA}(\boldsymbol{E}_i^{\to j})$. (c) $\sum_{i=1}^{N}\mathrm{IA}(\boldsymbol{E}_i^{\to j})$.

Figure 3. Information abundance of sub-embedding matrices for DCNv2, with field indices sorted by information abundance of corresponding raw embedding matrices. Higher or warmer indicates larger. It is observed that $\mathrm{IA}(\boldsymbol{E}_i^{\to j})$ are co-influenced by both $\mathrm{IA}(\boldsymbol{E}_i)$ and $\mathrm{IA}(\boldsymbol{E}_j)$.

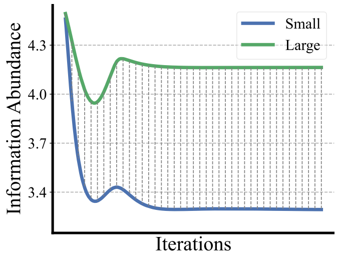


Figure 4. $\mathrm{IA}(\boldsymbol{E}_1)$ for toy experiments. "Small" and "Large" refers to the cardinality of $\mathcal{X}_3$.

where h is the combined feature as mentioned before. Without loss of generality, we discuss one specific row $\boldsymbol{e}_1$ of $\boldsymbol{E}_1$ and keep other embedding matrices fixed. Consider a mini-batch with batch size B. Denote $\sigma_{i,k}$ as the k-th singular value of $\boldsymbol{E}_i$, and denote $\boldsymbol{u}_{i,k}$, $\boldsymbol{v}_{i,k}$ as corresponding singular vectors. We have

$$
\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \boldsymbol{e}_1} &= \frac{1}{B}\sum_{b=1}^{B}\frac{\partial \ell^{(b)}}{\partial h^{(b)}}\cdot\frac{\partial h^{(b)}}{\partial \boldsymbol{e}_1} = \frac{1}{B}\sum_{b=1}^{B}\frac{\partial \ell^{(b)}}{\partial h^{(b)}}\cdot\sum_{i=2}^{N}\boldsymbol{E}_i^\top \mathbf{1}_{x_i^{(b)}}\\
&= \frac{1}{B}\sum_{b=1}^{B}\frac{\partial \ell^{(b)}}{\partial h^{(b)}}\cdot\sum_{i=2}^{N}\sum_{k=1}^{K}\sigma_{i,k}\boldsymbol{v}_{i,k}\boldsymbol{u}_{i,k}^\top \mathbf{1}_{x_i^{(b)}}\\
&= \sum_{i=2}^{N}\sum_{k=1}^{K}\left(\frac{1}{B}\sum_{b=1}^{B}\frac{\partial \ell^{(b)}}{\partial h^{(b)}}\boldsymbol{u}_{i,k}^\top \mathbf{1}_{x_i^{(b)}}\right)\sigma_{i,k}\boldsymbol{v}_{i,k}\\
&= \sum_{i=2}^{N}\sum_{k=1}^{K}\alpha_{i,k}\sigma_{i,k}\boldsymbol{v}_{i,k} = \sum_{i=2}^{N}\boldsymbol{\theta}_i,\\
\text{where}\quad \boldsymbol{\theta}_i &= \sum_{k=1}^{K}\alpha_{i,k}\sigma_{i,k}\boldsymbol{v}_{i,k}.
\end{aligned}
$$

The equation means that the gradient can be decomposed into field-specific terms. We consider the component $\boldsymbol{\theta}_i$ for a certain field i, which is further decomposed into spectra for the corresponding embedding matrix $\boldsymbol{E}_i$. From the form of $\boldsymbol{\theta}_i$, it is observed that $\{\alpha_{i,k}\}$ are $\boldsymbol{\sigma}_i$-agnostic scalars determined by the training data and objective function. Thus, the variety of $\boldsymbol{\sigma}_i$ significantly influences the composition of $\boldsymbol{\theta}_i$. For those larger $\sigma_{i,k}$, the gradient component $\boldsymbol{\theta}_i$ will be weighted more heavily along the corresponding spectra $\boldsymbol{v}_{i,k}$. When $\boldsymbol{E}_i$ is low-information-abundance, the components of $\boldsymbol{\theta}_i$ weigh imbalancely, resulting in the degeneration of $\boldsymbol{e}_1$. Since different $\boldsymbol{e}_1$ affects only $\alpha_{i,k}$ instead of $\sigma_{i,k}$ and $\boldsymbol{v}_{i,k}$, all rows of $\boldsymbol{E}_1$ degenerates in similar manners and finally form a collapsed matrix.

To further illustrate, we conduct a toy experiment over synthetic data. Suppose there are $N = 3$ fields, and we set D_3 to different values with $D_3 < K$ and $D_3 \gg K$ to simulate low-information-abundance and high-information-abundance cases, which matches the diverse range of the field cardinality in real-world scenarios. We train $\boldsymbol{E}_1$ while keeping $\boldsymbol{E}_2, \boldsymbol{E}_3$ fixed. Details of experiment setups are discussed in Appendix G. We show the information abundance of $\boldsymbol{E}_1$ along the training process for the two cases in Figure 4. It is observed that interacting with a low-information-abundance matrix will result in a collapsed embedding.

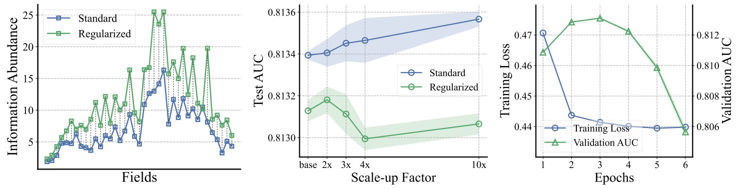
Summary: How is collapse caused in recommendation models? Evidence I highlights that interacting with a low-information-abundance field will result in a more collapsed sub-embedding. By considering the fact that sub-embeddings reflect the effect when fields interact since it originates from raw embeddings, we recognize the inherent mechanism of feature interaction to cause collapse, which is further confirmed by our theoretical analysis. We conclude the *interaction-collapse theory*:

> *Finding 1 (Interaction-Collapse Theory). In feature interaction of recommendation models, fields with low-information-abundance embeddings constrain the information abundance of other fields, resulting in collapsed embedding matrices.*

The interaction-collapse theory generally suggests that feature interaction is the primary catalyst for collapse, thereby imposing constraints on the ideal scalability.

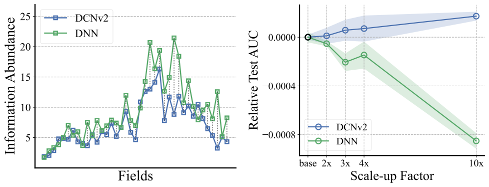
4.2. Is It Sufficient to Avoid Collapse for Scalability?

Following our discussion above, we have shown that the feature interaction process of recommendation models leads to collapse and thus limits the model scalability. We now discuss its *negative proposition*, *i.e.*, whether suppressing the feature interaction to mitigate collapse leads to model scalability. To answer this question, we design the following two experiments to compare standard models and models with feature interaction suppressed.



(a) IA w/ 10x size. (b) Test AUC w.r.t. size. (c) Training curve.

Figure 5. Experimental results of Evidence III. Restricting DCNv2 leads to higher information abundance, yet the model suffers from over-fitting, thus resulting in non-scalability.



(a) IA w/ 10x size. (b) Test AUC w.r.t. size.

Figure 6. Experimental results of Evidence IV. Despite higher information abundance, the performance of DNN drops w.r.t. model size.

Evidence III: Limiting the modules in interaction that leads to collapse. Evidence I shows that a projection $\boldsymbol{W}_{i\to j}$ is learned to adjust information abundance for sub-embeddings and lead to collapse. We now inverstigate how surpressing such effect would result in model scalability by introducing the following regularization with learnable parameter λ_{ij}:

$$\ell_{reg} = \sum_{i=1}^{N}\sum_{j=1}^{N} \left\| \boldsymbol{W}_{i\to j}^{\top}\boldsymbol{W}_{i\to j} - \lambda_{ij}\boldsymbol{I} \right\|_{\mathrm{F}}^{2},$$

which regularizes the projection matrix to be a multiplication of an unitary matrix. This way, $\boldsymbol{W}_{i\to j}$ will preserve all normalized singular values and maintain the information abundance after projection. We experiment with various embedding sizes and compare the changes in performance, the information abundances, and the optimization dynamics for standard and regularized models. Results are shown in Figure 5. As anticipated, regularization in DCNv2 helps learn embeddings with higher information abundance. Nevertheless, the model presents unexpected results whereby the scalability does not improve or worsen even if the collapse is alleviated, and it is found that such a model overfits during the learning process with the training loss consistently decreasing and the validation AUC dropping.

Evidence IV: Directly avoiding explicit interaction. We now investigate how directly suppressing the feature interaction would affect the scalability. We discuss DNN, which consists of a plain interaction module that concatenates all feature vectors from different fields and processes them with an MLP. Since DNN does not conduct explicit 2-order feature interaction (Rendle et al., 2020), it would suffer less from collapse following our previous interaction-collapse theory. We compare the learned embeddings of DCNv2 and DNN and their performance with the growth of embedding size. Considering that different architectures or objectives may differ in modeling, we mainly discuss the performance trend as a fair comparison. Results are shown in Figure 6. DNN learns less-collapsed embedding matrices, reflected by higher information abundance than DCNv2. Yet perversely, the AUC of DNN drops when increasing the embedding size. Such observations show that DNN falls into the issue of overfitting and lacks scalability, even though it suffers less from collapse.

Summary: Does suppressing collapse definitely improve scalability? Regularized DCNv2 and DNN are both models with feature interaction suppressed, and they learn less-collapsed embedding matrices than DCNv2, as expected. Yet observations in evidence III&IV demonstrate that regularized DCNv2 and DNN both do not scale in AUC with the growth of model size and suffer from serious overfitting. We conclude the following finding:

> *Finding 2. A less-collapsed model with feature interaction suppressed improperly is insufficient for scalability due to overfitting concern.*

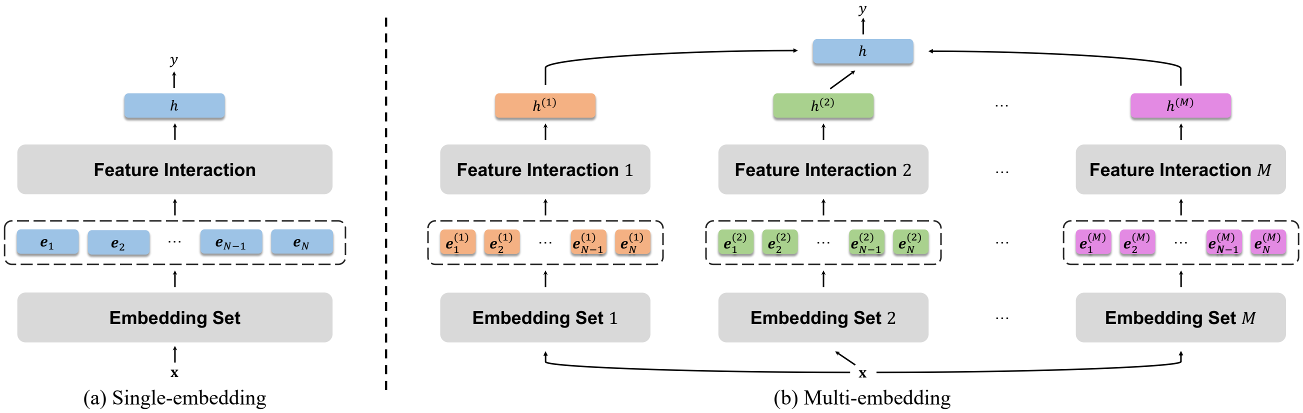
Such a finding is plausible, considering that feature interaction brings domain knowledge of higher-order correlations in recommender systems and helps form generalizable representations. When feature interaction is suppressed, models tend to fit noise as the embedding size increases, resulting in reduced generalization.

5. Multi-Embedding Design

In this section, we present a simple *multi-embedding* design, which serves as an effective scaling mechanism applicable to a wide range of recommendation model architectures. We introduce the overall architecture, present experimental results, and analyze how multi-embedding works. We also discuss the role of data to give a comprehensive analysys for multi-embedding.

5.1. Multi-Embedding

The two-sided effect of feature interaction for scalability implies a *principle* for model design. That is, a scalable model should be capable of less-collapsed embeddings within the existing feature interaction framework instead of removing interaction. Based on this principle, we propose *multi-embedding* or *ME* as a simple yet efficient design to improve scalability. Specifically, we scale up the number of


y
h
Feature Interaction
e_1 e_2 ⋯ e_{N-1} e_N
Embedding Set
x
y
h
$h^{(1)}$
$h^{(2)}$
⋯
$h^{(M)}$
Feature Interaction 1
Feature Interaction 2
Feature Interaction M
$e_1^{(1)}$ $e_2^{(1)}$ ⋯ $e_{N-1}^{(1)}$ $e_N^{(1)}$
$e_1^{(2)}$ $e_2^{(2)}$ ⋯ $e_{N-1}^{(2)}$ $e_N^{(2)}$
$e_1^{(M)}$ $e_2^{(M)}$ ⋯ $e_{N-1}^{(M)}$ $e_N^{(M)}$
Embedding Set 1
Embedding Set 2
Embedding Set M
x

(a) Single-embedding

(b) Multi-embedding

Figure 7. Architectures of single-embedding (left) and multi-embedding (right) models.

independent and complete embedding sets instead of the embedding size, and incorporate embedding-set-specific feature interaction modules. In line with previous works such as group convolution (Krizhevsky et al., 2012), multi-head attention (Vaswani et al., 2017), and other decoupling-based works in recommender systems (Liu et al., 2019; 2022; Weston et al., 2013), such design allows the model to learn different interaction patterns jointly and result in embedding sets with large diversity, while a single-embedding model would be limited in pattern extraction and suffer from severe collapse. With multi-embedding, the model is less influenced by the interaction-collapse theory and mitigates embedding collapse while keeping the original interaction modules. Formally, a recommendation model with M sets of embeddings is defined as

$$\begin{aligned}
\boldsymbol{e}_i^{(m)} &= \left(\boldsymbol{E}_i^{(m)}\right)^\top \mathbf{1}_{x_i},\ \forall i \in \{1, 2, ..., N\}, \\
h^{(m)} &= I^{(m)}\left(\boldsymbol{e}_1^{(m)}, \boldsymbol{e}_2^{(m)}, ..., \boldsymbol{e}_N^{(m)}\right), \\
h &= \frac{1}{M}\sum_{m=1}^{M} h^{(m)}, \quad \hat{y} = F(h),
\end{aligned}$$

where m stands for the index of embedding set. One requirement of multi-embedding is that there should be non-linearities such as ReLU within the interaction module I; otherwise, the model is equivalent to single-embedding and hence does not capture different patterns.[1] As a solution, we add a non-linear projection after interaction for the models with linear interaction modules and reduce one MLP layer for postprocessing module F to achieve a fair comparison. An overall architecture comparison of single-embedding and mult-embedding models is shown in Figure 7.

5.2. Experiments

Setup. We conduct our experiments on two datasets for recommender systems: Criteo (Jean-Baptiste Tien, 2014) and Avazu (Steve Wang, 2014), which are large and challenging benchmark datasets widely used in recommender systems. We experiment on baseline models including DNN, IPNN (Qu et al., 2016), NFwFM (Pan et al., 2018), xDeepFM (Lian et al., 2018), DCNv2 (Wang et al., 2021), FinalMLP (Mao et al., 2023) and their corresponding multi-embedding variants with 2x, 3x, 4x and 10x model size. Here NFwFM is a variant of NFM (He & Chua, 2017) by replacing FM with FwFM. All experiments are performed with 8/1/1 training/validation/test splits, and we apply early stopping based on validation AUC. More details are shown in Appendix C.2.

Results. We repeat each experiment 3 times and report the average test AUC with different scaling factors of the model size. Results are shown in Table 1. For the experiments with single-embedding, we observe that all the models demonstrate poor scalability. Only DCNv2 and NFwFM show slight improvements with increasing embedding sizes, with gains of 0.00036 on Criteo and 0.00093 on Avazu, respectively. For DNN, xDeepFM, and FinalMLP, which rely highly on non-explicit interaction, the performance even drops (0.00136 on Criteo and 0.00118 on Avazu) when scaled up to 10x, as discussed in Section 4.2. In contrast to single-embedding, our multi-embedding shows consistent and remarkable improvement with the growth of the embedding size, and the highest performance is always achieved with the largest 10x size. For DCNv2 and NFwFM, multi-embedding gains 0.00099 on Critio and 0.00223 on Avazu by scaling up to 10x, which is never obtained by single-embedding. Over all models and datasets, compared with baselines, the largest models averagely achieve 0.00110 im-

[1] See Appendix E.

Table 1. Test AUC for different models. Higher indicates better. Underlined and bolded values refer to the best performance with single-embedding (SE) and multi-embedding (ME), respectively.

Model		Criteo					Avazu				
		base	2x	3x	4x	10x	base	2x	3x	4x	10x
DNN	SE	0.81228	0.81222	0.81207	0.81213	0.81142	0.78744	0.78759	0.78752	0.78728	0.78648
	ME		0.81261	**0.81288**	**0.81289**	**0.81287**		0.78805	0.78826	0.78862	**0.78884**
IPNN	SE	0.81272	0.81273	0.81272	0.81271	0.81262	0.78732	0.78741	0.78738	0.78750	0.78745
	ME		0.81268	0.81270	0.81273	**0.81311**		0.78806	0.78868	0.78902	**0.78949**
NFwFM	SE	0.81059	0.81087	0.81090	0.81112	0.81113	0.78684	0.78757	0.78783	0.78794	0.78799
	ME		0.81128	0.81153	0.81171	**0.81210**		0.78868	0.78901	0.78932	**0.78974**
xDeepFM	SE	0.81217	0.81180	0.81167	0.81137	0.81116	0.78743	0.78750	0.78714	0.78735	0.78693
	ME		0.81236	0.81239	0.81255	**0.81299**		0.78848	0.78886	0.78894	**0.78927**
DCNv2	SE	0.81339	0.81341	0.81345	0.81346	0.81357	0.78786	0.78835	0.78854	0.78852	0.78856
	ME		0.81348	0.81361	**0.81382**	**0.81385**		0.78862	0.78882	0.78907	**0.78942**
FinalMLP	SE	0.81259	0.81262	0.81248	0.81240	0.81175	0.78751	0.78797	0.78795	0.78742	0.78662
	ME		0.81290	**0.81302**	**0.81303**	**0.81303**		0.78821	**0.78831**	**0.78836**	**0.78830**

provement on the test AUC[2]. Multi-embedding provides a methodology to break through the non-scalability limit of existing models. We visualize the scalability of multi-embedding on Criteo dataset in Figure 8a. The standard deviation and detailed scalability comparison are shown in Appendix C.3.

Mitigation of embedding collapse. We compare the information abundance of single-embedding and multi-embedding DCNv2 with the largest 10x embedding size to measure the mitigation of collapse, with all embedding sets for a single field concatenated together as the overall embedding in multi-embedding DCNv2. Results are shown in Figure 8b. It is observed that multi-embedding DCNv2 consistently increases the information abundance for all fields compared with single-embedding DCNv2, especially for fields with larger cardinality. These results indicate that multi-embedding is a simple yet effective method to mitigate embedding collapse for scalability gain without introducing significant computational resources or hyper-parameters.

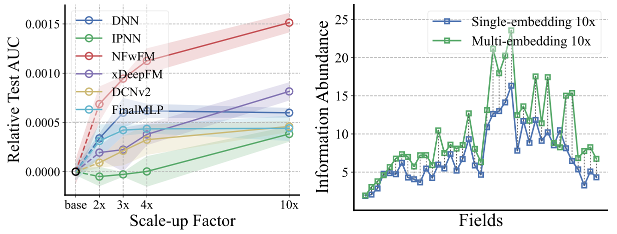
Deployment in the online system. After the online A/B testing in January 2023, the multi-embedding paradigm has been successfully deployed in Tencent's Online Advertising Platform, one of the largest advertisement recommendation systems. Upgrading the click prediction model from the single-embedding paradigm to our proposed multi-embedding paradigm in WeChat Moments leads to a 3.9% GMV (Gross Merchandise Value) lift, which brings hundreds of millions of dollars in revenue lift per year. Details are introduced in Pan et al. (2024).

5.3. How Multi-Embedding works?

Less influenced by the interaction-collapse theory. Following our previous interaction-collapse theory and corresponding analysis, embedding collapse is caused by feature interaction between different fields, reflected by the co-influence on the information abundance of sub-embeddings. We show that multi-embedding suffers less from such an effect. Recall in Section 4 that we calculate $\sum_{i=1}^{N} \mathrm{IA}(\boldsymbol{E}_i^{\to j})$ to compare how $\mathrm{IA}(\boldsymbol{E}_i^{\to j})$ is influenced by the field to interact with. We here correspondingly visualize the result for multi-embedding and single-embedding DCNv2, shown in Figure 9. It can be observed that the correlation factor in multi-embedding is significantly less than that in single-embedding (0.52 vs. 0.68). Thus, the information abundance is less influenced by the field it interacts with, suffering less from the impact of the interaction-collapse theory.

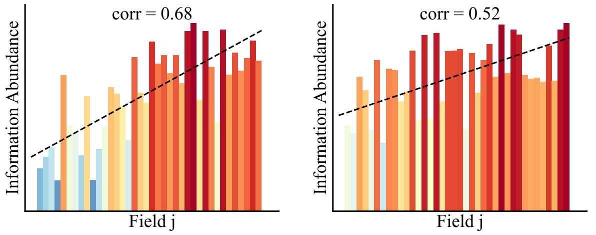
Mitigating collapse from embedding diversity. We further demonstrate that multi-embedding mitigates collapse by allowing the diversity of embedding sets. To illustrate, we introduce the cosine of *principal angle* (Miao & Ben-Israel, 1992), $\cos\left(\phi_i^{m \leftrightarrow m'}\right)$, to measure space similarity between a pair of embedding sets m, m' for any certain field i, calculated by the following further singular value decomposition:

[2] A slightly higher AUC at 0.001-level is regarded significant (Cheng et al., 2016; Guo et al., 2017; Song et al., 2019; Tian et al., 2023)



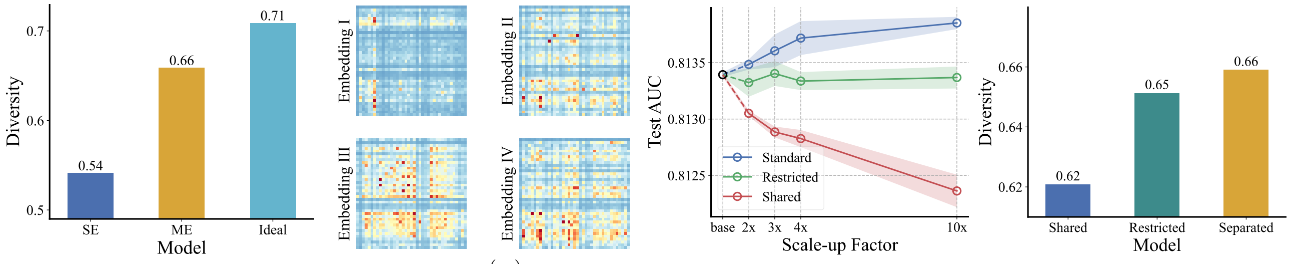
(a) Multi-embedding on Criteo. (b) $\mathrm{IA}(\boldsymbol{E}_i)$ on DCNv2.

Figure 8. Effectiveness of multi-embedding. **(a)** Model performance consistently improves with increasing size when using multi-embedding on 6 mainstream models. **(b)** Multi-embedding consistently enhances information abundance across all fields compared with single-embedding.



(a) $\sum_{i=1}^N \mathrm{IA}(\boldsymbol{E}_i^{\to j})$, SE. (b) $\sum_{i=1}^N \mathrm{IA}(\boldsymbol{E}_i^{\to j})$, ME.

Figure 9. Information abundance of sub-embedding matrices for single-embedding and multi-embedding on DCNv2, with field indices sorted by information abundance of corresponding raw embedding matrices. Higher or warmer indicates larger. $\mathrm{IA}(\boldsymbol{E}_i^{\to j})$ are less influenced by j or $\mathrm{IA}(\boldsymbol{E}_j)$ in multi-embedding.



(a) Diversity of ME & SE. (b) $\|\boldsymbol{W}_{i\to j}^{(m)}\|_{\mathrm{F}}$. (c) Scaling up ME variants. (d) Diversity of ME variants.

Figure 10. Analysis of multi-embedding (ME). **(a)**: Multi-embedding learns embedding sets with significantly larger diversity than single-embedding. **(b)**: Embedding-set-specific feature interaction modules capture different interaction patterns. **(c)**–**(d)**: Using variants of ME with non-separated interaction lead to worse scalability and lower embedding set diversity.

$$\left(\boldsymbol{U}_i^{(m)}\right)^{\top}\boldsymbol{U}_i^{(m')} = \boldsymbol{P}_i^{(m)}\mathrm{diag}\left(\cos\left(\boldsymbol{\phi}_i^{m\leftrightarrow m'}\right)\right)\left(\boldsymbol{P}_i^{(m')}\right)^{\top}.$$

A low-rank $\left(\boldsymbol{U}_i^{(m)}\right)^{\top}\boldsymbol{U}_i^{(m')}$ implies a high-rank overall embedding $[\boldsymbol{E}_i^{(m)}, \boldsymbol{E}_i^{(m')}]$.[3] We therefore discuss a gereralized measurement

$$\mathrm{div}(\boldsymbol{E}_i^{(m)}, \boldsymbol{E}_i^{(m')}) = 1 - \frac{1}{K}\left\|\cos\left(\boldsymbol{\phi}_i^{m\leftrightarrow m'}\right)\right\|_1$$

to describe the diversity of embedding sets. A larger diversity implies larger information abundance for the overall embedding or better mitigation of collapse. For comparision, we split the embedding of a single-embedding DCNv2 and an ideal random-initialized matrix into embedding sets, and compare with multi-embedding DCNv2. We show the average diversity across all embedding set pairs and all fields in Figure 10a. It is shown that multi-embedding can significantly lower the embedding set similarity compared with single-embedding, mitigating embedding collapse.

Yielding diversity from separated interaction. We further demonstrate the embedding diversity of multi-embedding models originates from embedding-set-specific feature interaction modules, which allows the embedding sets to capture diverse interaction patterns. On the one hand, we visualize $\|\boldsymbol{W}_{i\to j}^{(m)}\|_{\mathrm{F}}$ as the interaction pattern (Wang et al., 2021) for a multi-embedding DCNv2 model in Figure 10b. It is shown that the interaction modules learn various patterns. On the other hand, we compare multi-embedding with two variants with non-separated interaction: (a) All feature interaction modules are shared across all embedding sets, and (b) The divergence of $\|\boldsymbol{W}_{i\to j}^{(m)}\|_{\mathrm{F}}$ across all embedding sets are restricted by regularizations. Results are shown in Figure 10c and Figure 10d. Compared with the separated design in multi-embedding, the two variants of feature interaction design show worse scalability and embedding diversity, indicating that multi-embedding works from the separation of interaction modules.

5.4. Role of Data in Embedding Collapse

Throughout our work, we mainly focus on the model scalability and conclude the intrinsic issue, embedding collapse, of recommendation models. This considerable data amount of our benchmark datasets of experiments provides credibility to the embedding collapse phenomenon within a data-amount-agnostic manner. In this section we further discuss

[3] See Appendix F.

Table 2. Averaged information abundance under various embedding sizes and data amount.

Embedding Size	5	10	15	20	25	30	35	40	40, ME
1% data	3.74	5.96	7.97	9.23	10.44	11.51	12.39	13.30	14.99
3% data	3.16	4.80	5.96	6.67	7.30	8.10	8.46	8.84	10.59
10% data	**2.96**	4.28	5.08	5.73	6.43	6.79	7.11	7.51	8.63
30% data	2.97	**4.09**	**4.89**	**5.41**	**5.87**	6.19	6.53	6.78	7.70
100% data	3.29	4.73	5.41	5.70	5.95	**6.19**	**6.48**	**6.64**	7.64

how the embedding collapse phenomenon acts under various data amount. To illustrate, we conduct additional experiments using variously sized subsets of the Criteo dataset. We measured the average information abundance in embedding matrices across different model scales, summarized in Table 2. From the results, it is observed that the data size can indeed affect the information abundance of embedding matrices, but the information abundance does not strictly increase or even decreases along with the data size, especially for larger models. Behind this finding is that, embedding collapse is determined by two aspects: (1) the data size, which increases the information abundance, and (2) the interaction-collapse law, which decreases the information abundance. Among all the results, only for experiments with $10\% \sim 100\%$ data size and embedding size 5, and $30\% \sim 100\%$ data size and embedding size 10, 15, 20, 25, can we observe the collapse is caused by the limited data. And for most of the other cases, the abnormal decreasing trend shows that the collapse is caused by the interaction-collapse law rather than the limited data. Also throughout the results, multi-embedding consistently outperforms single-embedding under various data amount, indicating the universality of our proposed multi-embedding design.

6. Related Works

Modules in recommender systems. Plenty of existing works investigate the module design for recommender systems. A line of studies focuses on feature interaction process (Koren et al., 2009; Rendle, 2010; Juan et al., 2016; Qu et al., 2016; He & Chua, 2017; Guo et al., 2017; Pan et al., 2018; Lian et al., 2018; Song et al., 2019; Cheng et al., 2020; Sun et al., 2021; Wang et al., 2021; Mao et al., 2023; Tian et al., 2023), which is specific for recommender systems. These works are built up to fuse domain-specific knowledge of recommender systems. In contrast to proposing new modules, our work starts from a view of machine learning and analyzes the existing models for scalability.

Collapse phenomenon. Neural collapse or representation collapse describes the degeneration of representation vectors with restricted variation. This phenomenon is widely studied in supervised learning (Papyan et al., 2020; Zhu et al., 2021; Tirer & Bruna, 2022), unsupervised contrastive learning (Hua et al., 2021; Jing et al., 2021; Gupta et al., 2022), transfer learning (Aghajanyan et al., 2020; Kumar et al., 2022) and generative models (Mao et al., 2017; Miyato et al., 2018). Chi et al. (2022) discuss the representation collapse in sparse MoEs. Inspired by these works, we realize the embedding collapse of recommendation models when regarding embedding vectors as representations by their definition, yet we are facing the setting of field-level interaction, which has not previously been well studied.

Intrinsic dimensions and compression theories. To describe the complexity of data, existing works include intrinsic-dimension-based quantification (Levina & Bickel, 2004; Ansuini et al., 2019; Pope et al., 2020) and pruning-based analysis (Wen et al., 2017; Alvarez & Salzmann, 2017; Sun et al., 2021). Our SVD-based concept of information abundance is related to these works.

7. Conclusion

In this paper, we highlight the non-scalability issue of existing recommendation models and identify the embedding collapse phenomenon that hinders scalability. From empirical and theoretical analysis around embedding collapse, we conclude the two-sided effect of feature interaction on scalability, *i.e.*, feature interaction causes collapse while reducing overfitting. We propose a unified design of multi-embedding to mitigate collapse without suppressing feature interaction. Experiments on benchmark datasets demonstrate that multi-embedding consistently improves model scalability and effectively mitigates embedding collapse.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (62022050 and U2342217), the BNRist Innovation Fund (BNR2024RC01010), the Tencent Innovation Fund, and the National Engineering Research Center for Big Data Software. We also gratefully acknowledge the contributions of the following: Erheng Zhong, Xueming Qiu, Cong Li, Hui Tang, Xueyu Shi and Renjie Jiang for assistance with online deployment and experiments.

Impact Statement

This paper presents work whose goal is to advance the deep learning research for scaling up recommendation models. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Aghajanyan, A., Shrivastava, A., Gupta, A., Goyal, N., Zettlemoyer, L., and Gupta, S. Better fine-tuning by reducing representational collapse. In *ICLR*, 2020.

Alvarez, J. M. and Salzmann, M. Compression-aware training of deep networks. In *NeurIPS*, 2017.

Ansuini, A., Laio, A., Macke, J. H., and Zoccolan, D. Intrinsic dimension of data representations in deep neural networks. In *NeurIPS*, 2019.

Cheng, H.-T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., Anderson, G., Corrado, G., Chai, W., Ispir, M., et al. Wide & deep learning for recommender systems. In *DLRS*, 2016.

Cheng, W., Shen, Y., and Huang, L. Adaptive factorization network: Learning adaptive-order feature interactions. In *AAAI*, 2020.

Chi, Z., Dong, L., Huang, S., Dai, D., Ma, S., Patra, B., Singhal, S., Bajaj, P., Song, X., Mao, X.-L., et al. On the representation collapse of sparse mixture of experts. In *NeurIPS*, 2022.

Guo, H., Tang, R., Ye, Y., Li, Z., and He, X. Deepfm: a factorization-machine based neural network for ctr prediction. In *IJCAI*, 2017.

Gupta, K., Ajanthan, T., Hengel, A. v. d., and Gould, S. Understanding and improving the role of projection head in self-supervised learning. In *NeurIPS*, 2022.

He, X. and Chua, T.-S. Neural factorization machines for sparse predictive analytics. In *SIGIR*, 2017.

Hua, T., Wang, W., Xue, Z., Ren, S., Wang, Y., and Zhao, H. On feature decorrelation in self-supervised learning. In *ICCV*, 2021.

Jean-Baptiste Tien, joycenv, O. C. Display advertising challenge, 2014. URL https://kaggle.com/competitions/criteo-display-ad-challenge.

Jing, L., Vincent, P., LeCun, Y., and Tian, Y. Understanding dimensional collapse in contrastive self-supervised learning. In *ICLR*, 2021.

Juan, Y., Zhuang, Y., Chin, W.-S., and Lin, C.-J. Field-aware Factorization Machines for CTR Prediction. In *RecSys*, 2016.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., et al. Segment anything. *arXiv*, 2023.

Koren, Y., Bell, R., and Volinsky, C. Matrix factorization techniques for recommender systems. In *Computer*, 2009.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012.

Kumar, A., Raghunathan, A., Jones, R., Ma, T., and Liang, P. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *ICLR*, 2022.

Levina, E. and Bickel, P. Maximum likelihood estimation of intrinsic dimension. In *NeurIPS*, 2004.

Lian, J., Zhou, X., Zhang, F., Chen, Z., Xie, X., and Sun, G. xdeepfm: Combining explicit and implicit feature interactions for recommender systems. In *SIGKDD*, 2018.

Liu, F., Chen, H., Cheng, Z., Liu, A., Nie, L., and Kankanhalli, M. Disentangled multimodal representation learning for recommendation. In *TMM*, 2022.

Liu, N., Tan, Q., Li, Y., Yang, H., Zhou, J., and Hu, X. Is a single vector enough? exploring node polysemy for network embedding. In *SIGKDD*, 2019.

Mao, K., Zhu, J., Su, L., Cai, G., Li, Y., and Dong, Z. Finalmlp: An enhanced two-stream mlp model for ctr prediction. In *AAAI*, 2023.

Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z., and Paul Smolley, S. Least squares generative adversarial networks. In *ICCV*, 2017.

Miao, J. and Ben-Israel, A. On principal angles between subspaces in rn. In *Linear algebra and its applications*, 1992.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. Spectral normalization for generative adversarial networks. In *ICLR*, 2018.

OpenAI. Gpt-4 technical report. *arXiv*, 2023.

Pan, J., Xu, J., Ruiz, A. L., Zhao, W., Pan, S., Sun, Y., and Lu, Q. Field-weighted factorization machines for click-through rate prediction in display advertising. In *WWW*, 2018.

Pan, J., Xue, W., Wang, X., Yu, H., Liu, X., Quan, S., Qiu, X., Liu, D., Xiao, L., and Jiang, J. Ad recommendation in a collapsed and entangled world. *arXiv*, 2024.

Papyan, V., Han, X., and Donoho, D. L. Prevalence of neural collapse during the terminal phase of deep learning training. In *PNAS*, 2020.

Pope, P., Zhu, C., Abdelkader, A., Goldblum, M., and Goldstein, T. The intrinsic dimension of images and its impact on learning. In *ICLR*, 2020.

Qu, Y., Cai, H., Ren, K., Zhang, W., Yu, Y., Wen, Y., and Wang, J. Product-based neural networks for user response prediction. In *ICDM*, 2016.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. In *JMLR*, 2020.

Rendle, S. Factorization machines. In *ICDM*, 2010.

Rendle, S., Krichene, W., Zhang, L., and Anderson, J. Neural collaborative filtering vs. matrix factorization revisited. In *RecSys*, 2020.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

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 *CIKM*, 2019.

Steve Wang, W. C. Click-through rate prediction, 2014. URL https://kaggle.com/competitions/avazu-ctr-prediction.

Sun, Y., Pan, J., Zhang, A., and Flores, A. Fm2: Field-matrixed factorization machines for recommender systems. In *WWW*, 2021.

Tian, Z., Bai, T., Zhao, W. X., Wen, J.-R., and Cao, Z. Eulernet: Adaptive feature interaction learning via euler's formula for ctr prediction. *arXiv*, 2023.

Tirer, T. and Bruna, J. Extended unconstrained features model for exploring deep neural collapse. In *ICML*, 2022.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *NeurIPS*, 2017.

Wang, R., Shivanna, R., Cheng, D., Jain, S., Lin, D., Hong, L., and Chi, E. DCN V2: Improved Deep & Cross Network and Practical Lessons for Web-scale Learning to Rank Systems. In *WWW*, 2021.

Wen, W., Xu, C., Wu, C., Wang, Y., Chen, Y., and Li, H. Coordinating filters for faster deep neural networks. In *ICCV*, 2017.

Weston, J., Weiss, R. J., and Yee, H. Nonlinear latent factorization by embedding multiple user interests. In *RecSys*, 2013.

Zhang, W., Du, T., and Wang, J. Deep learning over multi-field categorical data: A case study on user response prediction. In *ECIR*, 2016.

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 *SIGIR*, 2022.

Zhu, Z., Ding, T., Zhou, J., Li, X., You, C., Sulam, J., and Qu, Q. A geometric analysis of neural collapse with unconstrained features. In *NeurIPS*, 2021.

A. Criticality of Embeddings

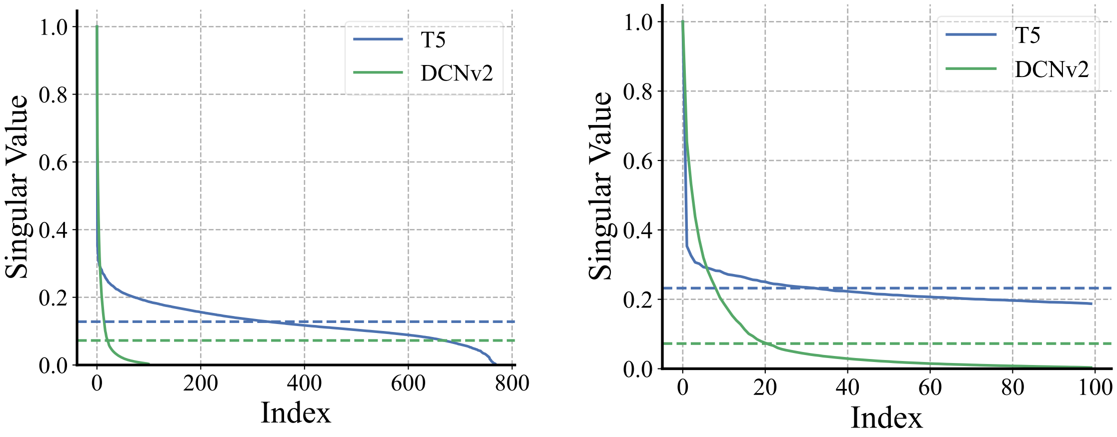
For recommendation models, the embedding module occupies the largest number of parameters ($> 92\%$ in our DCNv2 baseline for Criteo, and even larger for industrial models), and thus serves as the important and informative bottleneck part of the model. To further illustrate, we discuss the scaling up of the other modules of recommendation models, *i.e.*, the feature interaction module I and the postprocessing prediction module F. We experiment to increase #cross layers and #MLP layers in the DCNv2 baseline and show the results in Table 3. It is observed that increasing #cross layers or #MLP layers does not lead to performance improvement, hence it is reasonable and necessary to scale up the embedding size.

Table 3. Test AUC with enlarged feature interaction modules and portprocessing prediction modules. Higher indicates better.

DCNv2	1x	2x	4x
standard	0.81339	0.81341	0.81346
+ #cross layer	0.81325	0.81338	0.81344
+ #MLP depth	0.81337	0.81345	0.81342

B. Discussion on Embeddings in Language Models

To extend our analysis to other models, we examined the pretrained T5 (Raffel et al., 2020) model, evaluating its (normalized) singular values for comparison. Results are shown in Figure 11. It is observed that T5, in contrast to DCNv2, *(1)* maintains higher normalized singular values and *(2)* exhibits a lower proportion of insignificantly small singular values, despite its larger embedding dimensions. These observations suggest that T5 is less susceptible to the embedding collapse phenomenon, possibly because text-based models are not as affected by the interaction-collapse law in field interactions that causes embedding collapse.



(a) DCNv2 vs. T5

(b) DCNv2 vs. T5 (truncated)

Figure 11. Comparison between T5 and DCNv2. Dash lines indicate the average singular values.

C. Details of Experiment

C.1. Dataset Description

The statistics of Criteo and Avazu are shown in Table 4. It is shown that the data amount is ample and D_i can vary in a large range.

C.2. Experiment Settings

Specific multi-embedding design. For DCNv2, DNN, IPNN and NFwFM, we add one non-linear projection after the stacked cross layers, the concatenation layer, the inner product layer and the field-weighted dot product layer, respectively.

Table 4. Statistics of benchmark datasets for experiments.

Dataset	#Instances	#Fields	$\sum_i D_i$	$\max\{D_i\}$	$\min\{D_i\}$
Criteo	45.8M	39	1.08M	0.19M	4
Avazu	40.4M	22	2.02M	1.61M	5

For xDeepFM, we directly average the output of the compressed interaction network, and process the ensembled DNN the same as the pure DNN model. For FinalMLP, we average the two-stream outputs respectively.

Hyperparameters. For all experiments, we split the dataset into $8:1:1$ for training/validation/test with random seed 0. We use the Adam optimizer with batch size 2048, learning rate 0.001 and weight decay 1e-6. For base size, we use embedding size 50 for NFwFM considering the pooling, and 10 for all other experiments. We find the hidden size and depth of MLP does not matters the result, and for simplicity, we set hidden size to 400 and set depth to 3 (2 hidden layers and 1 output layer) for all models. We use 4 cross layers for DCNv2 and hidden size 16 for xDeepFM. All experiments use early stopping on validation AUC with patience 3. We repeat each experiment for 3 times with different random initialization. All experiments can be done with a single NVIDIA GeForce RTX 3090.

C.3. Experimental Results

Here we present detailed experimental results with estimated standard deviation. Specifically, we show results on Criteo dataset in Table 5 and Figure 12 and Avazu dataset in Table 6 and Figure 13.

Table 5. Results on Criteo dataset. Higher indicates better.

Model		Criteo				
		base	2x	3x	4x	10x
DNN	SE	$0.81228_{\pm 0.00004}$	$0.81222_{\pm 0.00002}$	$0.81207_{\pm 0.00007}$	$0.81213_{\pm 0.00011}$	$0.81142_{\pm 0.00006}$
	ME		$0.81261_{\pm 0.00004}$	$0.81288_{\pm 0.00015}$	$0.81289_{\pm 0.00007}$	$0.81287_{\pm 0.00005}$
IPNN	SE	$0.81272_{\pm 0.00003}$	$0.81273_{\pm 0.00013}$	$0.81272_{\pm 0.00004}$	$0.81271_{\pm 0.00007}$	$0.81262_{\pm 0.00016}$
	ME		$0.81268_{\pm 0.00009}$	$0.81270_{\pm 0.00002}$	$0.81273_{\pm 0.00015}$	$0.81311_{\pm 0.00008}$
NFwFM	SE	$0.81059_{\pm 0.00012}$	$0.81087_{\pm 0.00008}$	$0.81090_{\pm 0.00012}$	$0.81112_{\pm 0.00011}$	$0.81113_{\pm 0.00022}$
	ME		$0.81128_{\pm 0.00017}$	$0.81153_{\pm 0.00002}$	$0.81171_{\pm 0.00012}$	$0.81210_{\pm 0.00010}$
xDeepFM	SE	$0.81217_{\pm 0.00003}$	$0.81180_{\pm 0.00002}$	$0.81167_{\pm 0.00008}$	$0.81137_{\pm 0.00005}$	$0.81116_{\pm 0.00009}$
	ME		$0.81236_{\pm 0.00006}$	$0.81239_{\pm 0.00022}$	$0.81255_{\pm 0.00011}$	$0.81299_{\pm 0.00009}$
DCNv2	SE	$0.81339_{\pm 0.00002}$	$0.81341_{\pm 0.00007}$	$0.81345_{\pm 0.00009}$	$0.81346_{\pm 0.00011}$	$0.81357_{\pm 0.00004}$
	ME		$0.81348_{\pm 0.00005}$	$0.81361_{\pm 0.00014}$	$0.81382_{\pm 0.00015}$	$0.81385_{\pm 0.00005}$
FinalMLP	SE	$0.81259_{\pm 0.00009}$	$0.81262_{\pm 0.00007}$	$0.81248_{\pm 0.00008}$	$0.81240_{\pm 0.00002}$	$0.81175_{\pm 0.00020}$
	ME		$0.81290_{\pm 0.00017}$	$0.81302_{\pm 0.00005}$	$0.81303_{\pm 0.00004}$	$0.81303_{\pm 0.00012}$

D. More Baseline Methods

We also conduct experiments on AutoInt (Song et al., 2019) and compare the performance under single-embedding and multi-embedding. Due to the limited computational resource, we only scale up the model to 4x on Criteo dataset. Results are shown in Table 7. It is observed that single-embedding suffers from non-scalability while our multi-embedding consistently enhance performance along with the model size, achieving 6e-4 AUC improvement by simply scaling up.

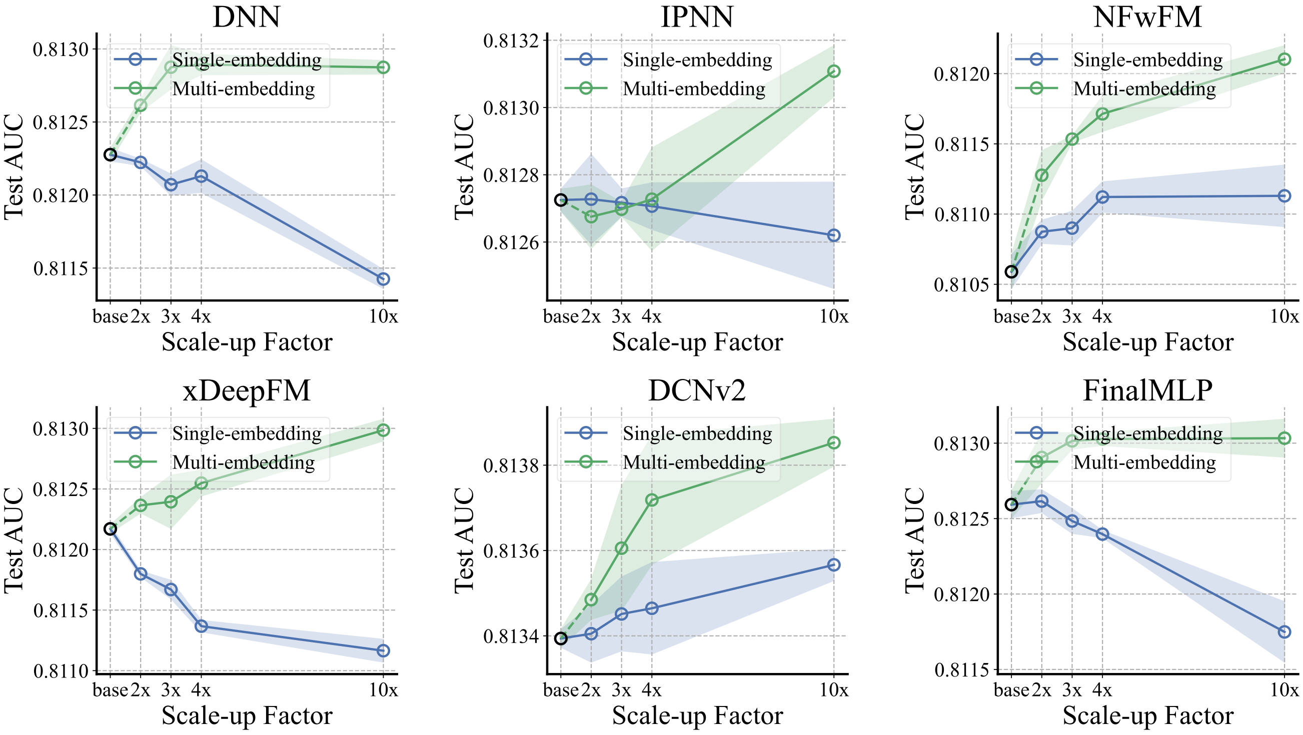


Figure 12. Visualization of scalability on Criteo dataset.

E. Non-Linearity for Multi-Embedding

We have mentioned that the embedding-set-specific feature interaction of multi-embedding should contain non-linearity, otherwise the model will degrade to a single-embedding model. For simplicity, we consider a stronger version of multi-embedding, where the combined features from different embedding sets are concatenated instead of averaged. To further illustrate, consider linear feature interaction modules $I^{(m)}:\left(\mathbb{R}^K\right)^N\rightarrow\mathbb{R}^h$, then we can define a linear feature interaction module $I_{\text{all}}:\left(\mathbb{R}^{MK}\right)^N\rightarrow\mathbb{R}^{Mh}$. For convenience, we denote $[f(i)]_{i=1}^n$ as $[f(1),f(2),...,f(n)]$, and $\boldsymbol{e}_i=[\boldsymbol{e}_i^m]_{m=1}^M$. The form of I_{all} can be formulated by

$$I_{\text{all}}\left(\boldsymbol{e}_1,\boldsymbol{e}_2,...,\boldsymbol{e}_N\right)=\left[I^{(m)}(\boldsymbol{e}_1^{(m)},...,\boldsymbol{e}_N^{(m)})\right]_{m=1}^M.$$

This shows a multi-embedding model is equivalent to a model by concatenating all embedding sets. We will further show that the deduced model with I_{all} is homogeneous to a single-embedding model with size MK, *i.e.*, multi-embedding is similar to single-embedding with linear feature interaction modules. Denote the feature interaction module of single-embedding as I. Despite I_{all} could have different forms from I, we further give three examples to show the homogeneity of I_{all} and I.

DNN. Ignoring the followed MLP, DNN incorporate a non-parametric interaction module by concatenating all fields together. Formally, we have

$$I(\boldsymbol{e}_1,...,\boldsymbol{e}_N)=\left[[\boldsymbol{e}_i^{(m)}]_{m=1}^M\right]_{i=1}^N,$$

$$I_{\text{all}}(\boldsymbol{e}_1,...,\boldsymbol{e}_N)=\left[[\boldsymbol{e}_i^{(m)}]_{i=1}^N\right]_{m=1}^M.$$

In other words, I and I_{all} only differ in a permutation, thus multi-embedding and single-embedding are equivalent.

Table 6. Results on Avazu dataset. Higher indicates better.

Model		Avazu				
		base	2x	3x	4x	10x
DNN	SE	$0.78744_{\pm 0.00008}$	$0.78759_{\pm 0.00011}$	$0.78752_{\pm 0.00031}$	$0.78728_{\pm 0.00036}$	$0.78648_{\pm 0.00013}$
	ME		$0.78805_{\pm 0.00017}$	$0.78826_{\pm 0.00013}$	$0.78862_{\pm 0.00026}$	$0.78884_{\pm 0.00005}$
IPNN	SE	$0.78732_{\pm 0.00020}$	$0.78741_{\pm 0.00022}$	$0.78738_{\pm 0.00010}$	$0.78750_{\pm 0.00007}$	$0.78745_{\pm 0.00018}$
	ME		$0.78806_{\pm 0.00012}$	$0.78868_{\pm 0.00023}$	$0.78902_{\pm 0.00009}$	$0.78949_{\pm 0.00028}$
NFwFM	SE	$0.78684_{\pm 0.00017}$	$0.78757_{\pm 0.00020}$	$0.78783_{\pm 0.00009}$	$0.78794_{\pm 0.00022}$	$0.78799_{\pm 0.00011}$
	ME		$0.78868_{\pm 0.00038}$	$0.78901_{\pm 0.00029}$	$0.78932_{\pm 0.00035}$	$0.78974_{\pm 0.00021}$
xDeepFM	SE	$0.78743_{\pm 0.00009}$	$0.78750_{\pm 0.00025}$	$0.78714_{\pm 0.00030}$	$0.78735_{\pm 0.00004}$	$0.78693_{\pm 0.00050}$
	ME		$0.78848_{\pm 0.00006}$	$0.78886_{\pm 0.00026}$	$0.78894_{\pm 0.00004}$	$0.78927_{\pm 0.00019}$
DCNv2	SE	$0.78786_{\pm 0.00022}$	$0.78835_{\pm 0.00023}$	$0.78854_{\pm 0.00010}$	$0.78852_{\pm 0.00003}$	$0.78856_{\pm 0.00016}$
	ME		$0.78862_{\pm 0.00011}$	$0.78882_{\pm 0.00012}$	$0.78907_{\pm 0.00011}$	$0.78942_{\pm 0.00024}$
FinalMLP	SE	$0.78751_{\pm 0.00026}$	$0.78797_{\pm 0.00019}$	$0.78795_{\pm 0.00017}$	$0.78742_{\pm 0.00015}$	$0.78662_{\pm 0.00025}$
	ME		$0.78821_{\pm 0.00013}$	$0.78831_{\pm 0.00029}$	$0.78836_{\pm 0.00018}$	$0.78830_{\pm 0.00022}$

Table 7. Results on Criteo dataset. Higher indicates better.

AutoInt	base	2x	3x	4x
SE	0.81231	0.81236	0.81216	0.81193
ME		0.81264	0.81285	0.81291

Projected DNN. If we add a linear projection after DNN, then we can split the projection for fields and embedding sets, and derive

$$I(\boldsymbol{e}_1, ..., \boldsymbol{e}_N) = \sum_{i=1}^{N} \sum_{m=1}^{M} \boldsymbol{W}_{i,m} \boldsymbol{e}_i^{(m)},$$

$$I_{\text{all}}(\boldsymbol{e}_1, ..., \boldsymbol{e}_N) = \left[\sum_{i=1}^{N} \boldsymbol{W}_{i,m} \boldsymbol{e}_i^{(m)} \right]_{m=1}^{M}.$$

In other words, I and I_{all} only differ in a summation. Actually if we average the combined features for I_{all} rather than concatenate to restore our proposed version of multi-embedding, then multi-embedding and single-embedding are equivalent by the scalar $1/M$.

DCNv2. DCNv2 incorporates the following feature interaction by

$$I(\boldsymbol{e}_1, ..., \boldsymbol{e}_N) = \left[\boldsymbol{e}_i \odot \sum_{j=1}^{N} \boldsymbol{W}_{j \to i} \boldsymbol{e}_j \right]_{i=1}^{N},$$

thus by splitting $\boldsymbol{W}_{i \to j}$ we have

$$I(\boldsymbol{e}_1, ..., \boldsymbol{e}_N) = \left[[\boldsymbol{e}_i^{(m)} \odot \sum_{j=1}^{N} \sum_{m'=1}^{M} \boldsymbol{W}_{j \to i}^{(m,m')} \boldsymbol{e}_j^{(m')}]_{m=1}^{M} \right]_{i=1}^{N}$$

$$I_{\text{all}}(\boldsymbol{e}_1, ..., \boldsymbol{e}_N) = \left[[\boldsymbol{e}_i^{(m)} \odot \sum_{j=1}^{N} \boldsymbol{W}_{j \to i}^{(m)} \boldsymbol{e}_j^{(m)}]_{i=1}^{N} \right]_{m=1}^{M}.$$

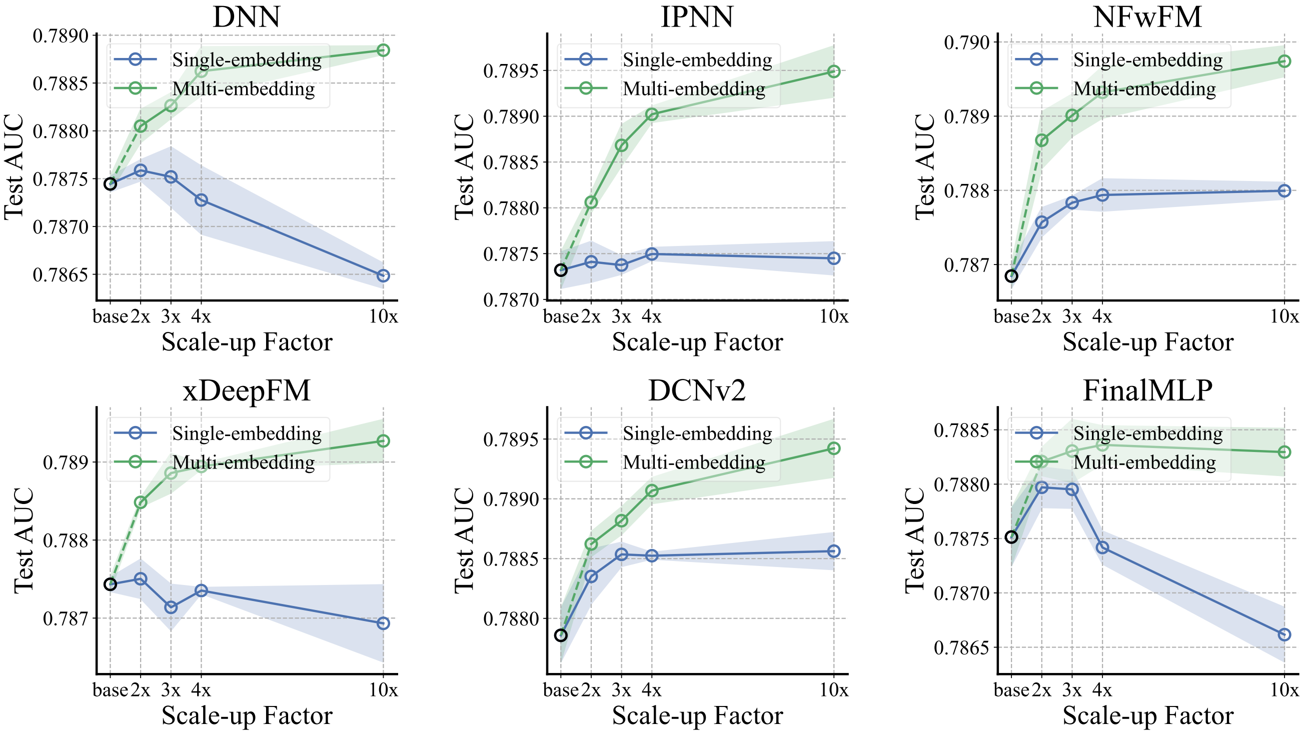


Figure 13. Visualization of scalability on Avazu dataset.

By simply letting $\boldsymbol{W}^{(m,m)} = \boldsymbol{W}^{(m)}$ and $\boldsymbol{W}^{(m,m')} = \boldsymbol{O}$ for $m \neq m'$, we convert a multi-embedding model into a single-embedding model under permutation. Therefore, multi-embedding is a special case of single-embedding for DCNv2.

Summary. In summary, a linear feature interaction module will cause homogenity between single-embedding and multi-embedding. Hence it is necessary to use or introduce non-linearity in feature interaction module.

F. Detailed Explanation of Embedding Diversity

In Section 5.3, we propose to use principal angle to measure embedding set diversity. Here we introduce the motivation and an example. Note that

$$\begin{aligned}
\operatorname{rank}\left(\left[\boldsymbol{E}^{(m)}, \boldsymbol{E}^{(m')}\right]\right) &= \operatorname{rank}\left(\left[\boldsymbol{U}^{(m)}\boldsymbol{\Sigma}^{(m)}\left(\boldsymbol{V}^{(m)}\right)^\top, \boldsymbol{U}^{(m')}\boldsymbol{\Sigma}^{(m')}\left(\boldsymbol{V}^{(m')}\right)^\top\right]\right) \\
&= \operatorname{rank}\left(\left[\boldsymbol{U}^{(m)}\boldsymbol{\Sigma}^{(m)}\left(\boldsymbol{V}^{(m)}\right)^\top, \boldsymbol{U}^{(m')}\boldsymbol{\Sigma}^{(m')}\left(\boldsymbol{V}^{(m')}\right)^\top\right]\begin{bmatrix}\boldsymbol{V}^{(m)} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{V}^{(m')}\end{bmatrix}\right) \\
&= \operatorname{rank}\left(\left[\boldsymbol{U}^{(m)}\boldsymbol{\Sigma}^{(m)}, \boldsymbol{U}^{(m')}\boldsymbol{\Sigma}^{(m')}\right]\right) \\
&= \operatorname{rank}\left(\boldsymbol{U}^{(m)}\boldsymbol{\Sigma}^{(m)}\right) + \operatorname{rank}\left(\boldsymbol{U}^{(m')}\boldsymbol{\Sigma}^{(m')}\right) - \operatorname{rank}\left(\left(\boldsymbol{U}^{(m)}\boldsymbol{\Sigma}^{(m)}\right)^\top \boldsymbol{U}^{(m')}\boldsymbol{\Sigma}^{(m')}\right) \\
&= \operatorname{rank}\left(\boldsymbol{E}^{(m)}\right) + \operatorname{rank}\left(\boldsymbol{E}^{(m')}\right) - \operatorname{rank}\left(\left(\boldsymbol{U}^{(m)}\right)^\top \boldsymbol{U}^{(m')}\right),
\end{aligned}$$

where the second last line are derived from the orthonormality of $\boldsymbol{U}$. Note that

$$\operatorname{rank}\left(\left(\boldsymbol{U}^{(m)}\right)^\top \boldsymbol{U}^{(m')}\right) = \|\cos\left(\boldsymbol{\phi}^{m\leftrightarrow m'}\right)\|_0,$$

and we therefore generalize it to $\frac{1}{K}\|\cos\left(\boldsymbol{\phi}^{m\leftrightarrow m'}\right)\|_1$ to measure similarity, and use $1 -$ similarity as diversity.

Considering the following example of diversity. An embedding with size of 2 are learned as

$$\boldsymbol{E} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}$$

with $\operatorname{rank}(\boldsymbol{E}) = \mathrm{IA}(\boldsymbol{E}) = 2$. If enlarged to size of 4, due to interaction-collapse theory, it is likely to be learned as

$$\boldsymbol{E}^{(1)} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}, \quad \boldsymbol{E}^{(2)} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}, \quad \boldsymbol{E}^{\text{single}} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix},$$

with $\cos\left(\boldsymbol{\phi}^{1\leftrightarrow 2}\right) = (1, 1)$, $\operatorname{rank}(\boldsymbol{E}^{\text{single}}) = \mathrm{IA}(\boldsymbol{E}^{\text{single}}) = 2$, *i.e.*, the enlarged size does not increase information abundance. Here

When using multi-embedding, the embedding sets are possibly learned to be of large diversity, and the overall embedding are learned as

$$\boldsymbol{E}^{(1)} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}, \quad \boldsymbol{E}^{(2)} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \boldsymbol{E}^{\text{multi}} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix},$$

with $\cos\left(\boldsymbol{\phi}^{1\leftrightarrow 2}\right) = (1, 0)$, $\operatorname{rank}(\boldsymbol{E}^{\text{multi}}) = 3$ and $\mathrm{IA}(\boldsymbol{E}^{\text{multi}}) = 1 + \sqrt{2}$, indicating the effectiveness of multi-embedding.

G. Details of Toy Expermient

In this section, we present the detailed settings of the toy experiment. We consider a scenario with $N = 3$ fields and $D_1 = D_2 = 100$. For each $(x_1, x_2) \in \mathcal{X}_1 \times \mathcal{X}_2$, we randomly assign $x_3 \sim \mathcal{U}[\mathcal{X}_3]$, $y \sim \mathcal{U}\{0, 1\}$ and let $(\boldsymbol{x}, y)$ to be one piece of data, thus for different values of D_3, there are always 100^2 pieces of data, and they follow the same distribution when reduced on $\mathcal{X}_1 \times \mathcal{X}_2$. We set $D_3 = 3$ and $D_3 = 100$ to simulate the case with low-information-abundance and high-information-abundance, respectively. We randomly initialize all embedding matrices with normal distribution $\mathcal{N}(0, 1)$, fix $\boldsymbol{E}_2, \boldsymbol{E}_3$ and only optimize $\boldsymbol{E}_1$ during training. We use full-batch SGD with the learning rate of 1. We train the model for 5,000 iterations in total.

H. Empirical analysis on FFM in Evidence I

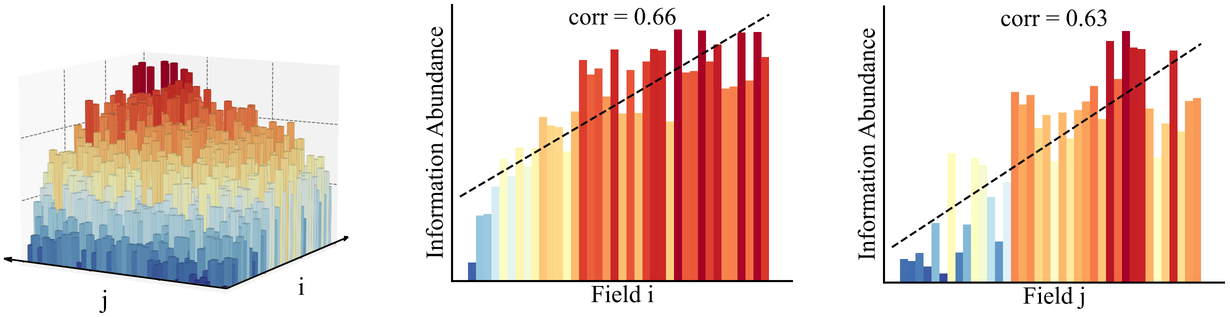
Field-aware factorization machines (FFM) (Juan et al., 2016) split an embedding matrix of field i into multiple sub-embeddings with

$$\boldsymbol{E}_i = \left[\boldsymbol{E}_i^{\to 1}, \boldsymbol{E}_i^{\to 2}, ..., \boldsymbol{E}_i^{\to (i-1)}, \boldsymbol{E}_i^{\to (i+1)}, ..., \boldsymbol{E}_i^{\to N}\right],$$

where sub-embedding $\boldsymbol{E}_i^{\to j} \in \mathbb{R}^{D_i \times K/(N-1)}$ is only used when interacting field i with field j for $j \neq i$. We perform the same experiments as Evidence I, and similarly find that $\mathrm{IA}(\boldsymbol{E}_i^{\to j})$ are co-influenced by both $\mathrm{IA}(\boldsymbol{E}_i)$ and $\mathrm{IA}(\boldsymbol{E}_j)$, as shown in Figure 14. This is amazing in the sense that even using independent embeddings to represent the same field features, these embeddings get different information abundance after learning.

I. Extension of Information Abundance

Our proposed information abundance is a fair metric when two embedding matrices have the same embedding size. To apply the definition between different embedding sizes, some possible extensions include $\frac{\mathrm{IA}(\boldsymbol{E})}{K}$ and $\frac{\mathrm{IA}(\boldsymbol{E})}{\mathbb{E}[\mathrm{IA}(\texttt{randn\_like}(\boldsymbol{E}))]}$, where K stands for the embedding size and $\texttt{randn\_like}(\boldsymbol{E})$ refers to a random matrix underlying the normal distribution with the same shape as $\boldsymbol{E}$. We compare the former one with different embedding sizes in Figure 15, and it is shown that the degree of collapse increases with respect to the embedding size, which is consistent with the observation in Figure 1b



(a) $\mathrm{IA}(\boldsymbol{E}_i^{\rightarrow j})$.

(b) $\sum_{j=1}^{N} \mathrm{IA}(\boldsymbol{E}_i^{\rightarrow j})$.

(c) $\sum_{i=1}^{N} \mathrm{IA}(\boldsymbol{E}_i^{\rightarrow j})$.

Figure 14. Information abundance of sub-embedding matrices for FFM, with field indices sorted by information abundance of corresponding raw embedding matrices. Higher or warmer indicates larger. Similarly, $\mathrm{IA}(\boldsymbol{E}_i^{\rightarrow j})$ are co-influenced by both $\mathrm{IA}(\boldsymbol{E}_i)$ and $\mathrm{IA}(\boldsymbol{E}_j)$.

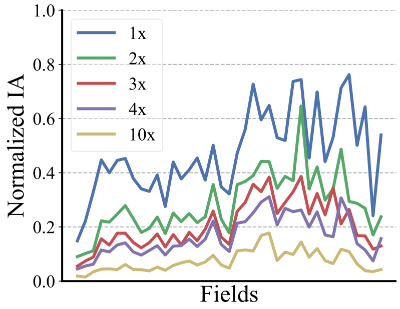


Figure 15. Normalized information abundance $\frac{\mathrm{IA}(\boldsymbol{E})}{K}$ for different embedding sizes on DCNv2.

J. Detailed Explanation of Regularized DCNv2

Regarding Evidence II, we proposed regularization of the weight matrix $\boldsymbol{W}_i^{\rightarrow j}$ to mitigate the collapse caused by the projection $\boldsymbol{W}_i^{\rightarrow j}$ in sub-embeddings. By regularizing $\boldsymbol{W}_i^{\rightarrow j}$ to be a unitary matrix (or the multiplication of unitary matrices), we ensure the preservation of all singular values of the sub-embedding. Consequently, the information abundance of sub-embeddings in regularized DCNv2 is larger than standard DCNv2. We plot the heatmap of information abundance of embeddings and sub-embeddings Figure 16. This clearly demonstrates that regularized DCNv2 exhibits a higher information abundance. Based on our Finding 1, regularized DCNv2 mitigates the problem of embedding collapse by increasing the information abundance of the sub-embeddings that directly interact with the embeddings.

K. How ME Performs When Feature Interaction Is Suppressed?

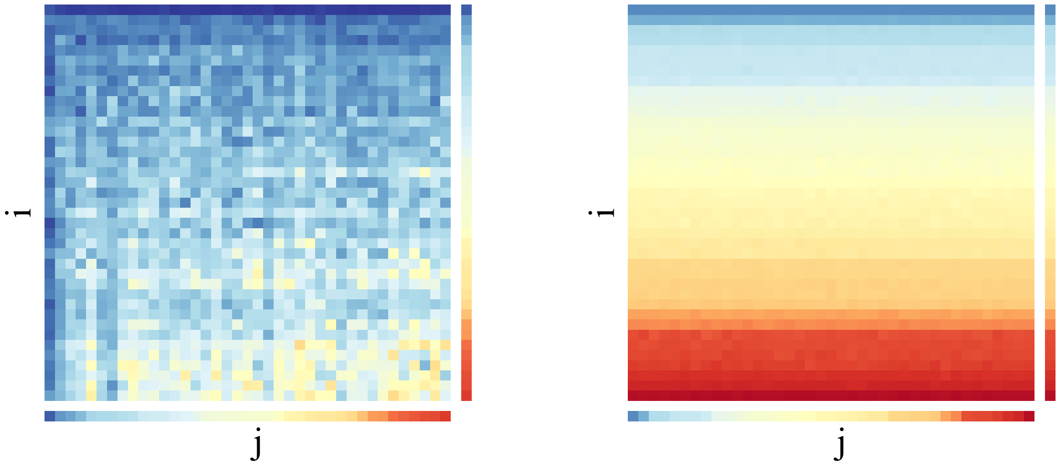
In this section we analysis of ME under models with feature interaction suppressed, as discussed Section 4.2, where SE suffers from overfiting.

Evidence III for ME. We add regularization

$$\ell_{reg} = \sum_{m=1}^{M} \sum_{i=1}^{N} \sum_{j=1}^{N} \left\| (\boldsymbol{W}_{i\rightarrow j}^{(m)})^\top \boldsymbol{W}_{i\rightarrow j}^{(m)} - \lambda_{ij}^{(m)} \boldsymbol{I} \right\|_{\mathrm{F}}^2$$

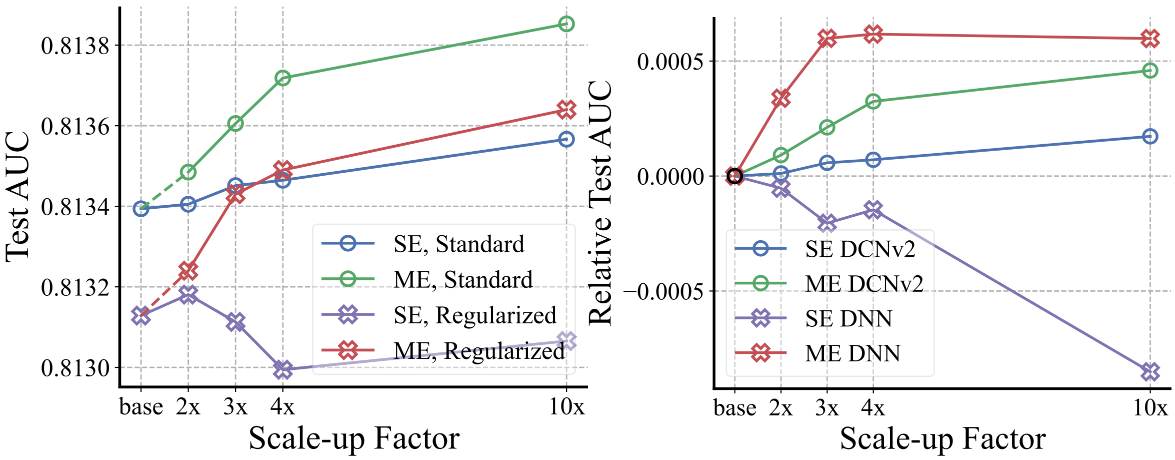
for ME DCNv2 and conduct experiments with different embedding sizes. Results are shown in Figure 17a. Even though the performance is worse than that without regularization, compared with SE, ME still consistently improves the performance with the growth of model size.

Evidence IV for ME. We compare the performance of SE/ME on DNN/DCNv2, as shown in Figure 17b. Compared with SE, ME DNN improves the performance with the growth of model size.



(a) Standard DCNv2 (b) Regularized DCNv2

Figure 16. Visualization of information abundance of embeddings and sub-embeddings for standard and regularized DCNv2, respectively. The rightmost and downmost bars correspond to $\mathrm{IA}(\boldsymbol{E}_i)$ or $\mathrm{IA}(\boldsymbol{E}_j)$. Compared with the standard DCNv2, the regularized can preserve singular values, and resulting in less-collapsed sub-embeddings and finally larger information abundance as in Figure 5a.



(a) Standard vs Regularized (b) DCNv2 vs DNN

Figure 17. Test AUC w.r.t. model size.

Summary: ME offers scalability even though feature interaction is suppressed. For models with feature interaction suppressed such as regularized DCNv2 and DNN, the performance of SE might dropped when enlarging models, since feature interaction provides domain knowledge and large models might suffer from overfitting. Experiments show that these models with ME can properly scale up. Such results are plausible since ME improves scalability by capturing diverse patterns instead of learning with a single interaction pattern.