

Computational Advertising: Recent Advances

Junwei Pan, Zhilin Zhang, Han Zhu

Jian Xu, Jie Jiang, Bo Zheng



Outline

- Part II, Prediction/Ranking
 - Perspectives
 - Feature Interaction
 - Sequential Models
 - Multi-Task and Multi-Domain Learning
 - Large Recommendation Models
 - LLM4Rec

- Part II, Prediction
 - **Perspectives**
 - Feature Interaction
 - Sequential Models
 - Multi-Task and Multi-Domain Learning
 - Large Recommendation Models
 - LLM4Rec

Perspectives and Off-the-shelf Tools

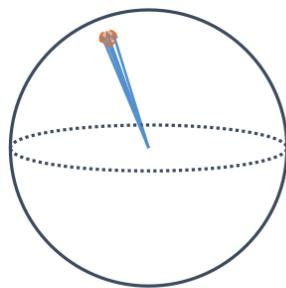
- Do the embeddings fully span the k -dimensional space?
 - Dimensional Robustness/Collapse
 - Tools: Singular Spectrum Analysis
- Does the model capture the correlation that we focus on?
 - Discriminability
 - Tools: Mutual Information Analysis
- Do the model disentangle factors in the data?
 - Disentanglement, Redundancy Reduction
 - Tools: Contradictory Preference Analysis, De-correlation Analysis

Perspective I: Dimensional Robustness

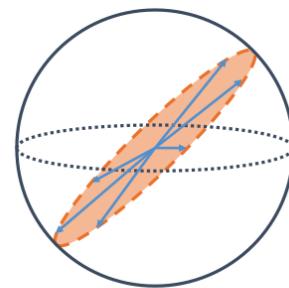
- Embeddings dominates the #params.
- How does the learned embedding span the k -dimensional space?
- What we want to avoid:

Dimensional Collapse

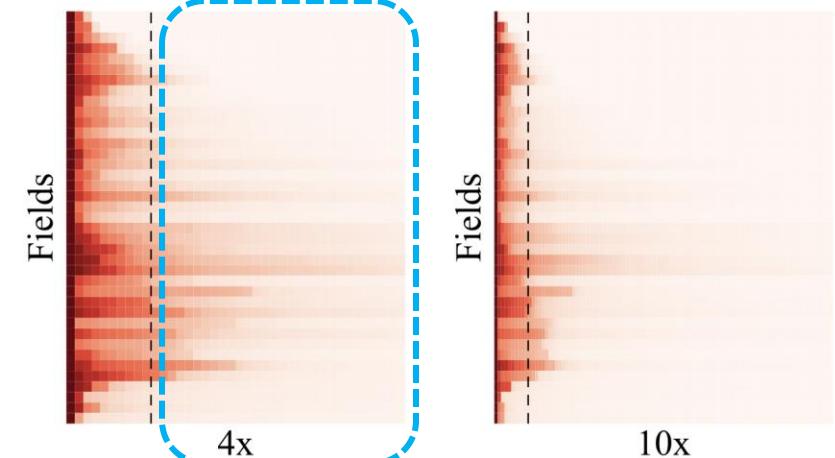
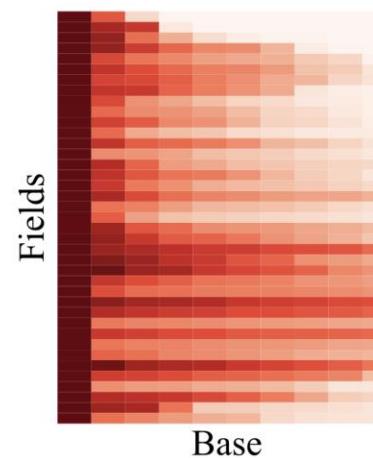
Definition: the embedding vectors end up spanning a **lower-dimensional subspace** instead of the entire available space



(b) complete collapse



(c) dimensional collapse



Singular Spectrum Analysis

$$E = U \Sigma V^T$$

E is the **embedding matrix**, it can be both sample-wise or feature-ID-wise. After **SVD**, The singular values Σ reflects the **scaling in each space dimension**.

Perspective II: Discriminability

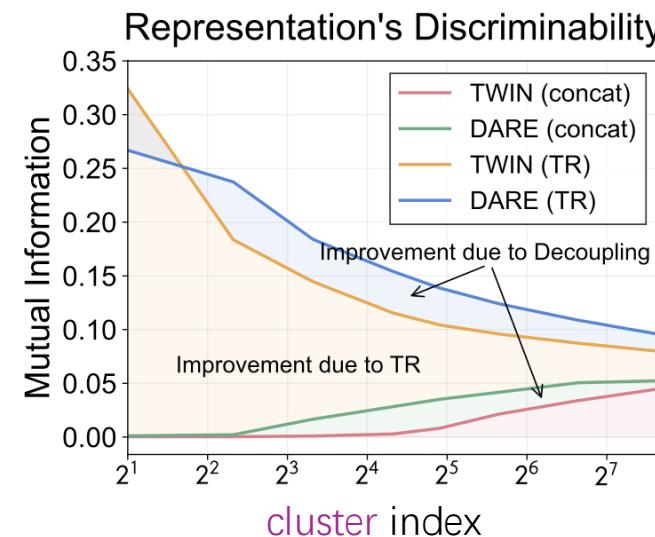
Whether the model capture **correlations** in hidden factors (e.g. layers) or explicit factors (e.g. features.)

Hidden factors

Discretized Mutual Information

$$\text{MI}(\text{Discret}(h), Y)$$

Mutual information between the discretized factors $\text{Discret}(h)$, i.e., **clusters**, and the target Y

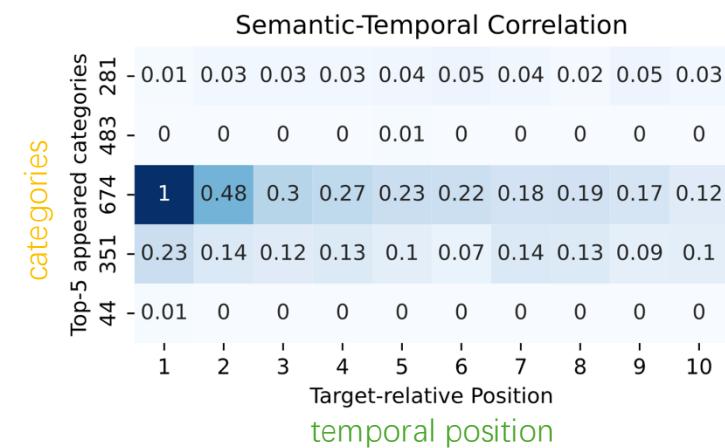


Explicit factors

Feature-wise Mutual Information

$$\text{MI}(X', Y)$$

Mutual information between specified features, e.g., **categories**, or **temporal position**, and the target Y .



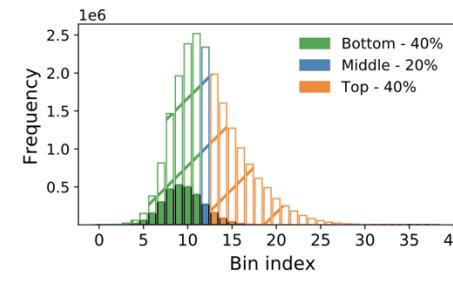
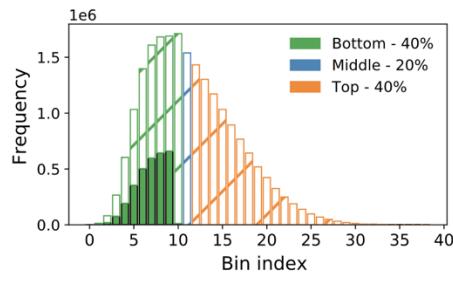
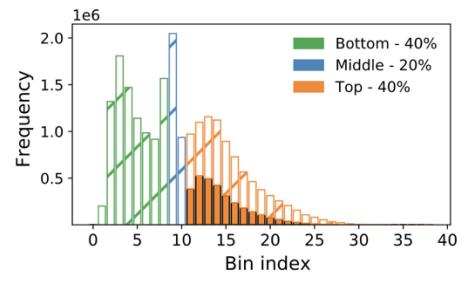
Perspective III: Disentanglement

Whether the model succeed in **decouple conflicting** or **redundant** interests?

For multi-task learning:

Contradictory Preference Analysis

- Pick up conflicting user-item pairs based on **single task** embeddings.
- Plot distance distribution of these user-item pairs using MTL embs.

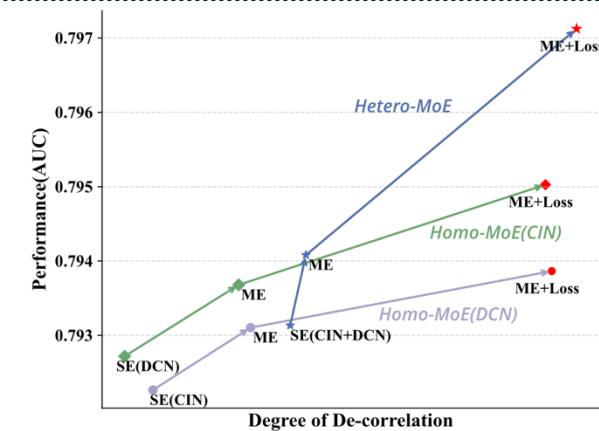


For MoE:

Inter-Expert De-Correlation

$$\sum_i \sum_j \text{Corr}(Ex_i, Ex_j)$$

The correlation between expert outputs Ex_i and Ex_j in the MoE architecture.



- Part II, Prediction
 - Perspectives
 - **Feature Interaction**
 - Sequential Models
 - Multi-Task and Multi-Domain Learning
 - Large Recommendation Models
 - LLM4Rec

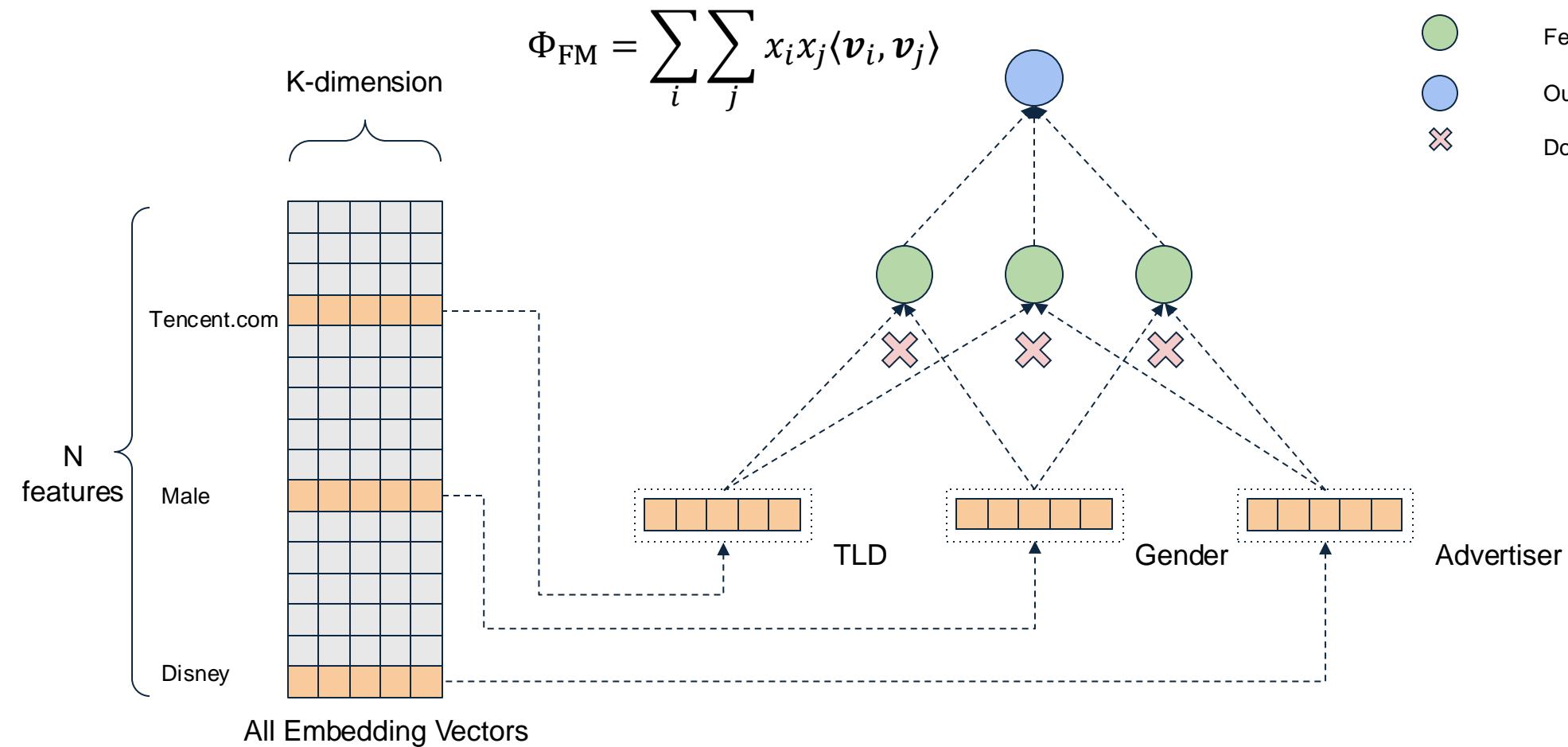
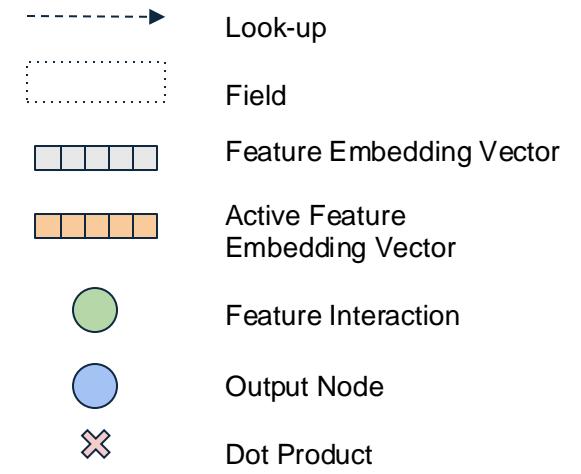
FuxiCTR / BARS - Benchmark

Feature vector \mathbf{x}											Target y		
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...	0.3	$y^{(1)}$		
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...	0.3	$y^{(2)}$		
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...	0.3	$y^{(3)}$		
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...	0	$y^{(4)}$		
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...	0	$y^{(5)}$		
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...	0.5	$y^{(6)}$		
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...	0.5			
A	B	C	...	TI	NH	SW	ST	...	TI	NH	SW	ST	...
User	Movie	Other Movies rated	Time	Last Movie rated									

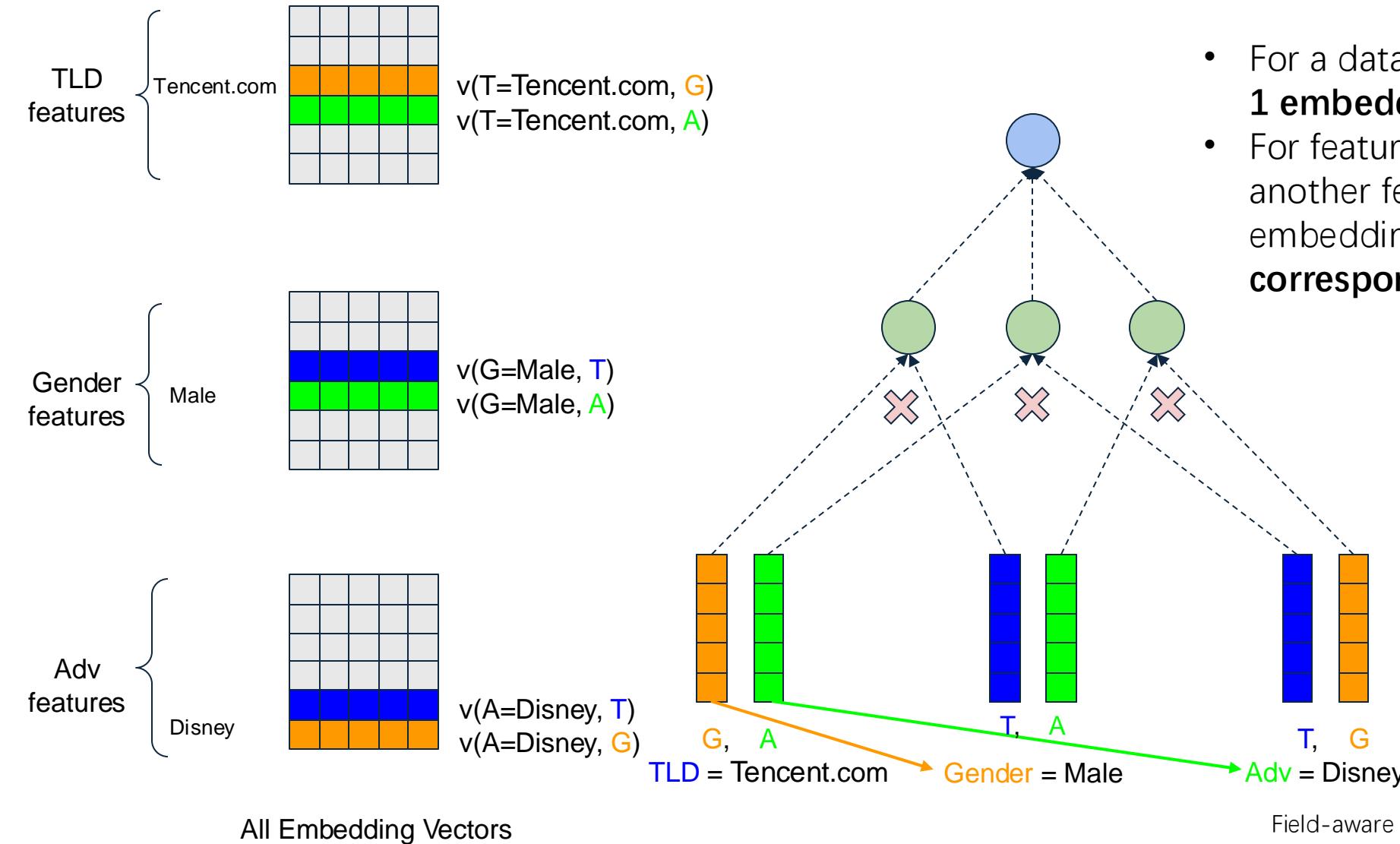
- Multi-field categorical data
- Tabular data

Year	Model	Criteo_x4		Avazu_x4		KKBox_x1	
		Logloss ($\times 10^{-2}$)	AUC(%)	Logloss ($\times 10^{-2}$)	AUC(%)	Logloss ($\times 10^{-2}$)	AUC(%)
2007	LR	45.68	79.34	38.15	77.75	57.46	76.78
2010	FM [77]	44.31	80.86	37.54	78.87	50.75	82.91
2015	CCPM [58]	44.15	81.04	37.45	78.92	50.13	83.72
2016	FFM [46]	44.07	81.13	37.20	79.31	49.74	83.76
2016	HOFM [20]	44.11	81.07	37.54	78.91	50.48	83.15
2016	PNM [75]	43.78	81.42	37.12(4)	79.44(3)	47.93	85.15
2016	DNN [27]	43.80	81.40	37.22	79.28	48.11	85.01
2016	Wide&Deep [25]	43.77	81.42	37.20	79.29	48.52	85.04
2016	DeepCrossing [84]	43.84	81.35	37.21	79.30	47.99	84.95
2017	NFM [39]	44.24	80.93	37.43	78.94	51.02	82.85
2017	AFM [112]	44.55	80.60	37.93	78.23	52.41	81.75
2017	DeepFM [37]	43.76(3)	81.43(5)	37.19	79.30	47.85	85.31(4)
2017	CrossNet [103]	44.56	80.60	37.79	78.40	52.83	81.16
2017	DCN [103]	43.76(3)	81.44(4)	37.19	79.31	47.66(1)	85.31(4)
2018	FwFM [70]	44.08	81.12	37.44	79.07	49.71	84.06
2018	CIN [55]	43.94	81.27	37.42	78.94	49.09	84.26
2018	xDeepFM [55]	43.76(3)	81.43(5)	37.18	79.33	47.72(2)	85.35(2)
2019	FiGNN [54]	43.83	81.38	37.36	79.15	48.96	84.72
2019	FiBiNET [43]	43.87	81.31	37.05(2)	79.53(2)	48.14	84.99
2019	AutoInt [90]	43.99	81.19	37.45	78.91	49.19	84.36
2019	AutoInt+ [90]	43.90	81.32	37.46	79.02	47.73(3)	85.34(3)
2019	HFM [98]	44.24	80.95	37.57	78.79	49.70	83.92
2019	HFM+ [98]	43.92	81.27	37.14	79.44(3)	47.81(5)	85.21
2019	FGCNN [57]	43.98	81.21	37.11(3)	79.44(3)	48.01	85.22
2020	LorentzFM [113]	44.34	80.83	37.56	78.85	51.88	82.02
2020	InterHAT [53]	44.14	81.04	37.49	78.82	48.63	84.59
2020	AFN [26]	44.02	81.15	37.40	79.07	49.10	84.26
2020	AFN+ [26]	43.84	81.38	37.26	79.29	48.42	84.89
2020	DeepIM [120]	43.75(2)	81.46(2)	37.16(5)	79.35	47.75(4)	85.37(1)
2020	ONN [118]	43.72(1)	81.48(1)	36.83(1)	79.92(1)	48.56	84.98
2021	FmFM [95]	43.97	81.24	37.47	79.01	49.46	84.07
2021	DCN-V2 [104]	43.75(2)	81.45(3)	37.19	79.31	47.87	85.31(4)

Factorization Machines



FFM (Field-aware Factorization Machines)



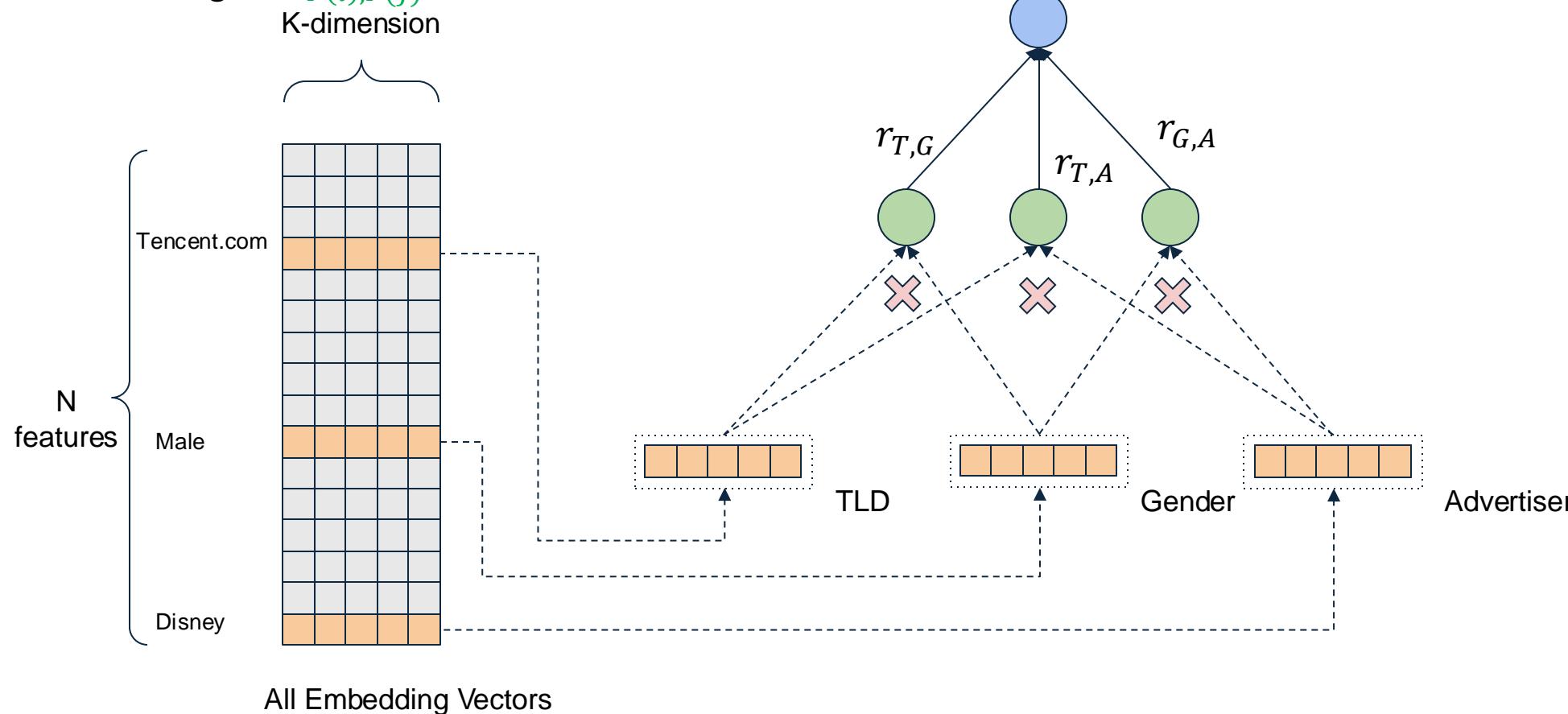
$$\Phi_{\text{FFM}} = \sum_i \sum_j x_i x_j \langle \mathbf{v}_{i,F(j)}, \mathbf{v}_{j,F(i)} \rangle$$

- For a dataset with M fields, FFM learns **M-1 embeddings** for each feature.
- For feature i, when being interacted with another feature j, among the M-1 embeddings, it chooses **the one corresponding to the field of j**, i.e., $\mathbf{v}_{i,F(j)}$

FwFM (Field-weighted Factorization Machines)

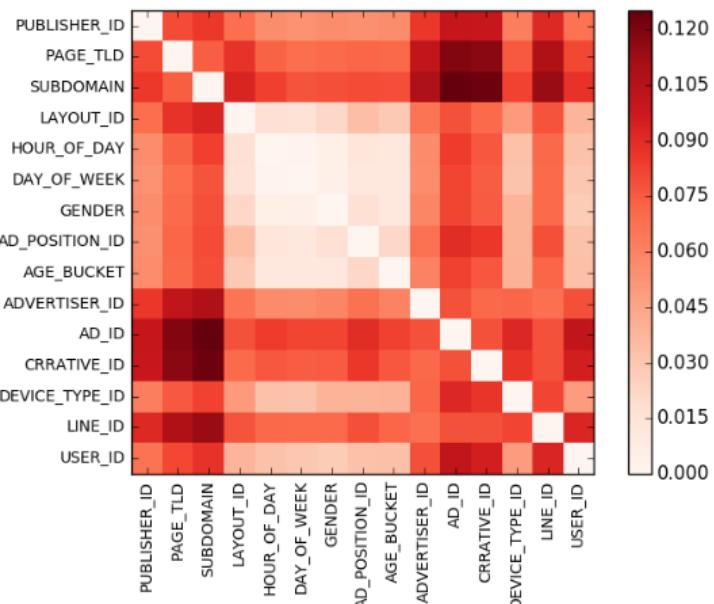
FFM is field-aware by using field-specific embeddings within interaction. Another efficient way is to assign field-wise weights $\textcolor{green}{r}_{F(i),F(j)}$

$$\Phi_{\text{FwFM}} = \sum_i \sum_j x_i x_j \langle \mathbf{v}_i, \mathbf{v}_j \rangle \textcolor{green}{r}_{F(i),F(j)}$$



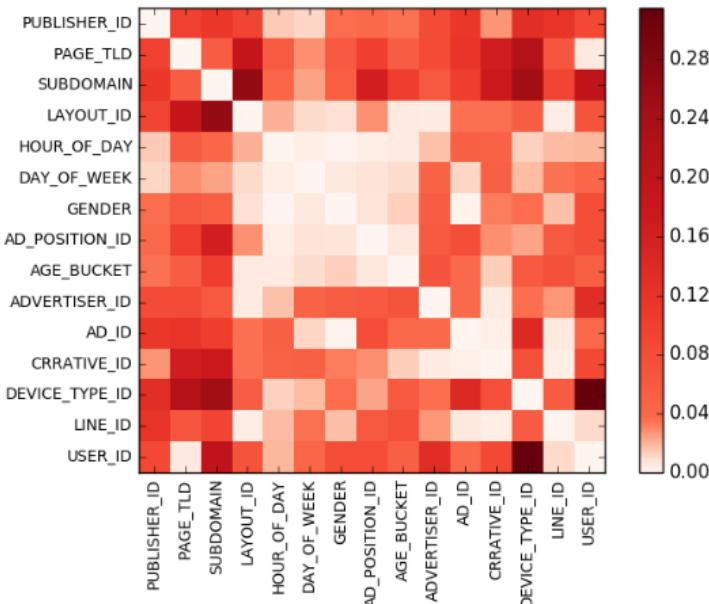
FwFM – A Discriminability Perspective

Ground-truth field-wise
mutual information.



(a) Heat map of mutual informations between field pairs and labels.

Learned field-wise correlation.



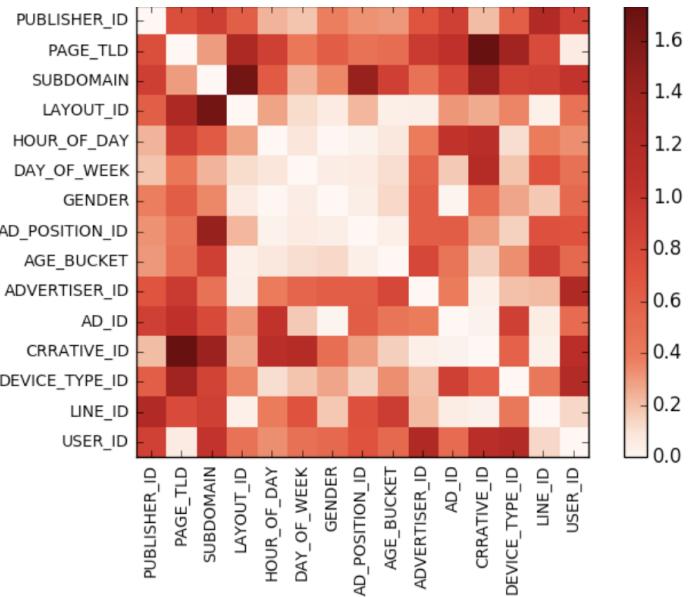
(d) Heat map of learned field interaction strengths of FwFMs. Pearson correlation coefficient with mutual informations: 0.4271.

$$I(i,j) = \frac{\sum_{(i,j) \in (F_k, F_l)} I(i,j) \cdot \#(i,j)}{\sum_{(i,j) \in (F_k, F_l)} \#(i,j)}$$

$$I(i,j) = |\langle \mathbf{v}_i, \mathbf{v}_j \rangle \cdot r_{F_k, F_l}|$$

$$\Phi_{\text{FwFM}} = \sum_i \sum_j x_i x_j \langle \mathbf{v}_i, \mathbf{v}_j \rangle r_{F(i), F(j)}$$

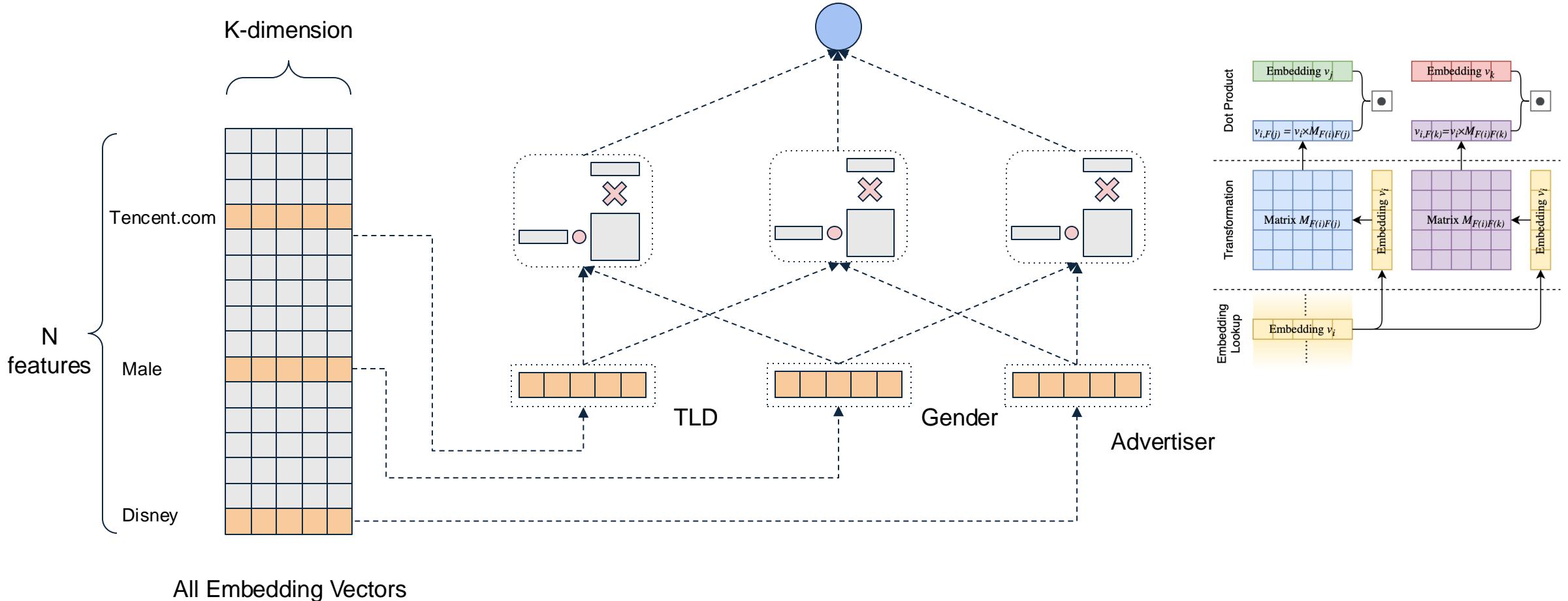
Field-pair weights $|r_{F(i), F(j)}|$



(a) Heat map of field interaction weight r_{F_k, F_l} . Its Pearson correlation coefficient with mutual information is 0.5554.

FmFM (Field-matrixed Factorization Machines) $\Phi_{\text{FmFM}} = \sum_i \sum_j x_i x_j (\mathbf{v}_i M_{F(i), F(j)}) \odot \mathbf{v}_j$

Involve field information with field-pair-wise projection matrix $M_{F(i), F(j)}$ within the interaction.



Feature Interaction Evolution

-A Dimensional Collapse Perspective

With performance enhances from FM, FwFM to FmFM, the **projection matrix within interaction** also evolves, from identity matrix, scaled identity matrix to full matrix.

Identity matrix

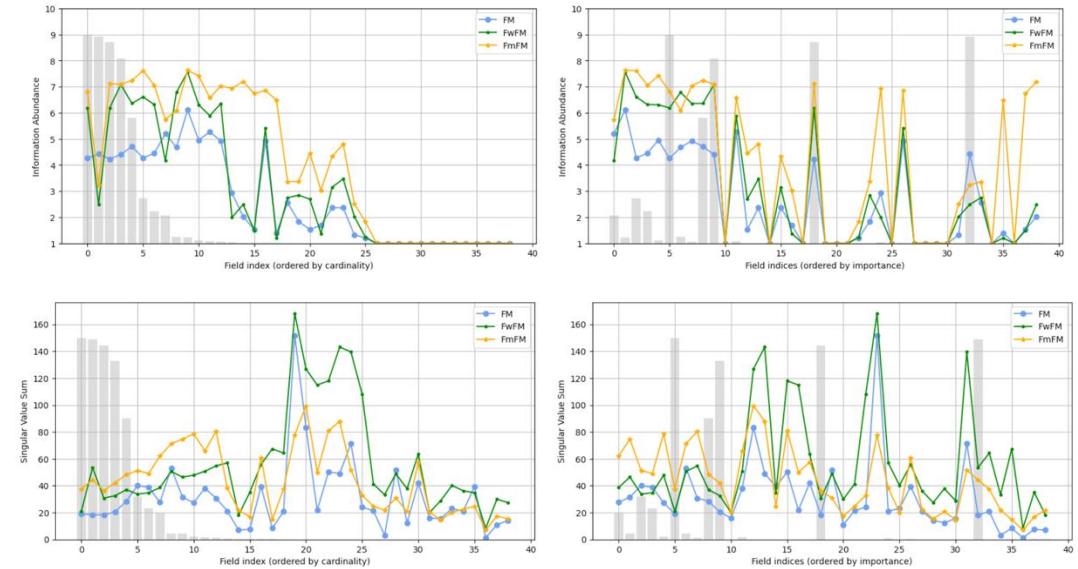
$$\text{FM: } \langle v_i, v_j \rangle = \langle v_i \mathbf{I}, v_j \rangle$$

Scaled identity matrix

$$\text{FwFM: } \langle v_i, v_j \rangle r_{F(i),F(j)} = \langle v_i \mathbf{I} \cdot \mathbf{r}_{F(i),F(j)}, v_j \rangle$$

Full matrix

$$\text{FmFM: } \langle v_i \mathbf{M}_{F(i),F(j)}, v_j \rangle$$



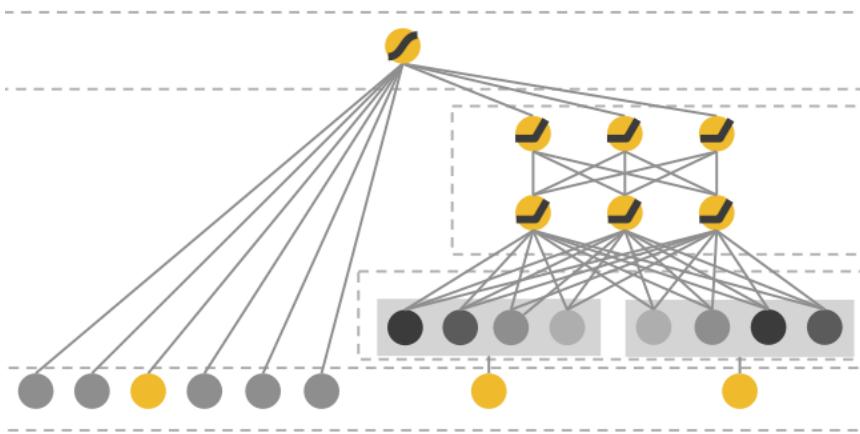
There is a trend of more dimensional robust embeddings from FM, FwFM to FmFM

Among the 2nd-order interaction models, the more complicated the projection matrix (i.e., from **identity**, **scaled identity**, to **full matrix**) within the feature Interaction Function, the more robust (less collapsed) the learned embeddings regarding both information abundance and singular value sum.

Wide & Deep, DeepFM

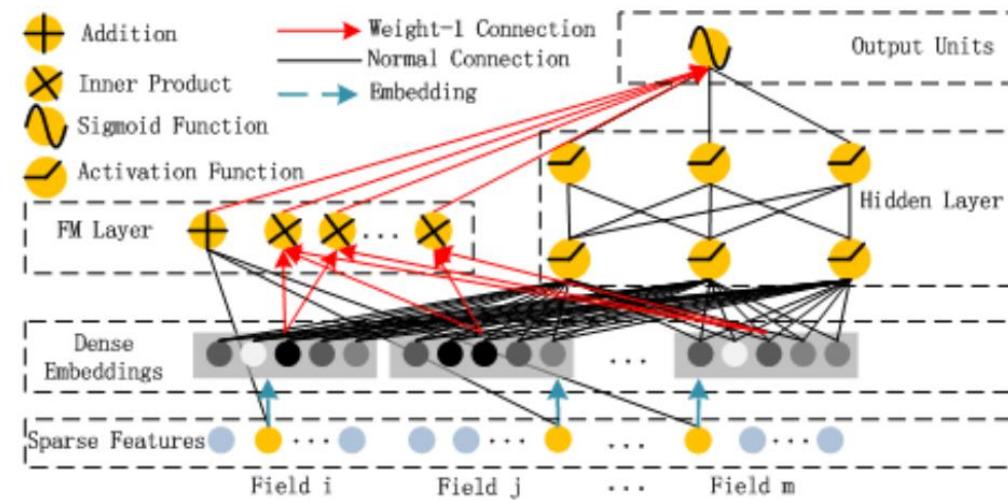
Follow a Wide & Deep framework

- Wide: 2nd-order explicit interactions.
- Deep: implicit high-order interactions.



Wide & Deep

- Wide: Logistic Regression
- Deep: DNN



DeepFM

- Wide: FM
- Deep: DNN

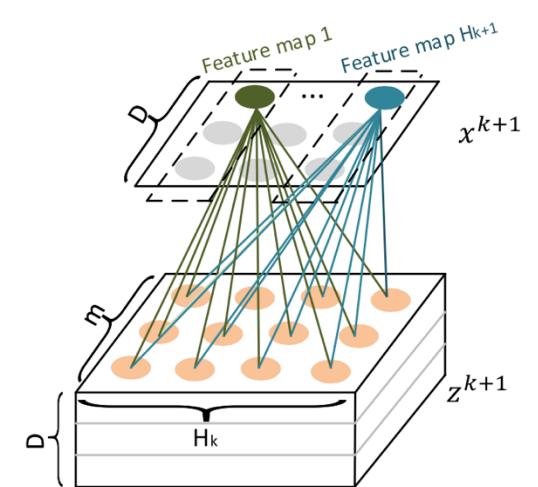
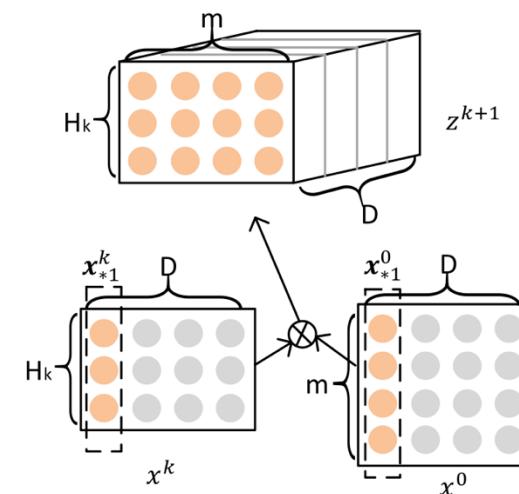
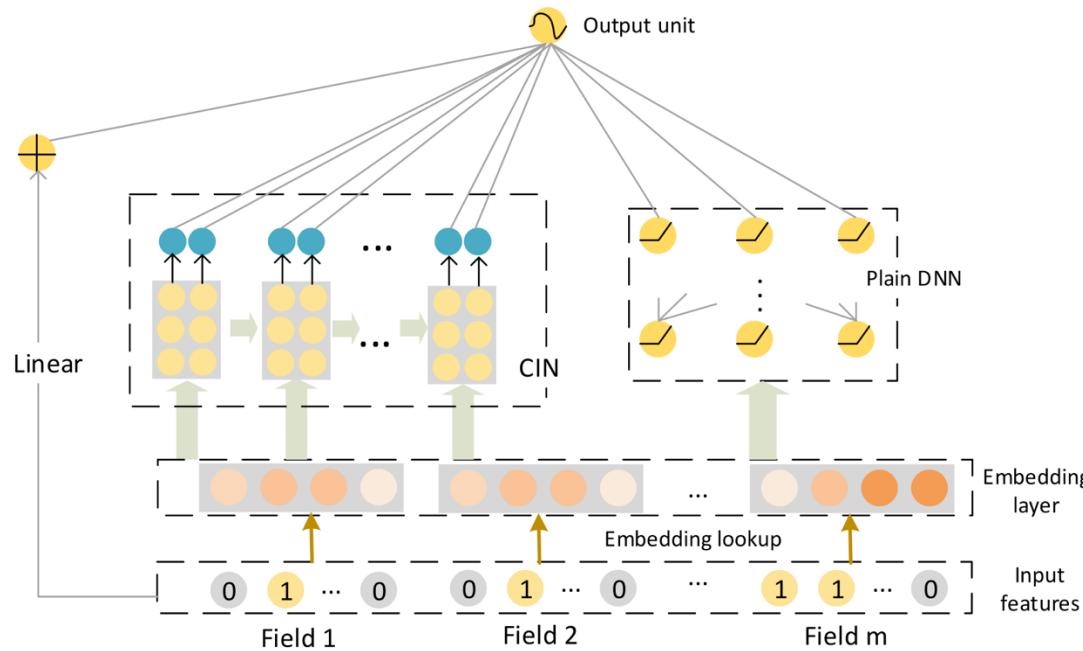
xDeepFM

Follow a Explicit and Implicit framework

- Explicit: Compressed Interaction Network (CIN).
- Implicit: DNN.

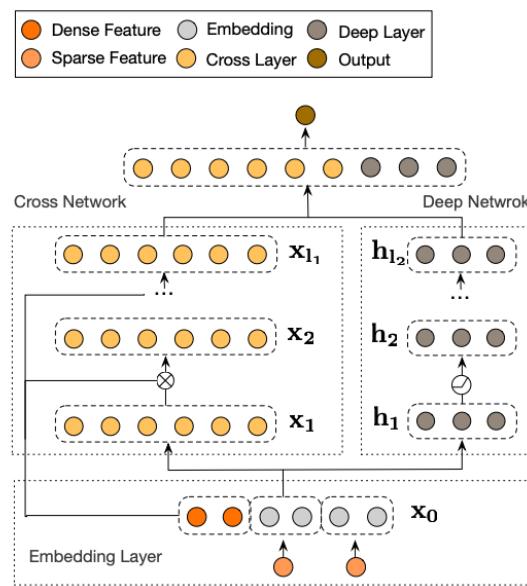
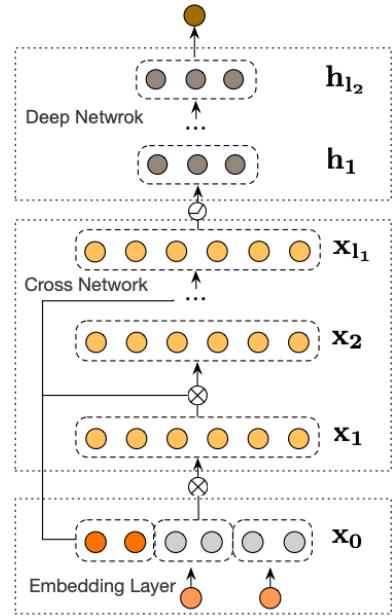
$$\mathbf{x}_k = \mathbf{x}_0 \mathbf{x}_{k-1}^T \mathbf{w}_k + \mathbf{b}_k + \mathbf{x}_{k-1}$$

field-pair-wise weights, similar to the weights $r_{F(i), F(j)}$ in FwFM



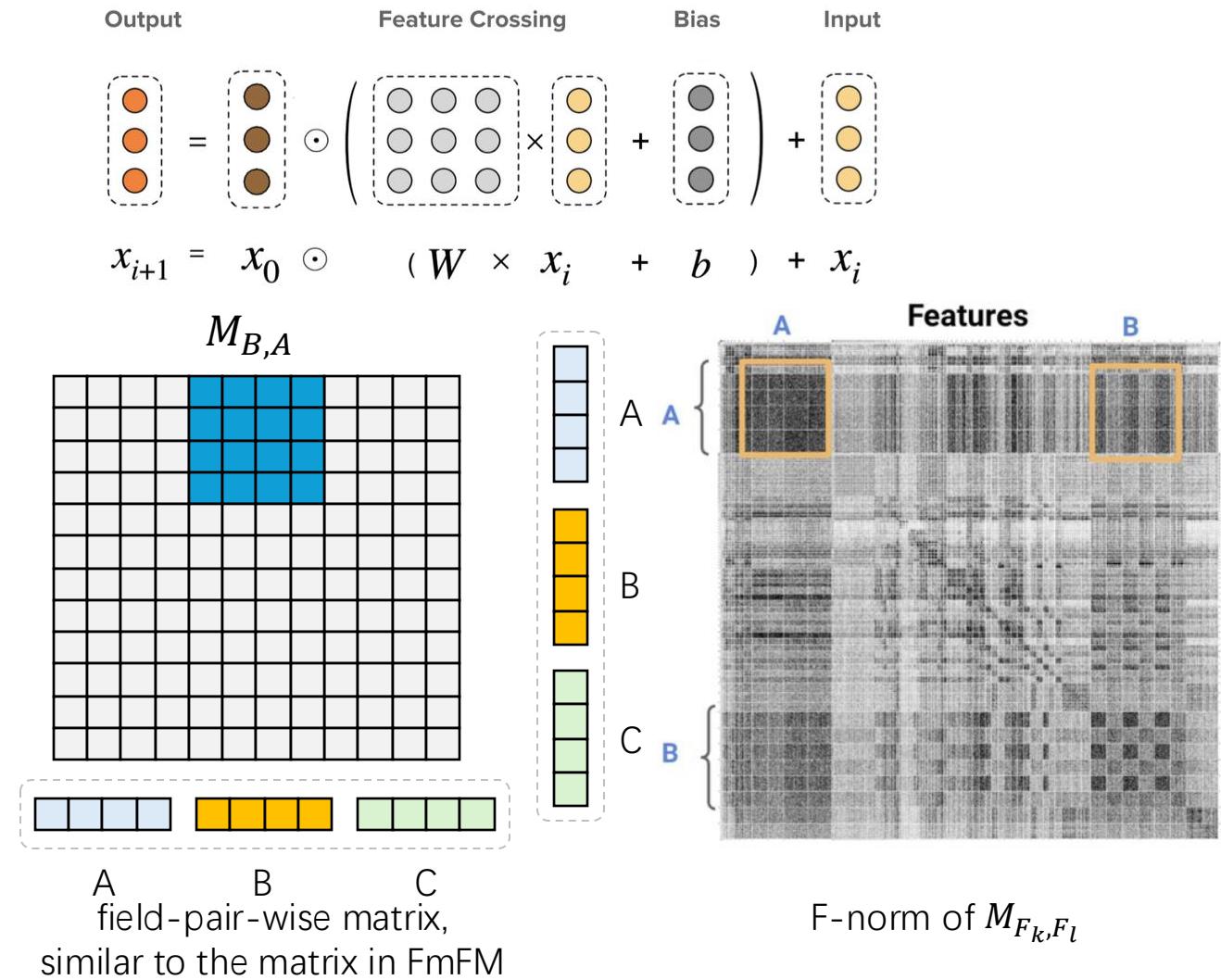
DCN V2

- Explicit: CrossNet
- Implicit: DNN.



(a) Stacked

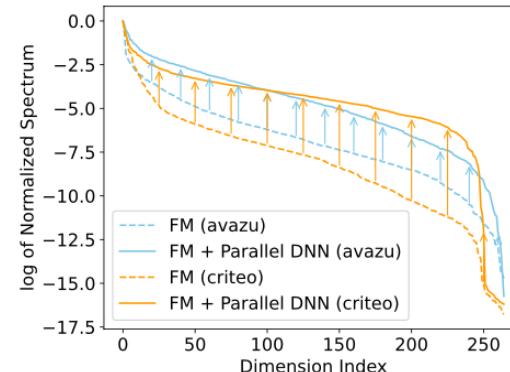
(b) Parallel



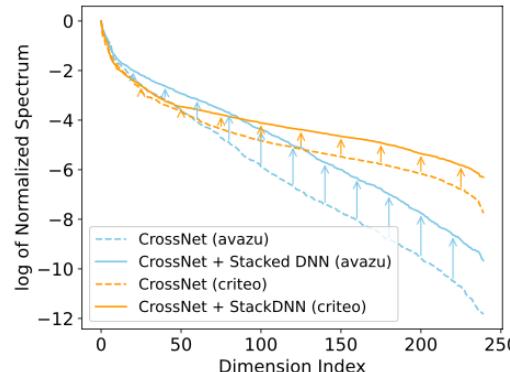
Role of DNN

- Why hybrid models work better?
- Does DNN really learn implicit interaction?
- Rendle proves that **DNN is hard to learn dot products.**

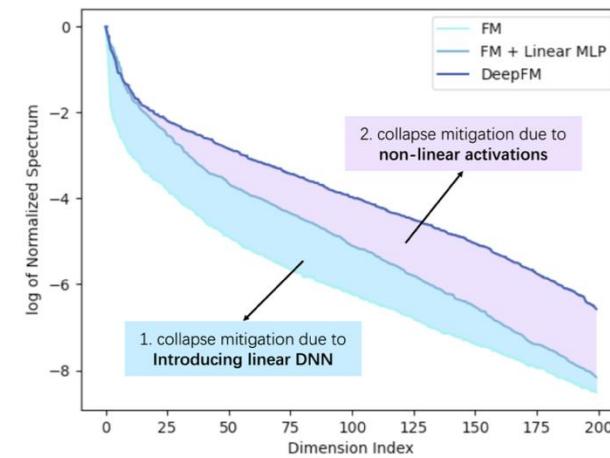
The **parallel and stacked DNN** mitigates the **dimensional collapse** of embeddings in recommendation models.



(a) Parallel DNN



(b) Stacked DNN



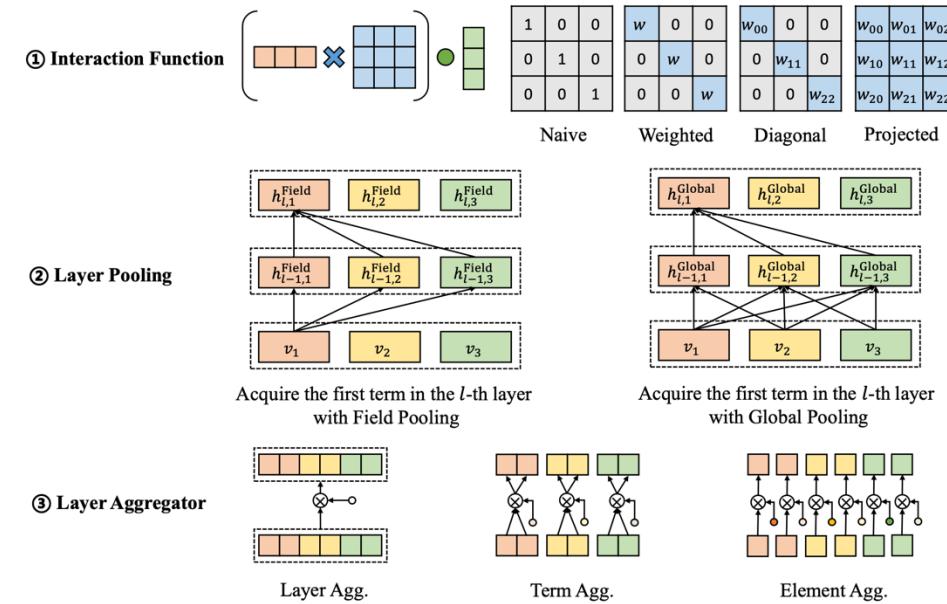
Both **linear DNN** and **non-linear activations** can mitigate **dimensional collapse**.

IPA: Unified Framework

A unified framework with three components can summarize most existing feature interaction models:

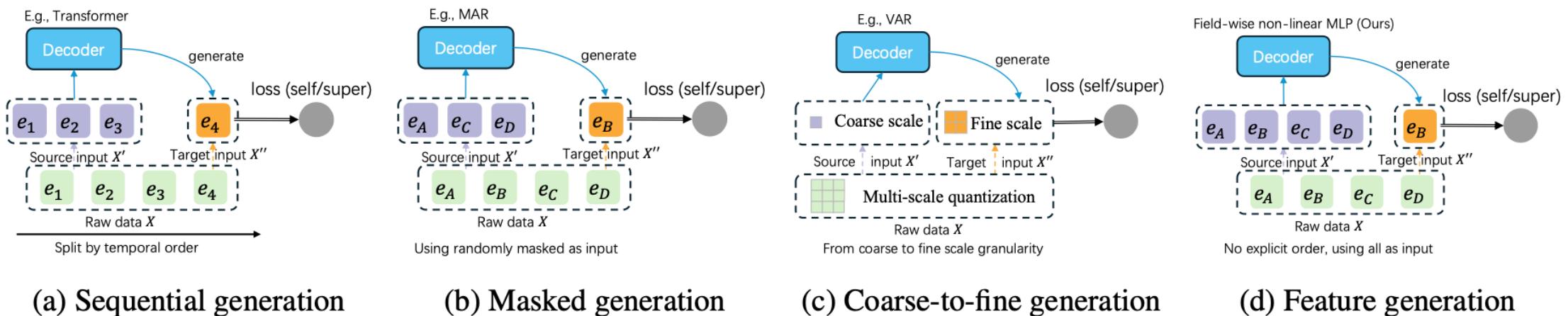
- Feature interaction function: $\langle v_i \mathcal{I}, v_j \rangle$, $\langle v_i \mathcal{I} \cdot r_{F(i), F(j)}, v_j \rangle$, $\langle v_i D(\{\sigma_i\}), v_j \rangle$, $\langle v_i M_{F(i), F(j)}, v_j \rangle$
- Layer pooling: **Field Pooling, Global Pooling**
- Layer aggregation: **direct/layer/term;element**-aggregation

Model	Interaction Function	Layer Pooling	Layer Aggregator	Criteo			Avazu		
				L	AUC	Logloss	O	AUC	Logloss
FM	naive	field/global	direct	2	0.8009(2e-4)	0.4507(4e-4)	2	0.7758(1e-4)	0.3821(2e-4)
DeepFM	naive	field/global	direct	2	0.8122(1e-4)	0.4399(2e-4)	2	0.7899(5e-4)	0.3741(5e-4)
HOFM	naive	field/global	direct	4	0.8040(3e-4)	0.4479(3e-4)	5	0.7781(8e-4)	0.3807(6e-4)
FwFM	weighted	field/global	direct	2	0.8095(2e-4)	0.4423(3e-4)	2	0.7854(4e-4)	0.3768(4e-4)
xDeepFM	weighted	global	term	4	0.8119(2e-4)	0.4401(2e-4)	5	0.7897(7e-4)	0.3741(8e-4)
FvFM	diagonal	field/global	direct	2	0.8103(2e-4)	0.4415(5e-3)	2	0.7870(3e-4)	0.3754(4e-4)
FmFM	projected	field/global	direct	2	0.8115(3e-4)	0.4403(3e-4)	2	0.7882(5e-4)	0.3750(3e-4)
FiBiNet	projected	field/global	direct	2	0.8113(2e-4)	0.4405(2e-4)	2	0.7907(4e-4)	0.3738(5e-4)
DCN V2	projected	field'	direct	4	0.8137(3e-4)	0.4384(4e-4)	5	0.7917(1e-4)	0.3729(4e-4)
WFL	weighted	field	layer	4	0.8124(2e-4)	0.4394(2e-4)	5	0.7891(3e-4)	0.3746(5e-4)
DFL	diagonal	field	layer	4	0.8123(1e-4)	0.4395(2e-4)	5	0.7903(9e-4)	0.3740(7e-4)
PFL	projected	field	layer	4	0.8138(3e-4)	0.4381(4e-4)	5	0.7916(4e-4)	0.3731(4e-4)
PFT	projected	field	term	4	0.8138(3e-4)	0.4381(3e-4)	5	0.7904(4e-4)	0.3738(6e-4)
PFE	projected	field	element	4	0.8138(3e-4)	0.4381(4e-4)	5	0.7907(2e-4)	0.3735(3e-4)
PFD	projected	field	direct	4	0.8131(4e-4)	0.4388(4e-4)	5	0.7912(5e-4)	0.3732(4e-4)



Feature Generation

- All existing feature interaction models are under the **discriminative paradigm** that models $P(Y|X)$.
- We aim to model $P(X)$. What's $P(X)$ in feature interaction?
- **Predict each feature embedding** based on a **decoder network** over the whole input



Supervised Feature Generation framework for CTR models, shifting from the **discriminative "feature interaction"** paradigm to the **generative "feature generation"** paradigm.

Feature Interaction v.s. Feature Generation

$$\Phi_{\text{FM}}^{\text{DIS}} = \sum_i \sum_j x_i x_j (\mathbf{v}_j \odot \mathbf{v}_i)$$

interaction

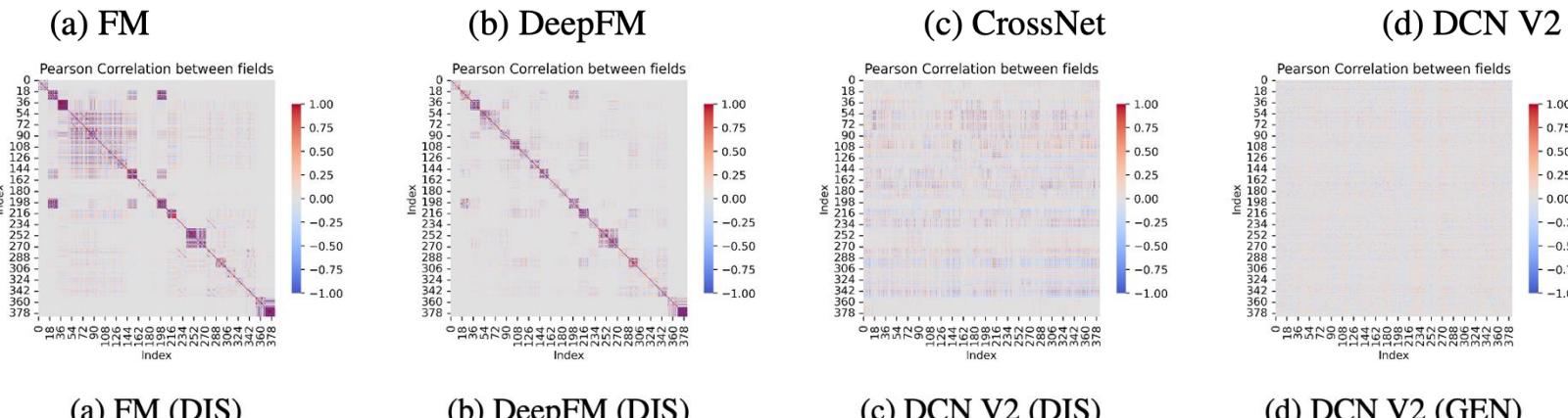
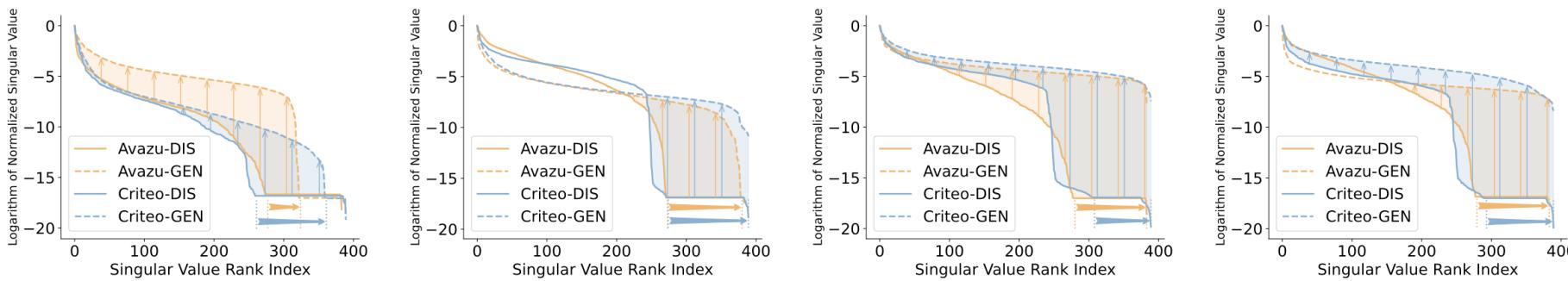
$$\Phi_{\text{FM}}^{\text{GEN}} = \sum_i \sum_j x_i x_j (\sigma([\mathbf{v}] \cdot W_{F(j)}) \odot \mathbf{v}_i)$$

decoder

Model	Discriminative	Generative
FM	$\sum_{i,j=1}^N \mathbf{v}_j \odot \mathbf{v}_i$	$\sum_{i,j=1}^N \sigma([\mathbf{v}] \cdot W_{F(j)}) \odot \mathbf{v}_i$
FmFM	$\sum_{i,j=1}^N \mathbf{v}_j \odot [\mathbf{v}_i \cdot M_{F(i) \rightarrow F(j)}]$	$\sum_{i,j=1}^N \sigma([\mathbf{v}] \cdot W_{F(j)}) \odot [\mathbf{v}_i \cdot M_{F(i) \rightarrow F(j)}]$
CrossNet V2	$\sum_{l=1}^L \sum_{i,j=1}^N \mathbf{v}_j^0 \odot (\mathbf{v}_i^l \cdot M_{F(i) \rightarrow F(j)}^l)$	$\sum_{l=1}^L \sum_{i,j=1}^N \sigma([\mathbf{v}]^l \cdot W_{F(j)}^l) \odot (\mathbf{v}_i^l \cdot M_{F(i) \rightarrow F(j)}^l)$
DeepFM	$\sum_{i,j=1}^N \mathbf{v}_j \odot \mathbf{v}_i + \text{DNN}([\mathbf{v}])$	$\sum_{i,j=1}^N \sigma([\mathbf{v}] \cdot W_{F(j)}) \odot \mathbf{v}_i$
xDeepFM	$\sum_{l=1}^L \sum_{i,j=1}^N \text{Conv}^l(\mathbf{v}_j^0 \odot \mathbf{v}_i^l) + \text{DNN}([\mathbf{v}])$	$\sum_{l=1}^L \sum_{i,j=1}^N \text{Conv}^l(\sigma([\mathbf{v}]^l \cdot W_{F(j)}^l) \odot \mathbf{v}_i^l) + \text{DNN}([\mathbf{v}])$
IPNN	$\text{DNN}([\mathbf{v}], \sum_{i,j=1}^N \mathbf{v}_j \odot \mathbf{v}_i)$	$\text{DNN}([\mathbf{v}], \sum_{i,j=1}^N \sigma([\mathbf{v}] \cdot W_{F(j)}) \odot \mathbf{v}_i)$
DCN V2	$\sum_{l=1}^L \sum_{i,j=1}^N \mathbf{v}_j^0 \odot (\mathbf{v}_i^l \cdot M_{F(i) \rightarrow F(j)}^l) + \text{DNN}([\mathbf{v}])$	$\sum_{l=1}^L \sum_{i,j=1}^N \sigma([\mathbf{v}]^l \cdot W_{F(j)}^l) \odot (\mathbf{v}_i^l \cdot M_{F(i) \rightarrow F(j)}^l) + \text{DNN}([\mathbf{v}])$

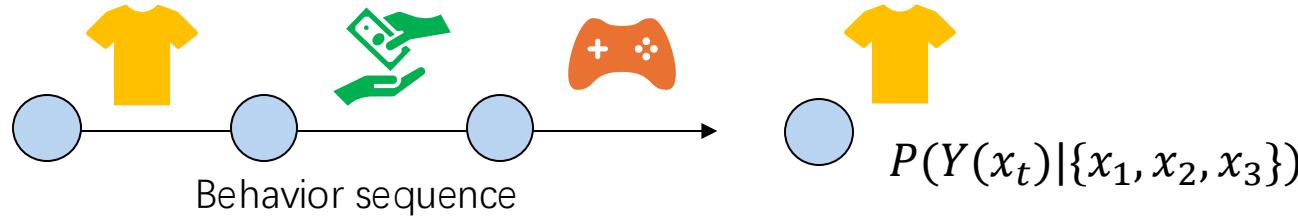
New Perspectives: Dimensional Collapse and Redundancy Reduction.

- **Similar work** treat it as gating/mask: FibiNet, MaskNet, Final, FinalMLP, PEPNet.
- We explain its effectiveness from the following perspectives:
 - Dimensional Collapse: Singular Spectrum
 - Redundancy Reduction: De-Correlation between Fields



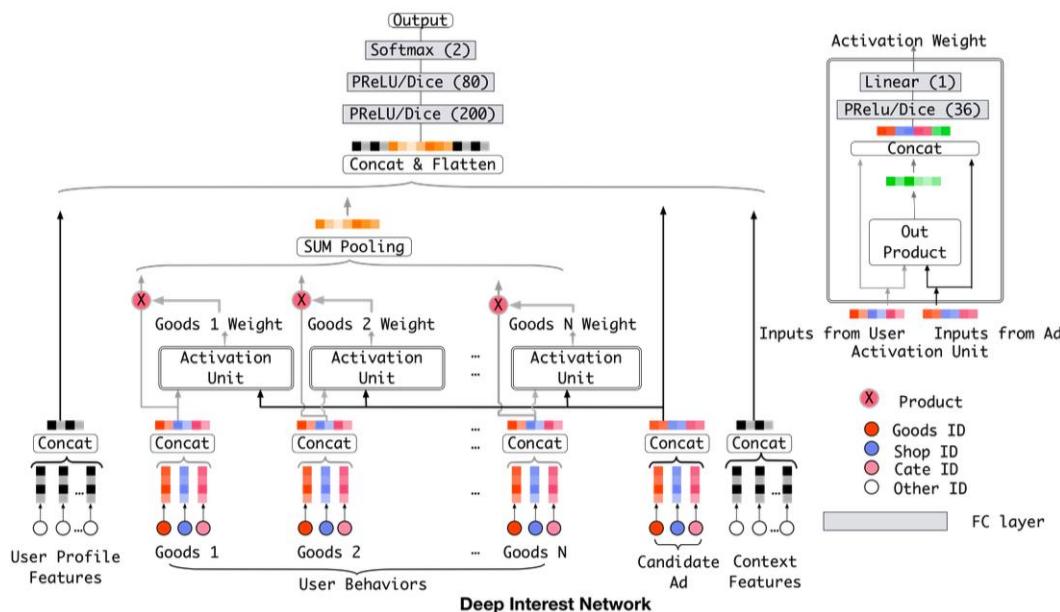
- Part II, Prediction
 - Perspectives
 - Feature Interaction
 - **Sequential Models**
 - Multi-Task and Multi-Domain Learning
 - Large Recommendation Models
 - LLM4Rec

DIN



In prediction, the most important sequential correlation is the behavior-target correlation. DIN: **Target Attention**, captures the correlation between each **behavior** and the **target**.

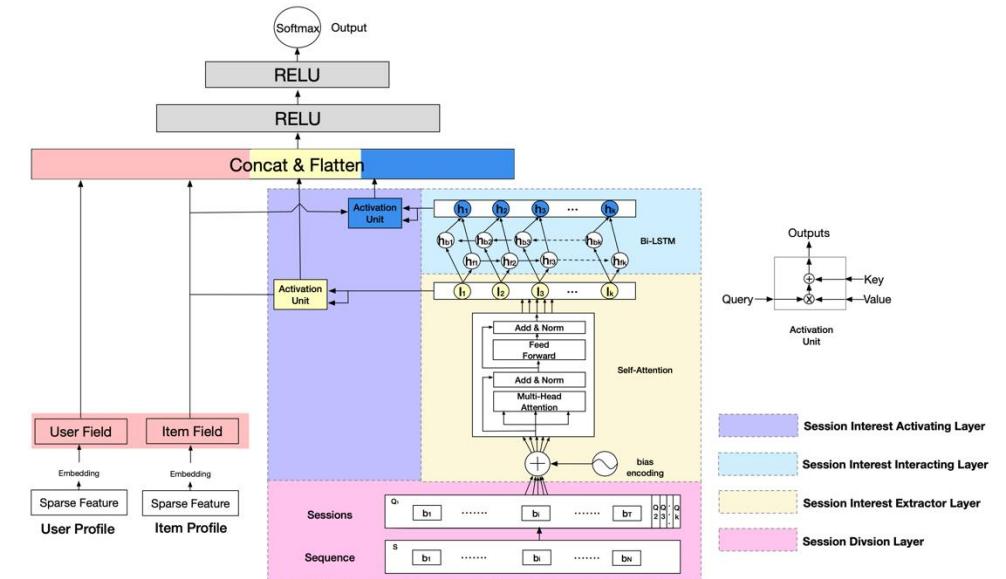
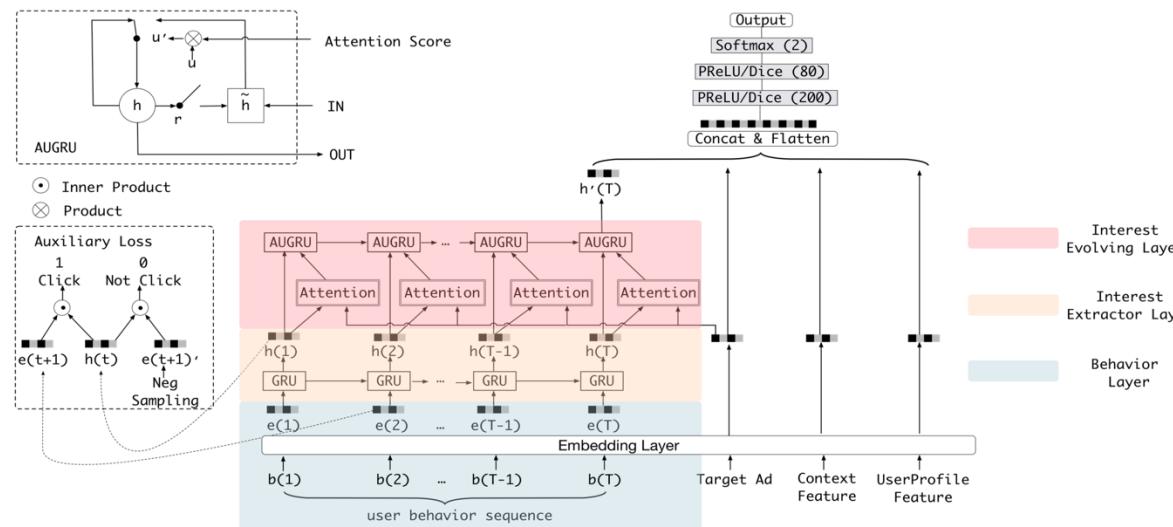
$$\alpha(e_i, v_t) \cdot e_i$$



DIEN, DSIN

DIEN: **GRU** to capture the temporal correlations

DSIN: **LSTM** to capture correlations within and between session.



Deep interest evolution network for click-through rate prediction. AAAI 2019.

Deep Session Interest Network for Click-Through Rate Prediction. 2019.

Tencent

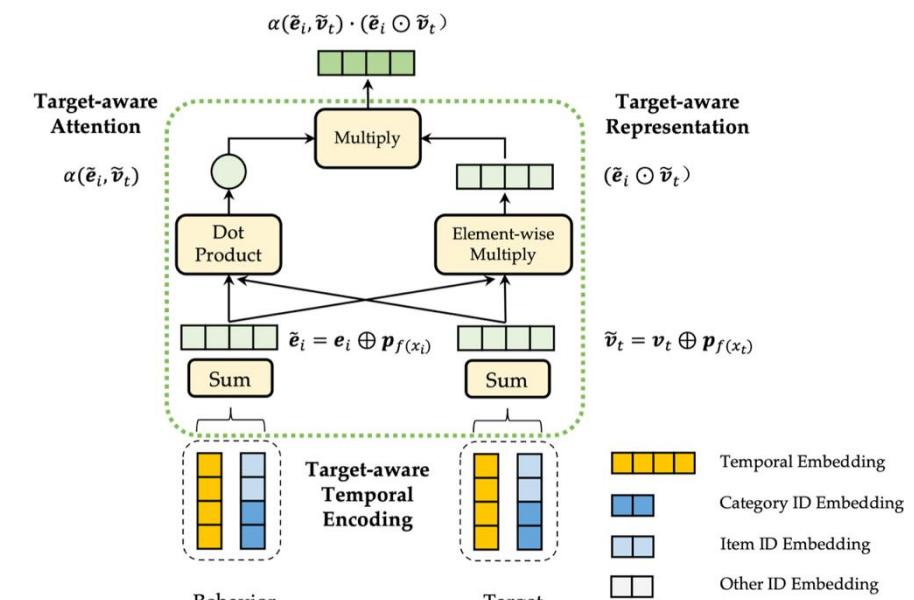
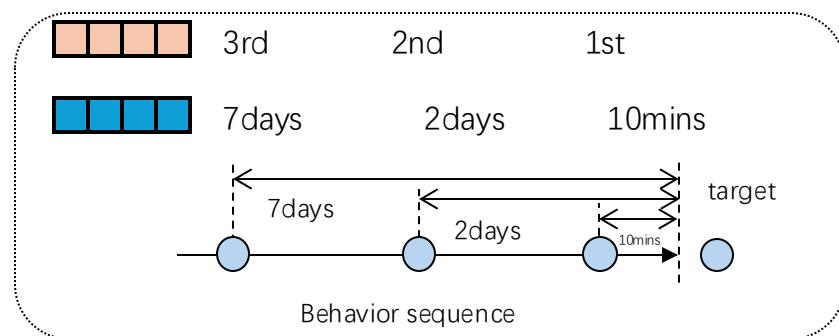
TIN (Temporal Interest Network)

$$\sum_{X_i \in \mathcal{H}} \underbrace{\alpha(\tilde{e}_i, \tilde{v}_t)}_{\text{TA}} \cdot \underbrace{(\tilde{e}_i \odot \tilde{v}_t)}_{\text{TR}}$$

A **Transformer variant** for sequential ranking

Three critical components to capture the temporal-semantic correlation

- **Target-aware Temporal Encoding (TTE)** to capture the temporal correlations: target-relative positions, target-relative time intervals
- **Target Attention (TA)**
- **Target-aware representation (TR)** to enhance the discriminability.

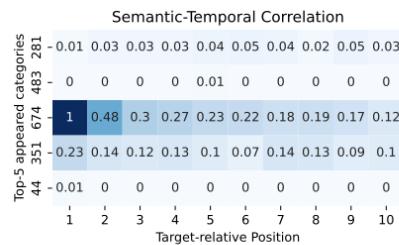


TIN – Temporal-Semantic Correlation Analysis

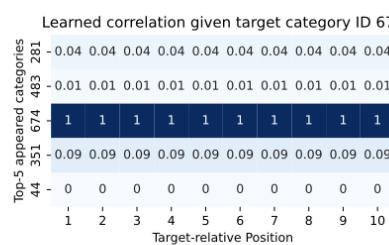
$$\sum_{X_i \in \mathcal{H}} \underbrace{\alpha(\tilde{e}_i, \tilde{v}_t)}_{\text{TA}} \cdot \underbrace{(\tilde{e}_i \odot \tilde{v}_t)}_{\text{TR}}$$

Mutual Information at category and position level

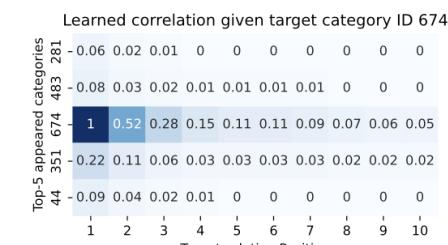
Categories



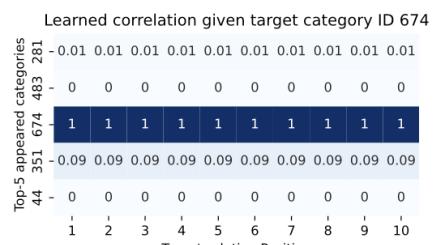
(a) Ground truth STC



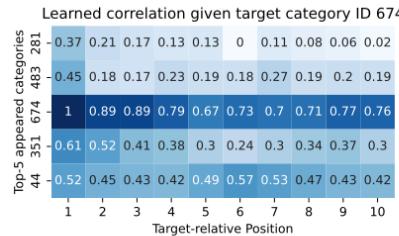
(b) DIN's learned STC (0)



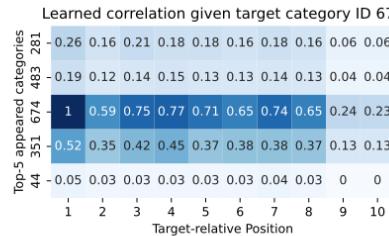
(a) TIN (0.9894)



(b) TIN w/o TTE (0)

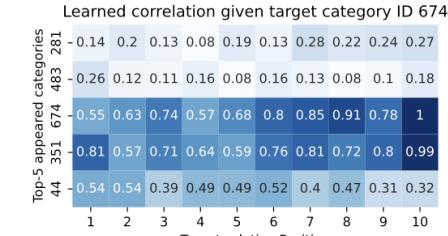


(c) SASRec's learned STC (0.8461)

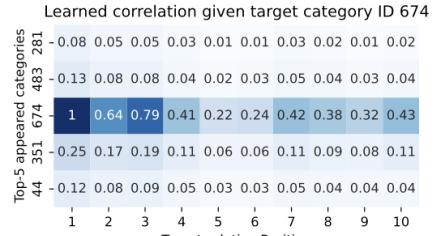


(d) BST's learned STC (0.6245)

temporal position



(c) TIN w/o TA (-0.6845)

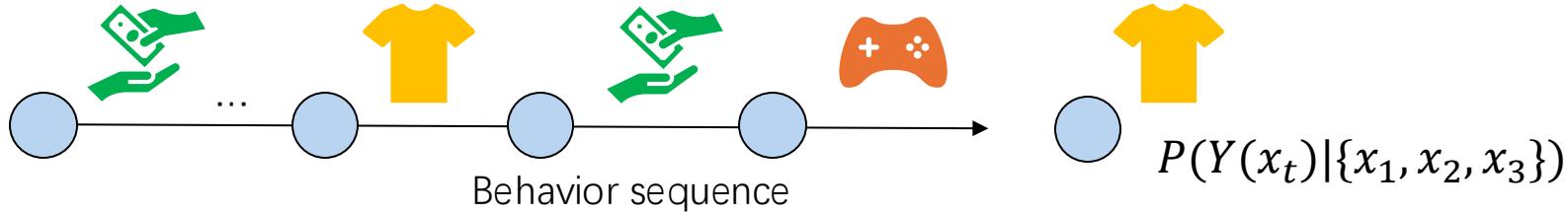


(d) TIN w/o TR (0.8179)

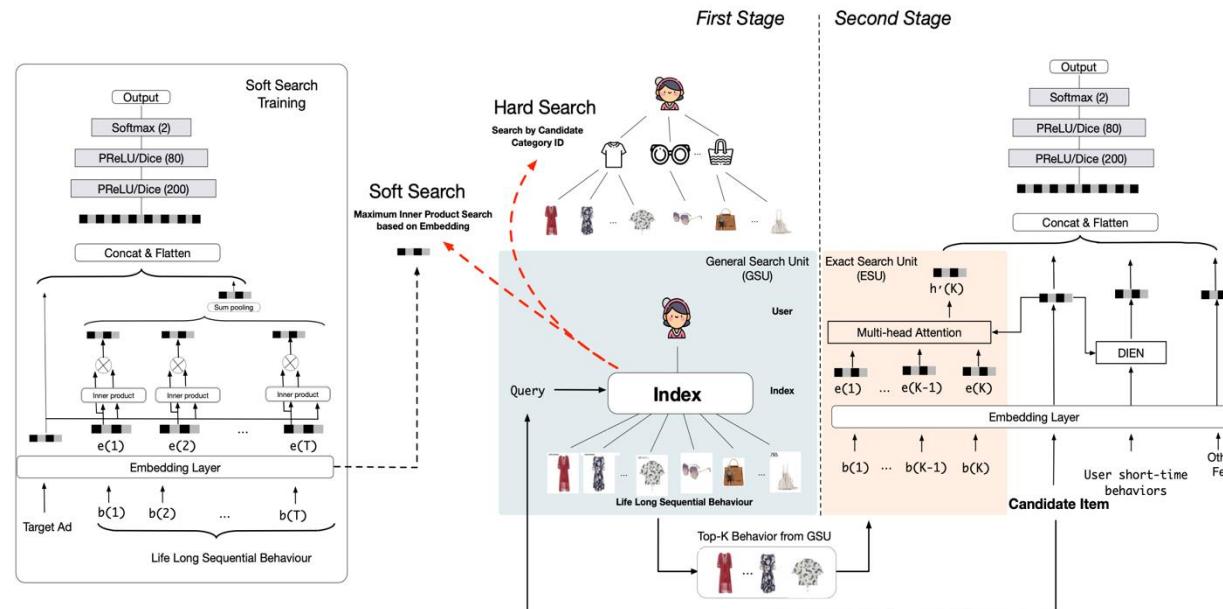
Every component is critical. Ablating any of them makes the model unable to capture the temporal-semantic correlation.

Tencent

SIM



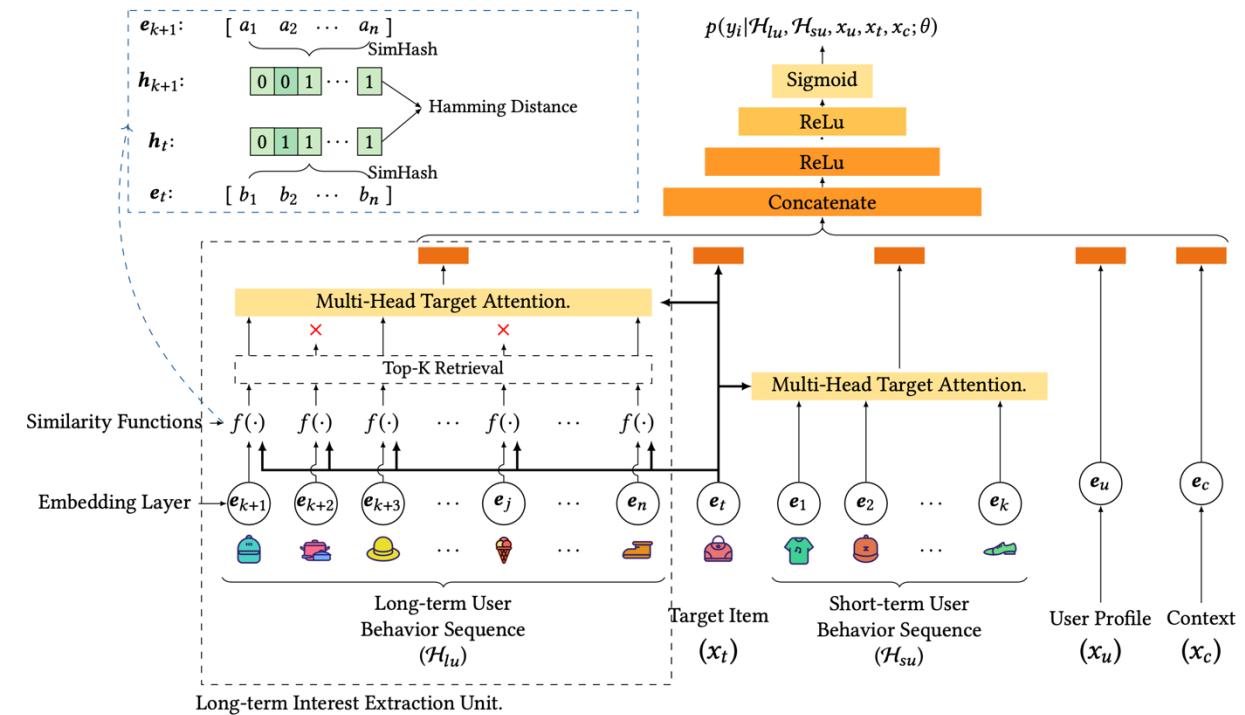
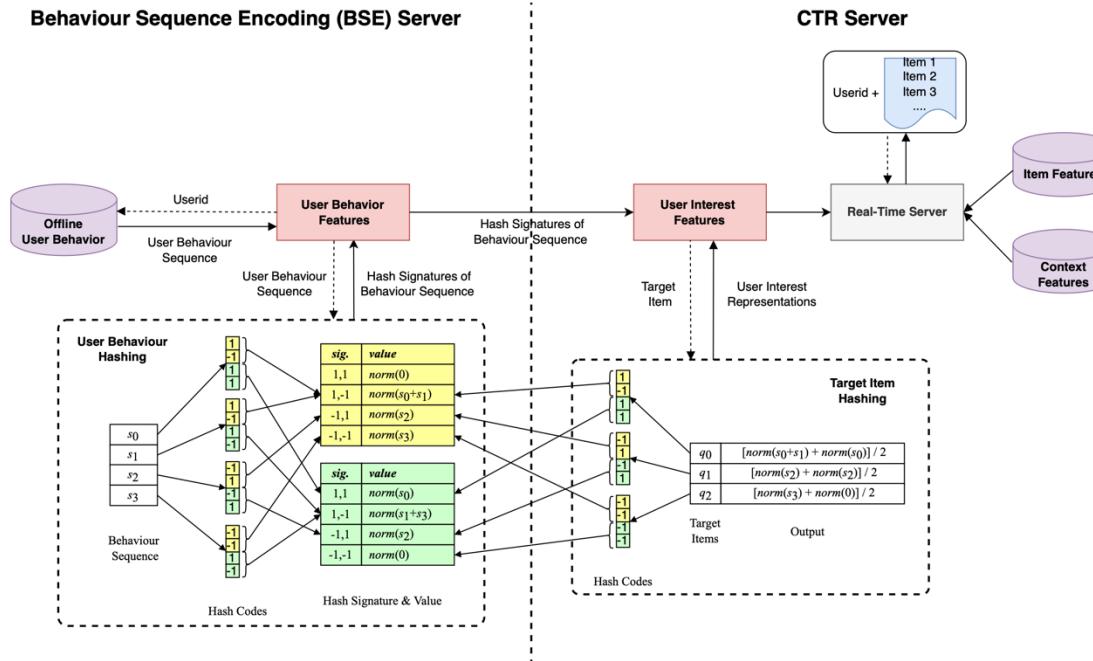
- User's behavior sequence with length of 1K ~ 1M, making **target-attention** infeasible
- Two-stage design for efficient user interest modeling
 - Stage 1: General Search Unit (**GSU**): retrieve the most informative behaviors
 - **Hard Search**: Same category.,.
 - **Soft Search**: Attention score.
 - Stage 2: Exact Search Unit (**ESU**): model user interest on the retrieved behaviors, e.g., DIN.



SDIM, ETA

The **pre-trained** embedding in GSU of SIM is **outdated**

Solution: **end-to-end training**, search by exact match or Hamming distance based on the **LSH**

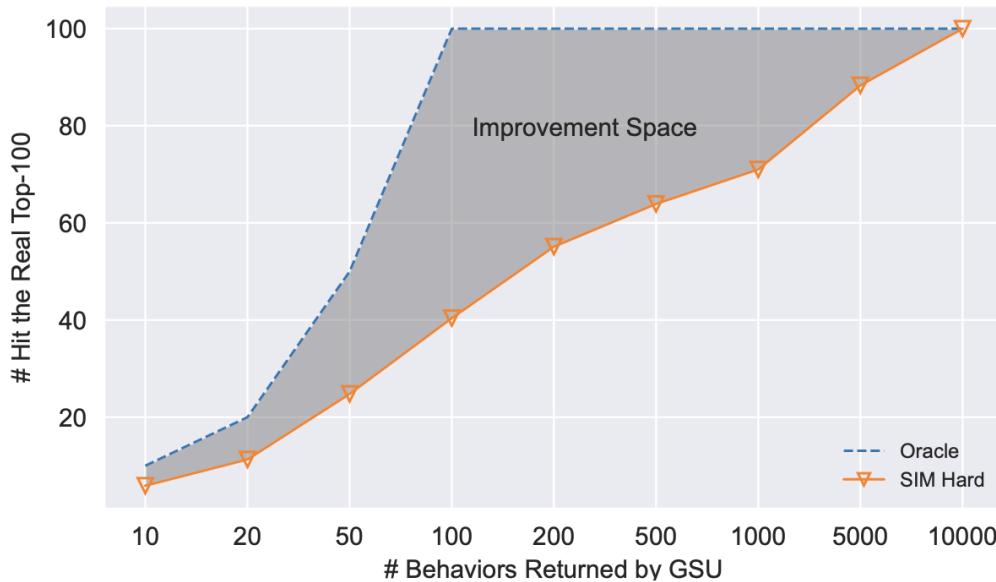


Sampling is all you need on modeling long-term user behaviors for CTR prediction. CIKM 2022.
 ETA: End-to-End User Behavior Retrieval in Click-Through Rate Prediction Model. 2021.

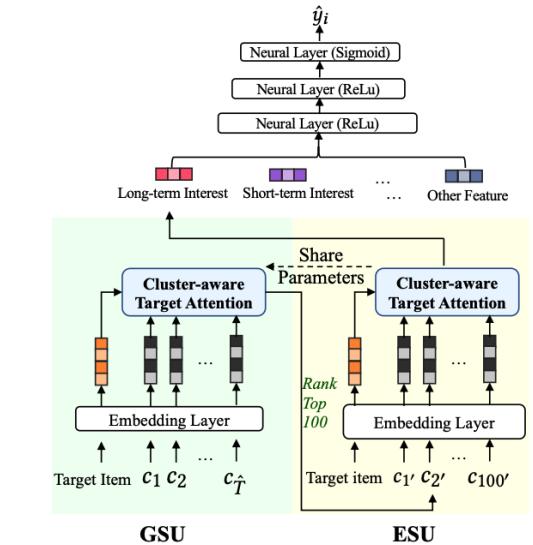
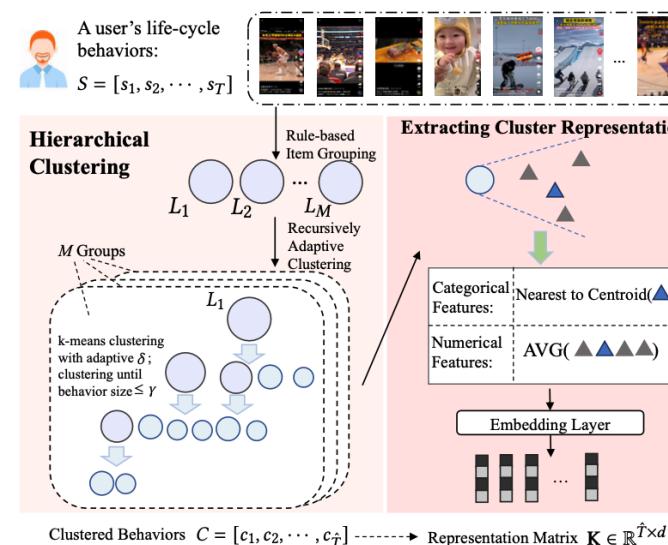
TWIN, TWIN-V2

Challenge: The relevance metric in GSU is **coarse** and **inconsistent** with ESU.

Solution: **identical** target-behavior relevance metrics in GSU as the Target-Attention in ESU.



Divide and Conquer with a hierarchical clustering method for efficient search.



TWIN: Two-stage Interest Network for Lifelong User Behavior Modeling in CTR Prediction at Kuaishou. KDD 2023.

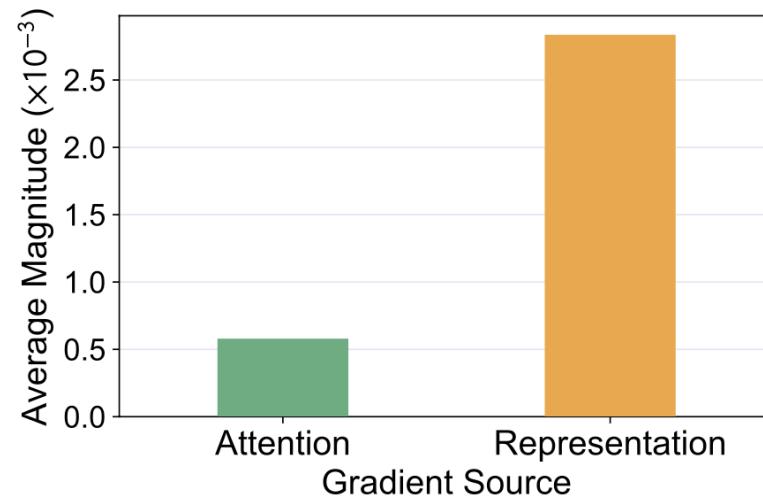
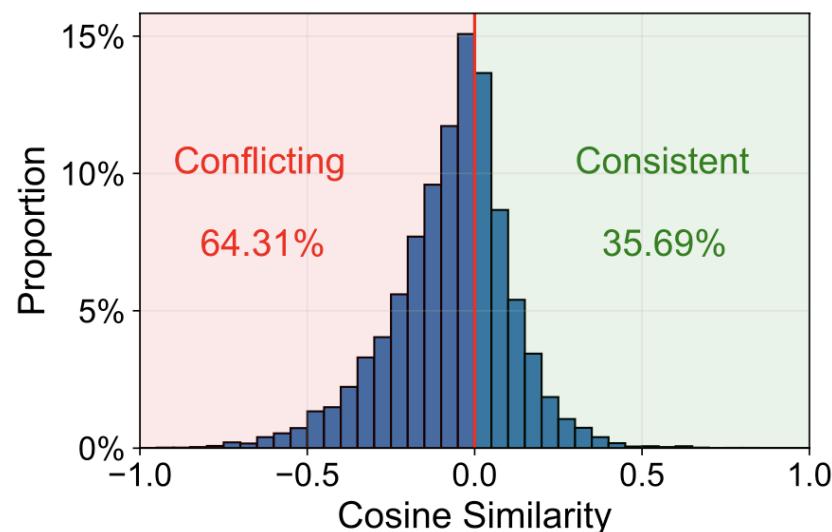
TWIN V2: Scaling Ultra-Long User Behavior Sequence Modeling for Enhanced CTR Prediction at Kuaishou. CIKM 2024.

Tencent

DARE (Decoupled Attention and Representation Embeddings)

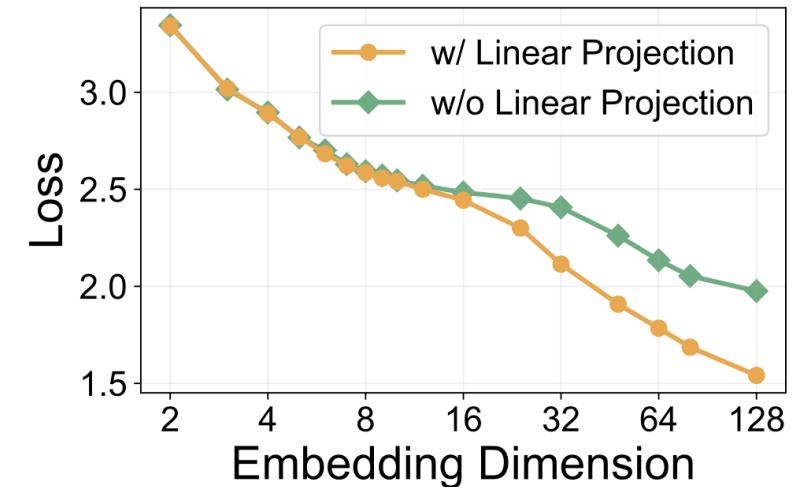
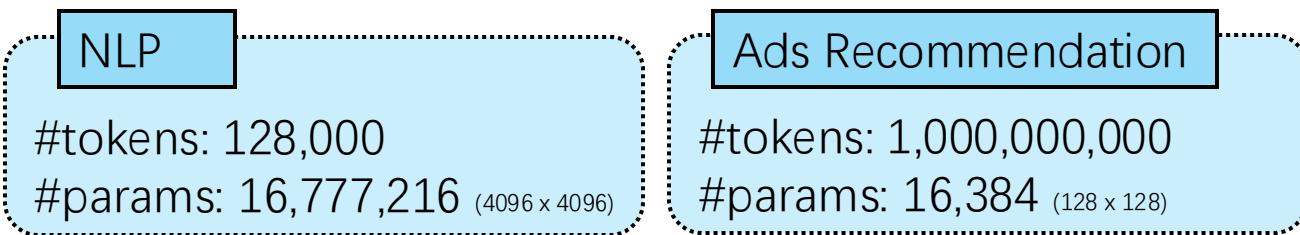
$$\sum_{X_i \in \mathcal{H}} \underbrace{\alpha(\tilde{e}_i, \tilde{v}_t) \cdot (\tilde{e}_i \odot \tilde{v}_t)}_{\text{TA}} \quad \underbrace{\text{TR}}$$

- TWIN and TWIN-V2 use the attention score for retrieval.
- There are **conflicts** between the target **attention** and target **representation**.
- The gradients are **dominated** by the target **representation**.
- Gradient domination deteriorates the attention learning and hence **harm the search quality**.



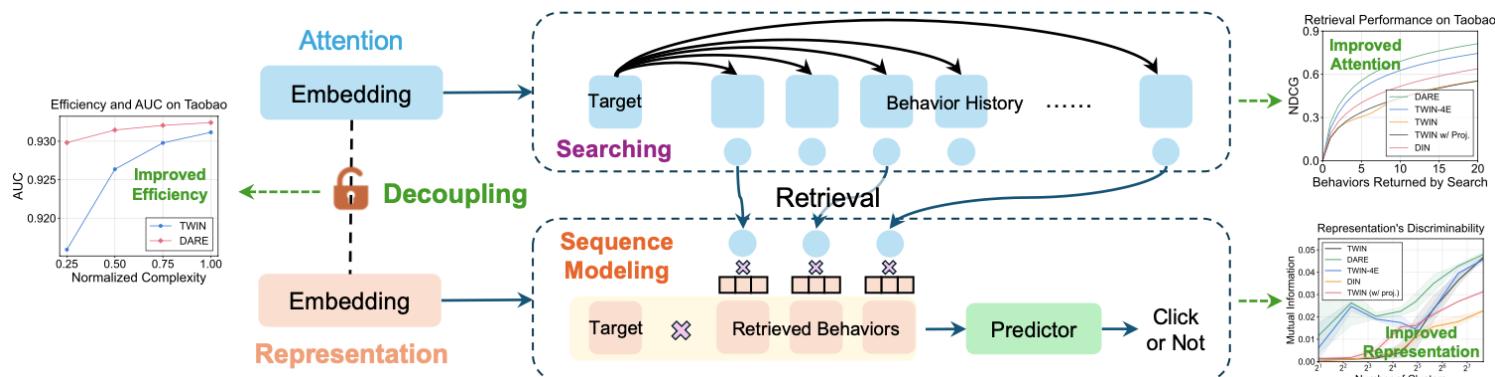
DARE

- Straightforward approach: projection matrix for attention and representation.
- Unlike NLP, the Q, K, U, V **projection matrix** in RecSys is much smaller, making it **unable to decouple** attention and representation.



Projection matrix only works with large embedding dimension.

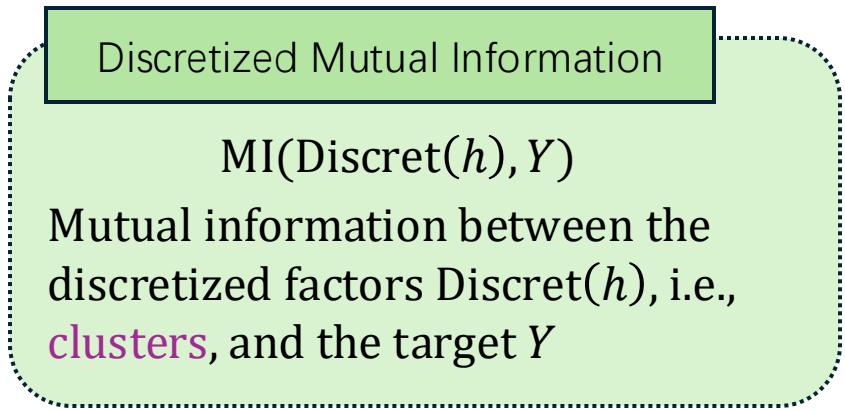
- Solution: **decouple the embeddings** for attention and representation.



DARE – Discriminability Analysis

Embedding decoupling resolves the conflicts between attention and representation, leading to:

- **Better correlation** modeling in attention
- **More discriminative** representation



Category-wise Target-aware Correlation										
Top-10 Appeared Categories										
Target-relative Position										
16	-0.01	0	0	0	0	0	0	0	0	0
11	-0	0	0	0	0	0	0	0	0	0
9	-0.02	0.01	0.01	0	0	0	0	0	0	0
87	-0.01	0.01	0	0	0	0	0	0	0	0
15	-1	0.87	0.67	0.59	0.51	0.47	0.41	0.37	0.34	0.33
1	-0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
19	-0.16	0.12	0.12	0.11	0.1	0.09	0.09	0.09	0.08	0.07
12	-0	0	0.01	0.01	0	0	0	0	0.01	0
20	-0.01	0.01	0	0	0	0	0	0	0	0

(a) GT mutual information

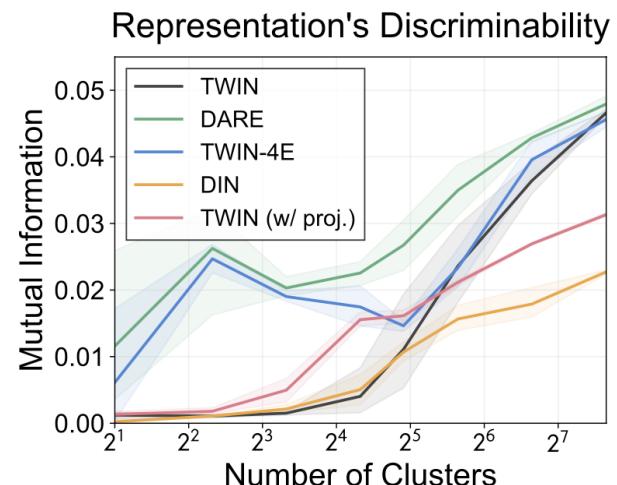
Attention Given Target Category ID 15										
Top-10 Appeared Categories										
Target-relative Position										
16	0.4	0.27	0.2	0.17	0.08	0.1	0.04	0.02	0.01	0.01
11	-0.45	0.3	0.23	0.19	0.1	0.12	0.05	0.03	0.02	0.01
9	-0.38	0.25	0.19	0.16	0.07	0.1	0.04	0.01	0.01	0
87	-0.36	0.24	0.18	0.15	0.07	0.09	0.03	0.01	0.01	0
15	-1	0.69	0.54	0.46	0.26	0.31	0.16	0.11	0.1	0.08
1	-0.53	0.36	0.28	0.23	0.12	0.15	0.07	0.04	0.03	0.02
19	-0.4	0.26	0.2	0.17	0.08	0.1	0.04	0.02	0.01	0
12	-0.72	0.49	0.38	0.33	0.17	0.21	0.1	0.07	0.06	0.04
18	-0.53	0.36	0.28	0.23	0.12	0.15	0.07	0.04	0.03	0.02
20	-0.36	0.24	0.18	0.15	0.07	0.09	0.03	0.01	0.01	0

(b) TWIN learned correlation

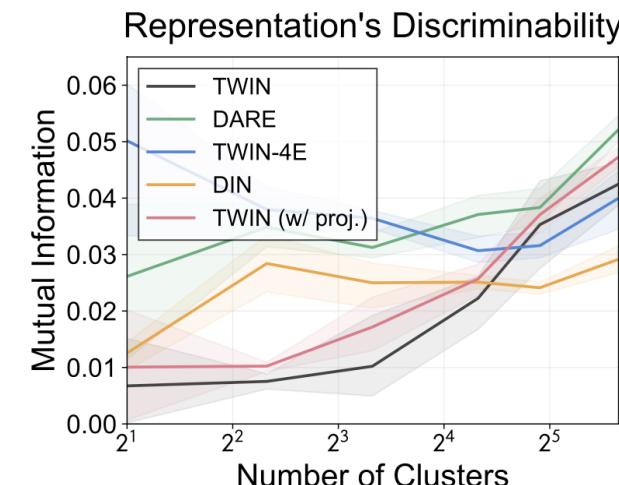
Attention Given Target Category ID 15										
Top-10 Appeared Categories										
Target-relative Position										
16	-0.04	0.03	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0
11	-0.08	0.05	0.03	0.02	0.02	0.02	0.01	0.01	0.01	0.01
9	-0.02	0.01	0.01	0	0	0	0	0	0	0
87	-0.04	0.03	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0
15	-1	0.62	0.42	0.29	0.22	0.23	0.21	0.18	0.16	0.13
1	-0.14	0.09	0.06	0.04	0.03	0.03	0.02	0.02	0.02	0.01
19	-0.04	0.03	0.02	0.01	0.01	0.01	0.01	0.01	0	0
12	-0.12	0.07	0.05	0.03	0.02	0.03	0.02	0.02	0.02	0.01
20	-0.01	0.01	0	0	0	0	0	0	0	0

(c) DARE learned correlation

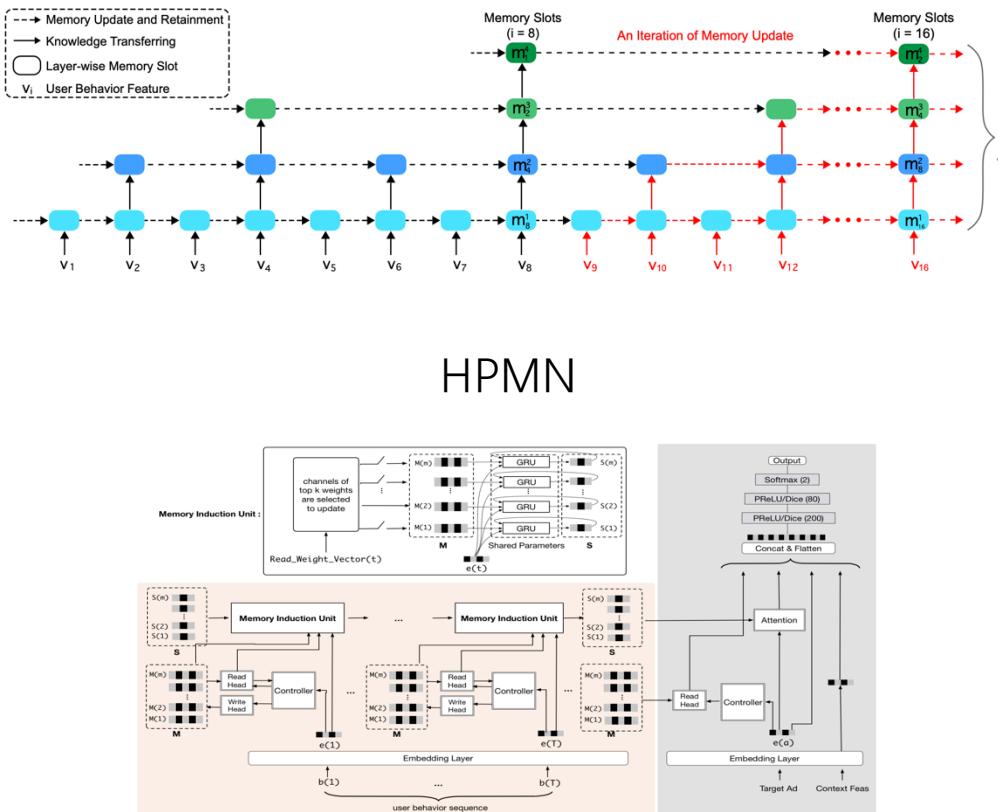
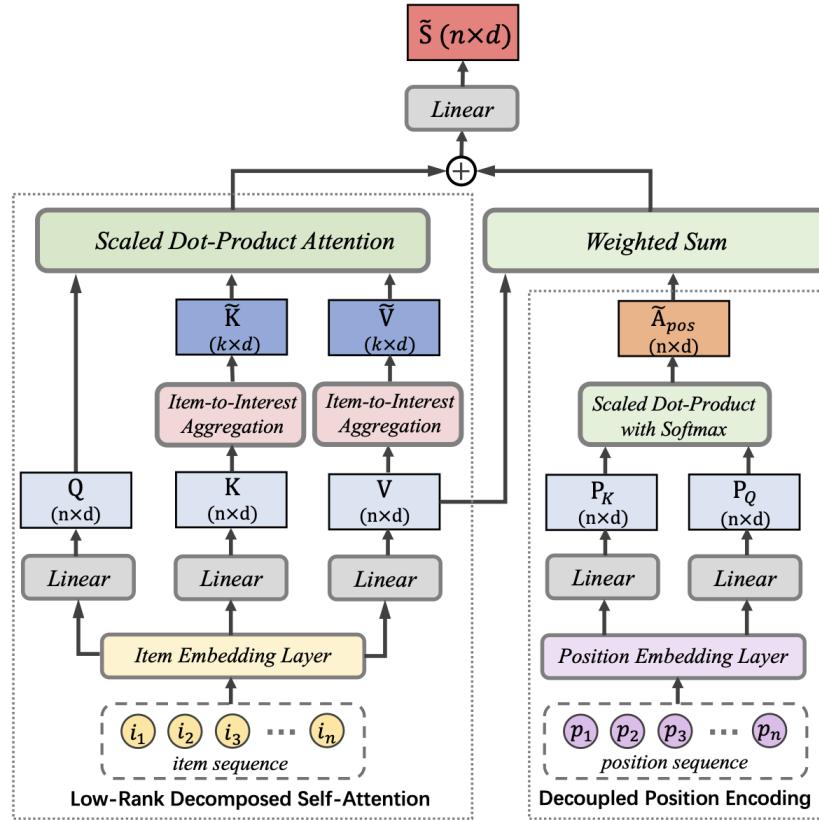
Temporal-Semantic Correlation



Representation Discriminability



Compression - Memory Networks, Low-rank Decomposition



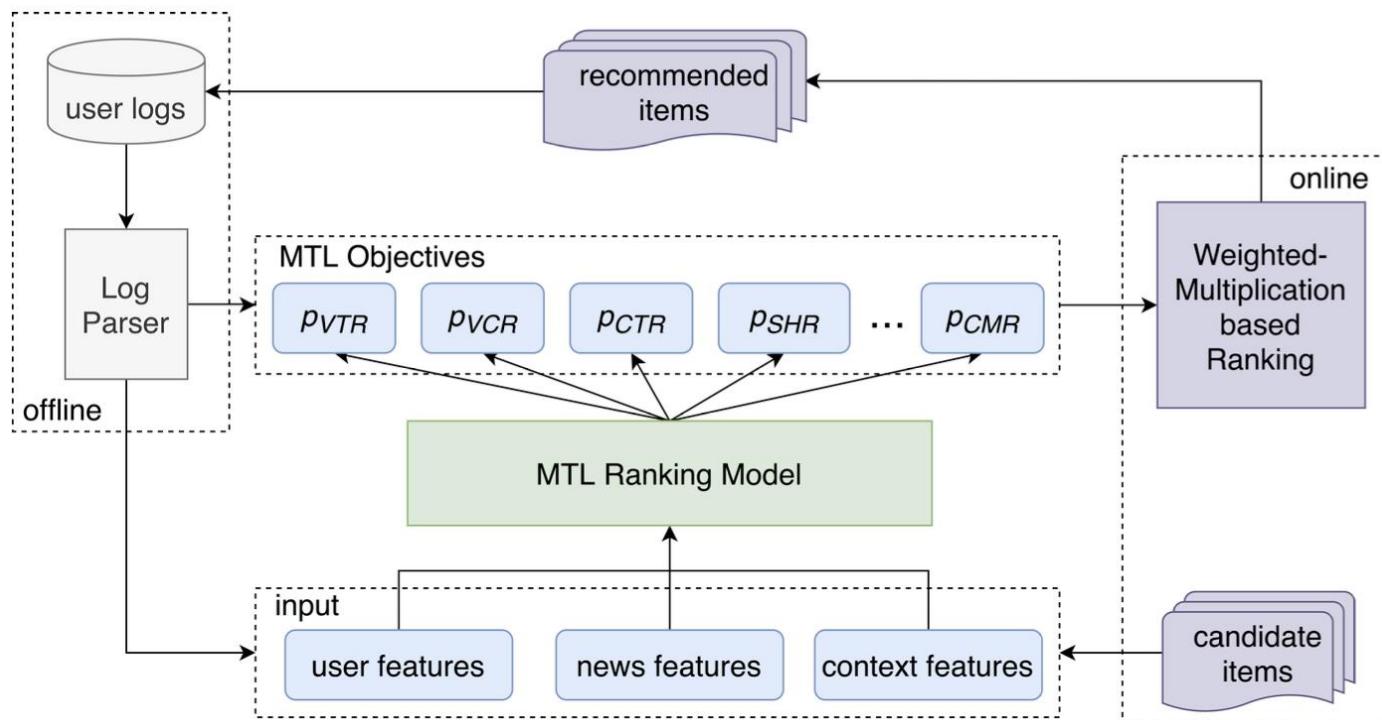
LightSANS

Lighter and Better: Low-Rank Decomposed Self-Attention Networks for Next-Item Recommendation. 2021.
Lifelong Sequential Modeling with Personalized Memorization for User Response Prediction. SIGIR 2019.
Practice on Long Sequential User Behavior Modeling for Click-Through Rate Prediction. KDD 2019.

- Part II, Prediction
 - Perspectives
 - Feature Interaction
 - Sequential Models
 - **Multi-Task and Multi-Domain Learning**
 - Large Recommendation Models
 - LLM4Rec

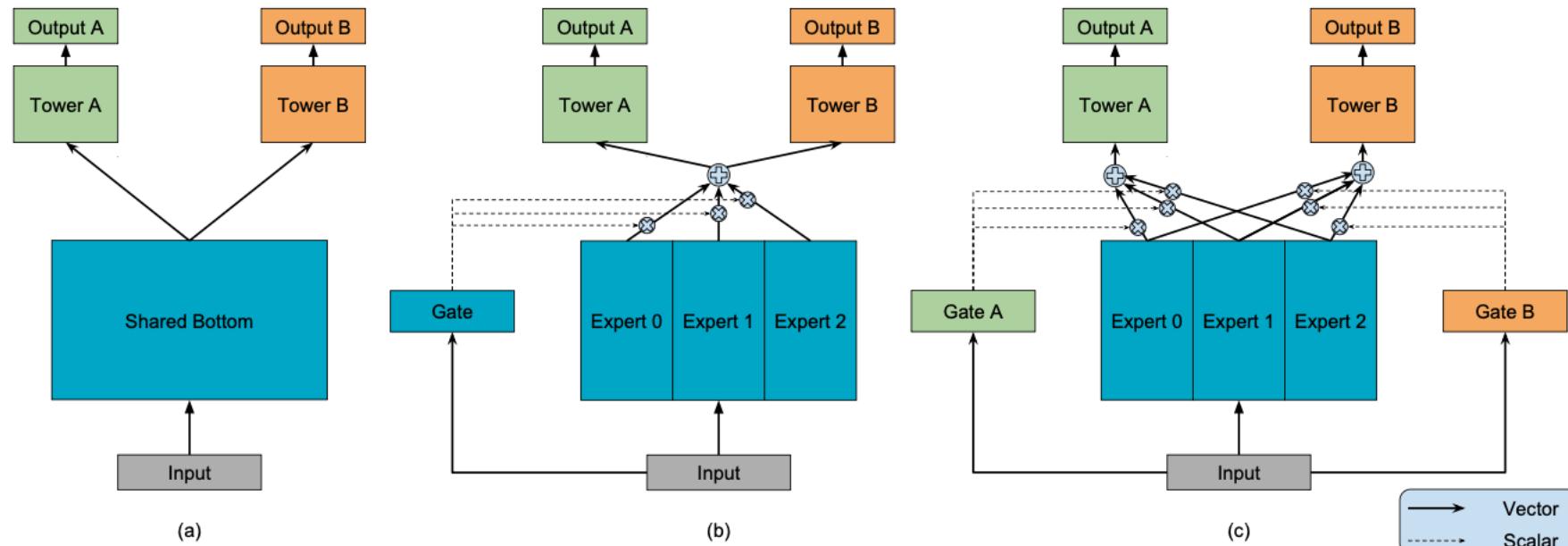
Multi-task/domain Learning

- Multi-task Learning aim to utilize the shared/transferable knowledge from other tasks to enhance every task.
- Challenges: **negative transfer, seesaw phenomenon.**



MMoE (Multi-Gate MoE)

- An art of balance between **shared** and **specific** components

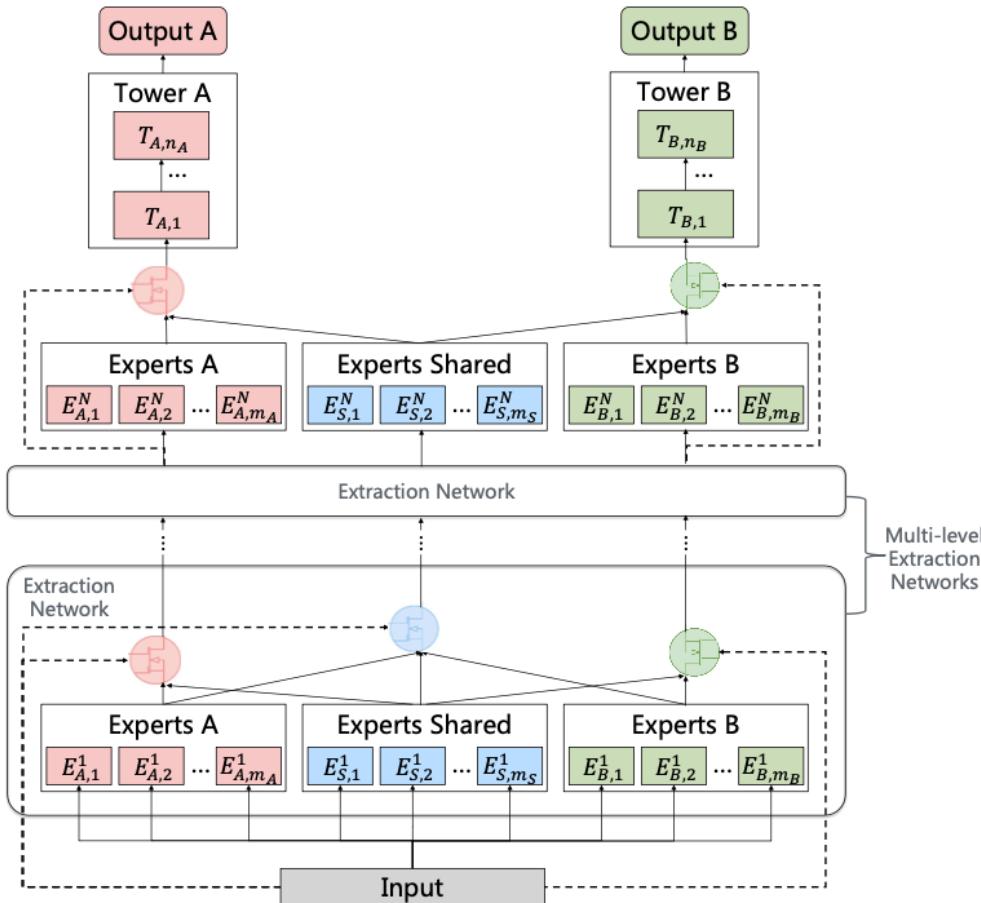


Shared-Bottom
One **shared** experts for all tasks.

MoE
one **shared** gating
multiple **shared** experts,

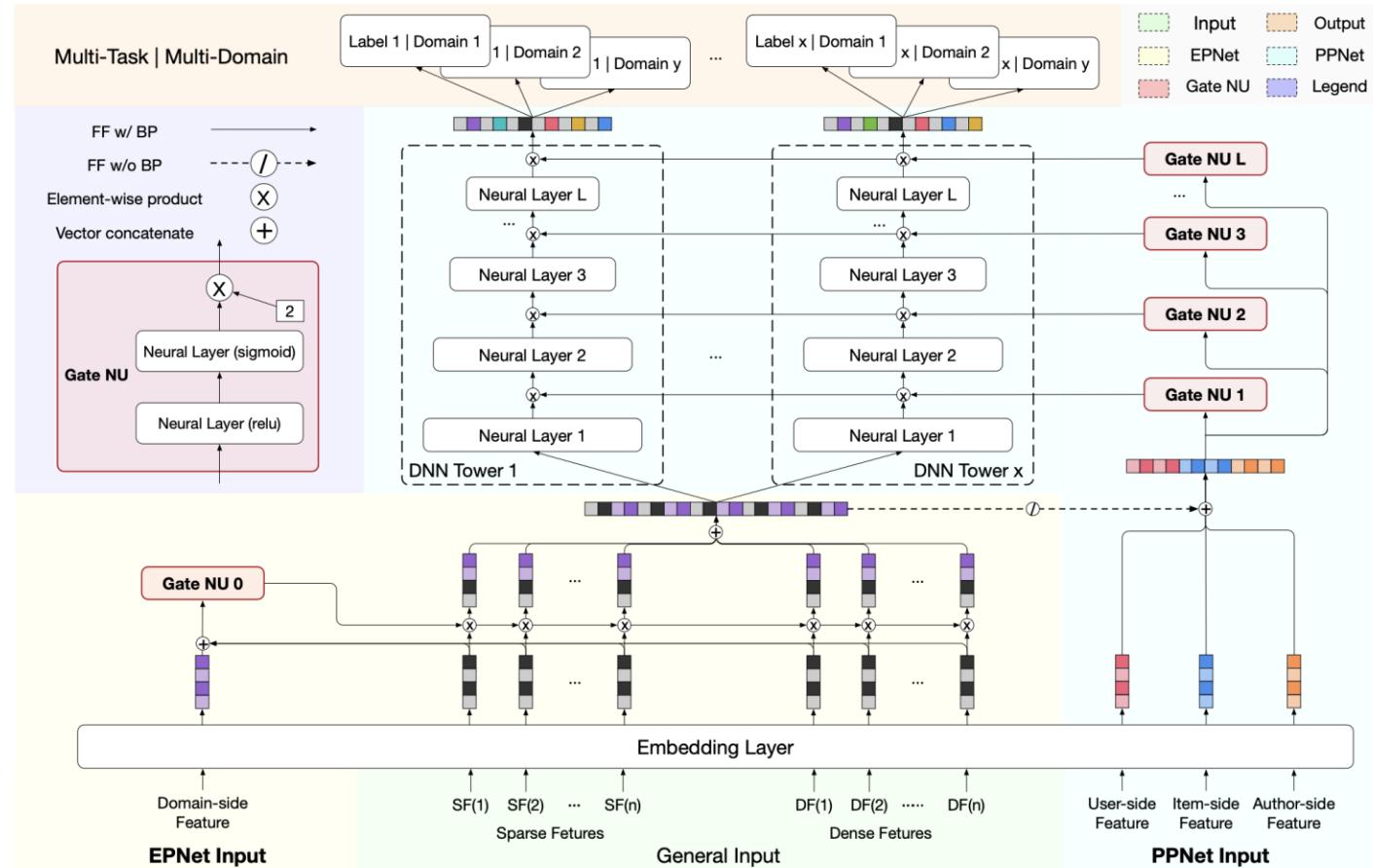
MMoE
task-specific gating
multiple **shared** experts,

PLE (Progressive Layered Extraction)



task-specific gating
shared and task-specific experts

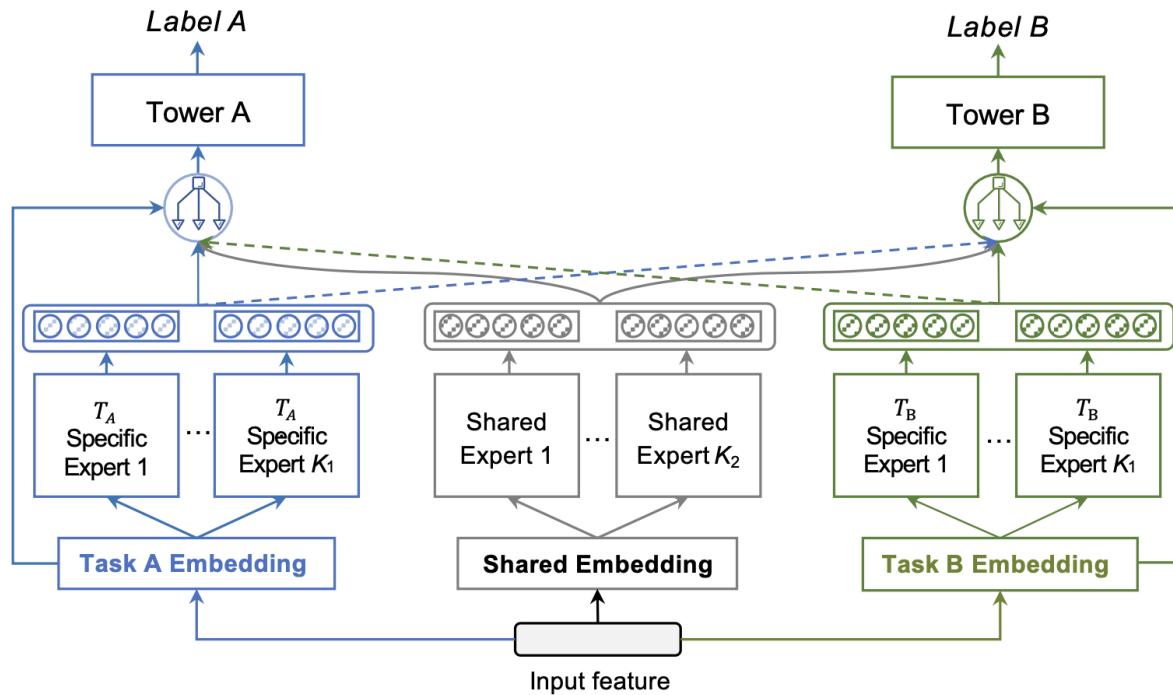
PEPNet (Parameter and Embedding Personalized Networks)



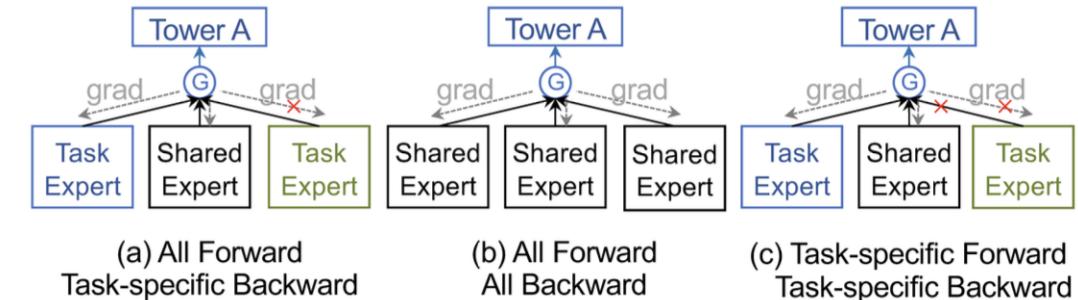
task-specific gating
task-specific embedding gating (experts)

STEM (Shared and Task-specific EMbeddings)

Learn shared and **task-specific embeddings**.



task-specific gating
shared and **task-specific** experts
shared and **task-specific** embeddings

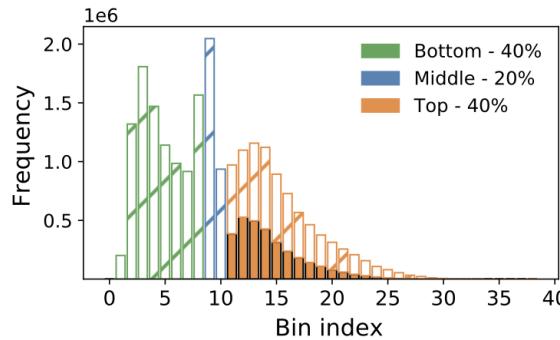


All Forward **Task-specific Backward** Gating

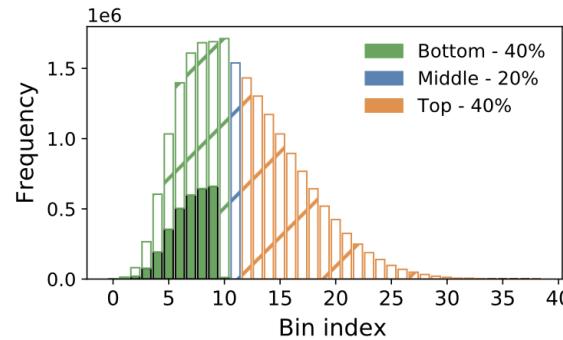
STEM – contradictory preference analysis

Pick up user-item pairs that have contradictory distances in task like and finish.

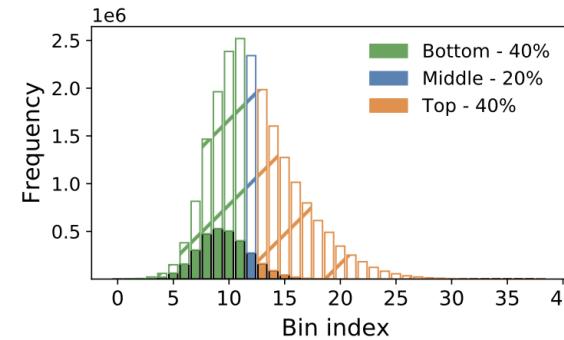
- E.g., user-item pairs that have **large** distance in the Like Single Model embedding, while have **small** distance in Finish Single Model Embedding



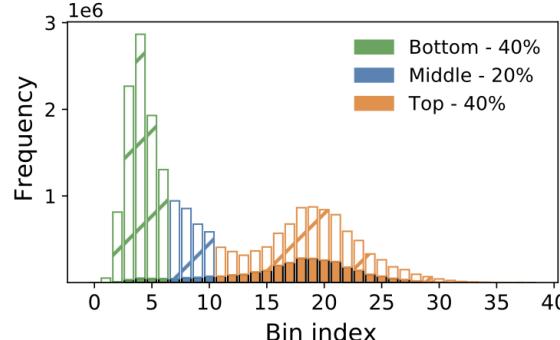
(a) Single Task (Like Embedding)



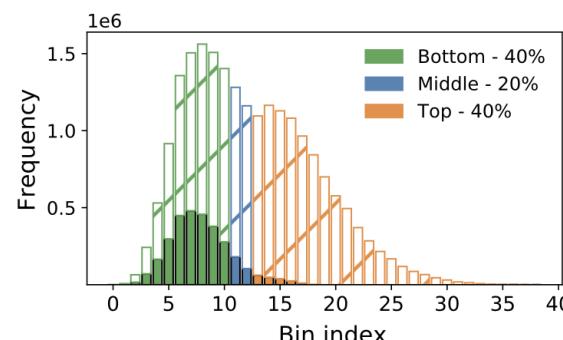
(b) Single Task (Finish Embedding)



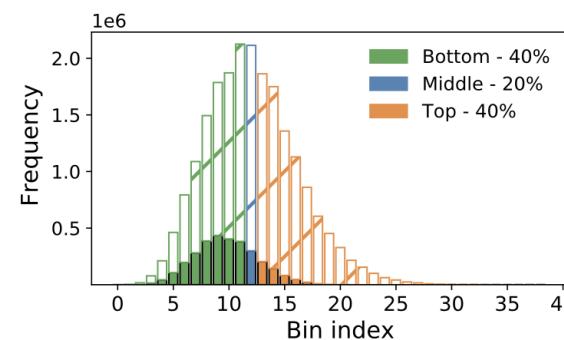
(c) PLE (Shared-Embedding)



(d) STEM-Net (Like Embedding)

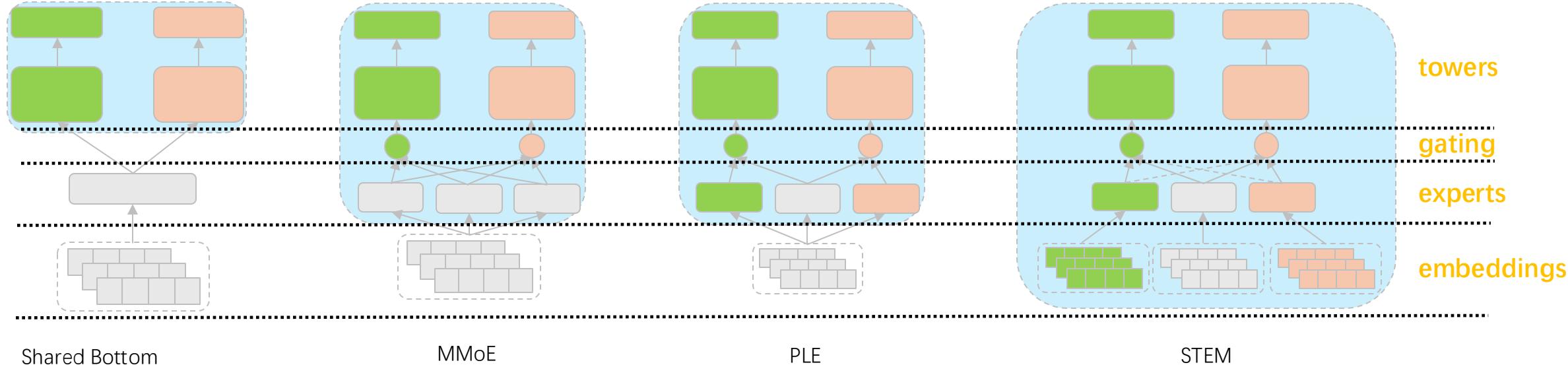


(e) STEM-Net (Finish Embedding)



(f) STEM-Net (Shared Embedding)

MTL Models Evolution



Shared Bottom

MMoE

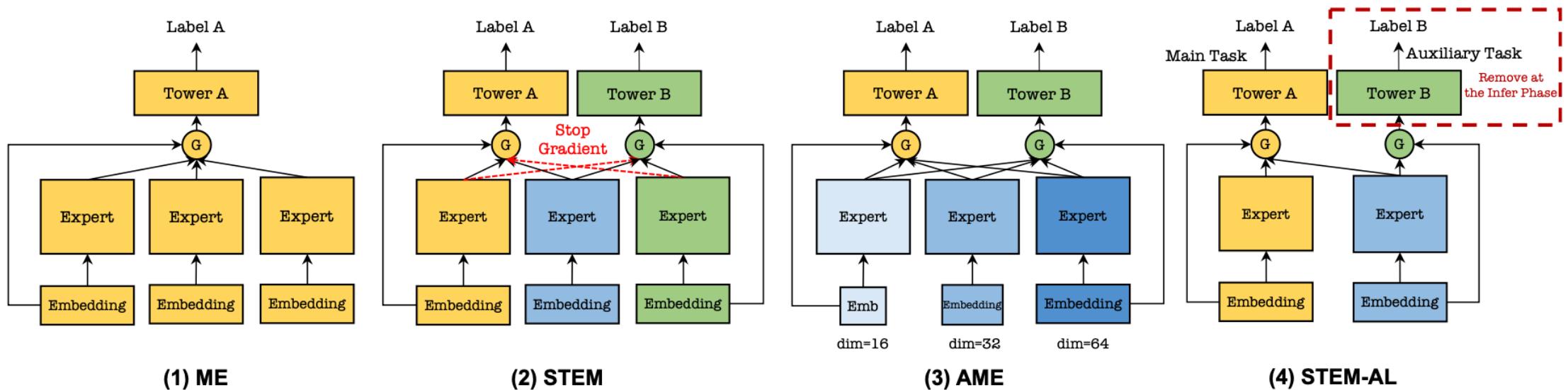
PLE

STEM

A trend to decouple more bottom components, i.e., from gating (MMoE) to experts (PLE, PEPNet) to embeddings (STEM)

AME (Asymmetric Multi-Embedding)

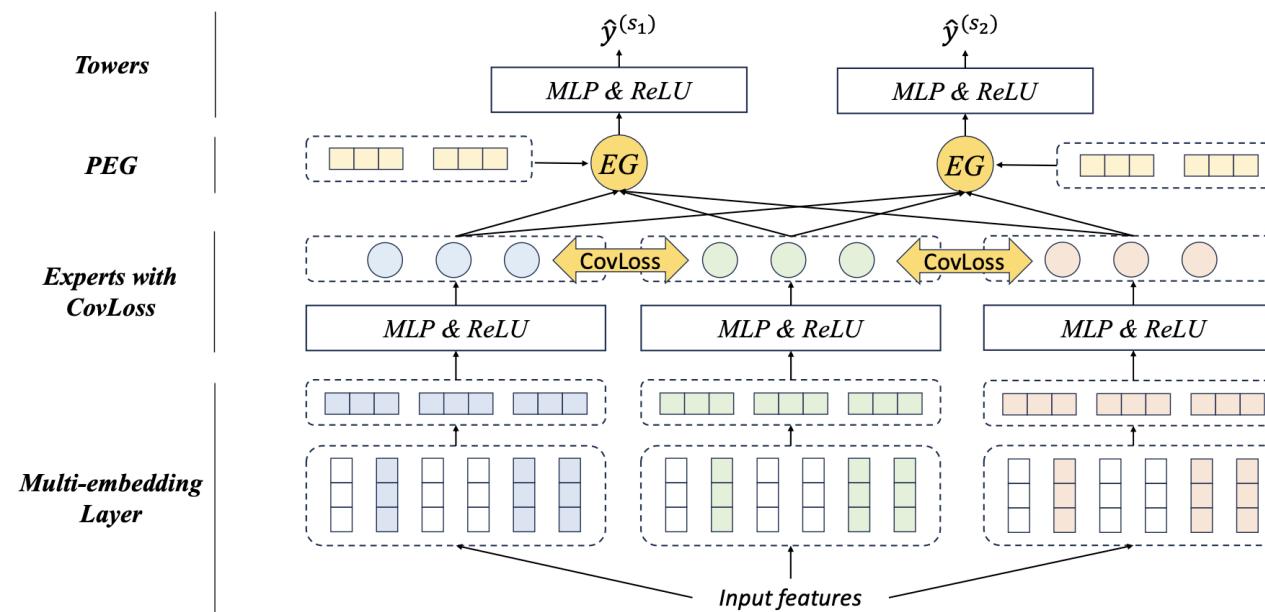
- STEM requires to learn task-specific embedding, which is **infeasible** with many tasks, e.g., more than **hundreds** in Tencent. We may group tasks, but how?
- AME: Learn multiple **shared** yet **asymmetric** embeddings and tasks, with different embedding dimensions **to break the homogeneity**.
- For auxiliary learning, try STEM-AL.



Crocodile (Cross-experts Covariance Loss for Disentangled Learning)

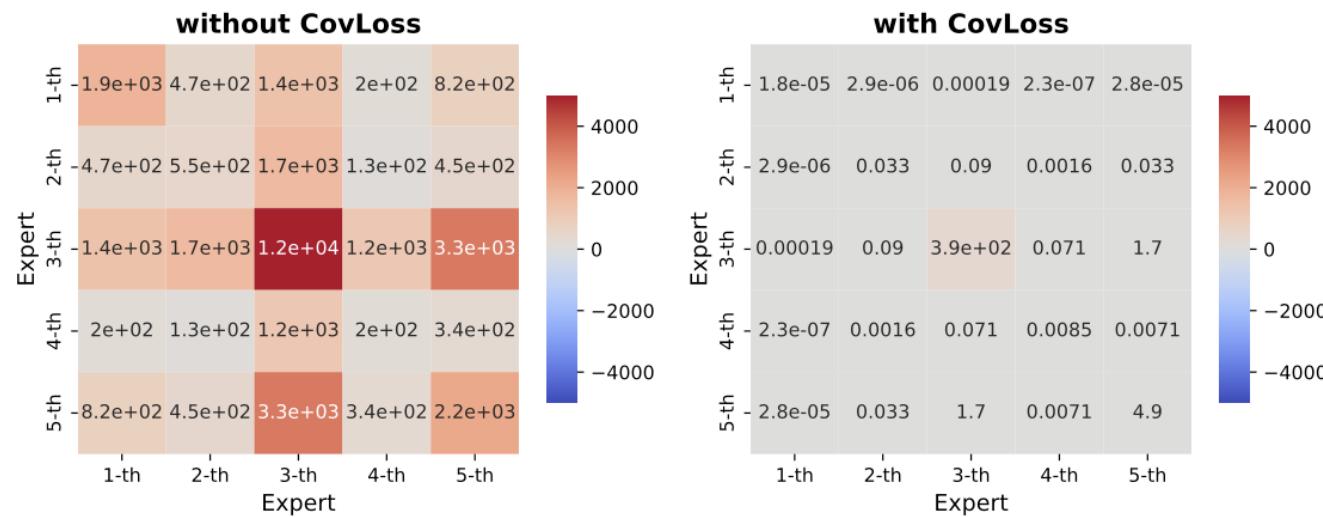
- Different embedding dimensions (AME) or heterogeneous expert modules (DHEN) **constrains** the architecture design.
- A more general approach: **Cross Expert Covariance** loss, to de-correlate the experts directly.

$$\mathcal{L}_{Cov} = \frac{1}{d^2} \sum_{p,q \in M \times M, p \geq q} \|[\mathbf{O}^{(p)} - \bar{\mathbf{O}}^{(p)}]^T [\mathbf{O}^{(q)} - \bar{\mathbf{O}}^{(q)}]\|_1$$

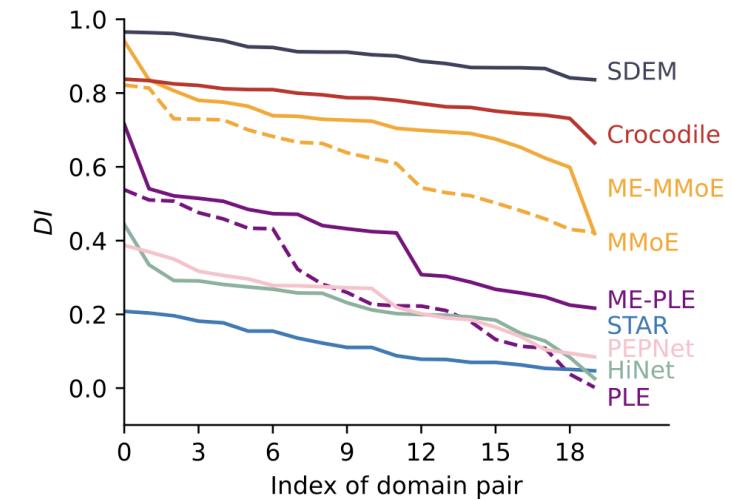


Crocodile - Disentanglement

The CovLoss significantly **reduces** the inter-expert correlations.



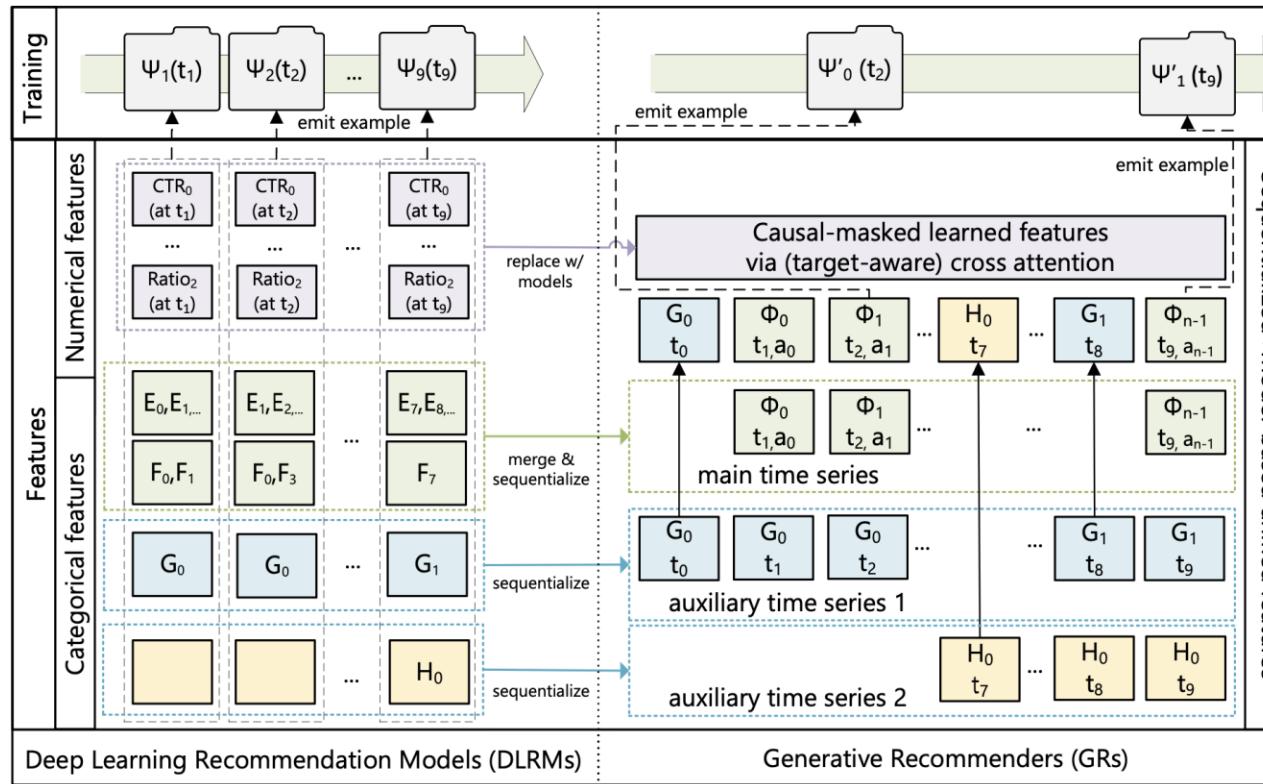
Better Diversity-Index based on the contradictory preference analysis.



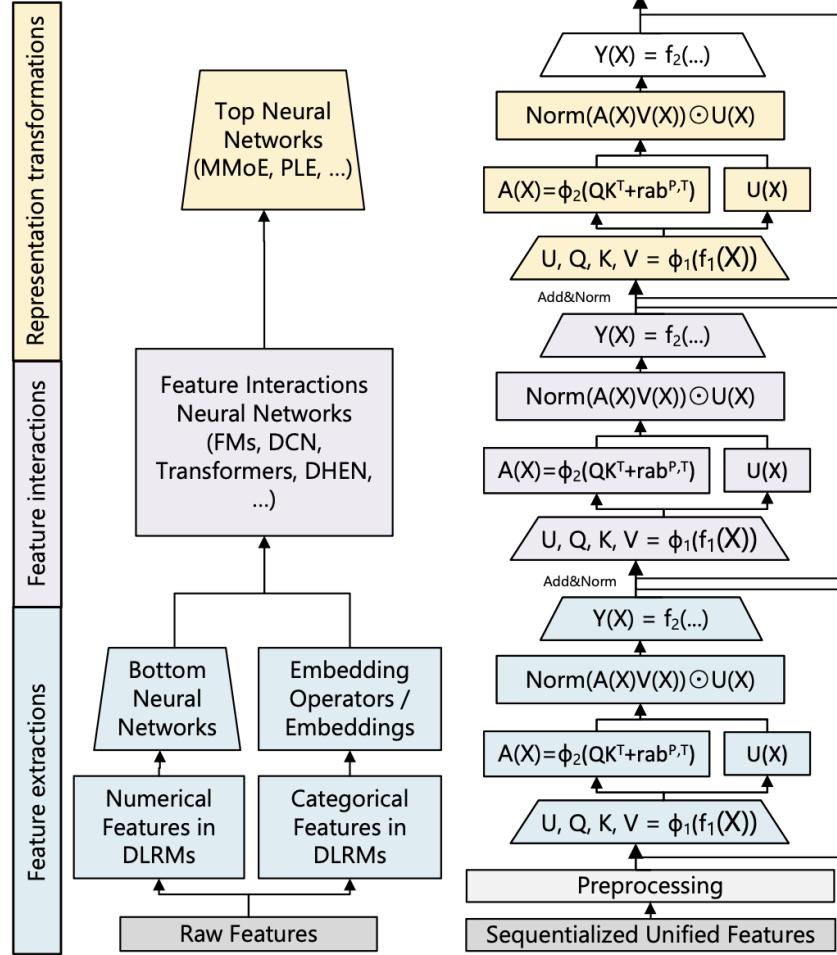
- Part II, Prediction
 - Perspectives
 - Feature Interaction
 - Sequential Models
 - Multi-Task and Multi-Domain Learning
 - **Large Recommendation Models**
 - LLM4Rec

HSTU

- **Scaling laws** in LLM.
- Formulate recommendation as sequential transduction tasks within a **generative** modeling framework.
- Build a universal sequence to unify categorical and numerical features.



HSTU



$$U(X), V(X), Q(X), K(X) = \text{Split}(\phi_1(f_1(X))) \quad (1)$$

$$A(X)V(X) = \phi_2 \left(Q(X)K(X)^T + \text{rab}^{p,t} \right) V(X) \quad (2)$$

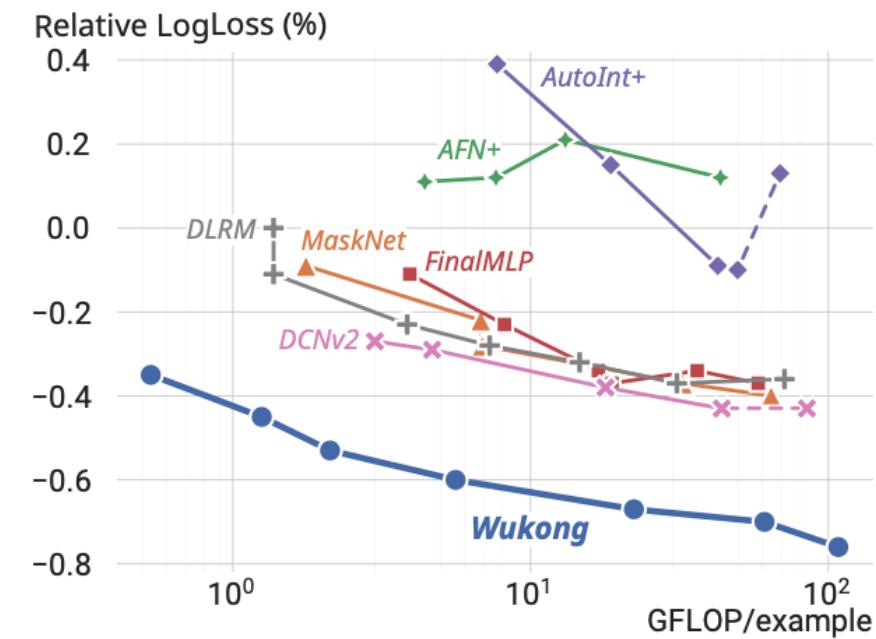
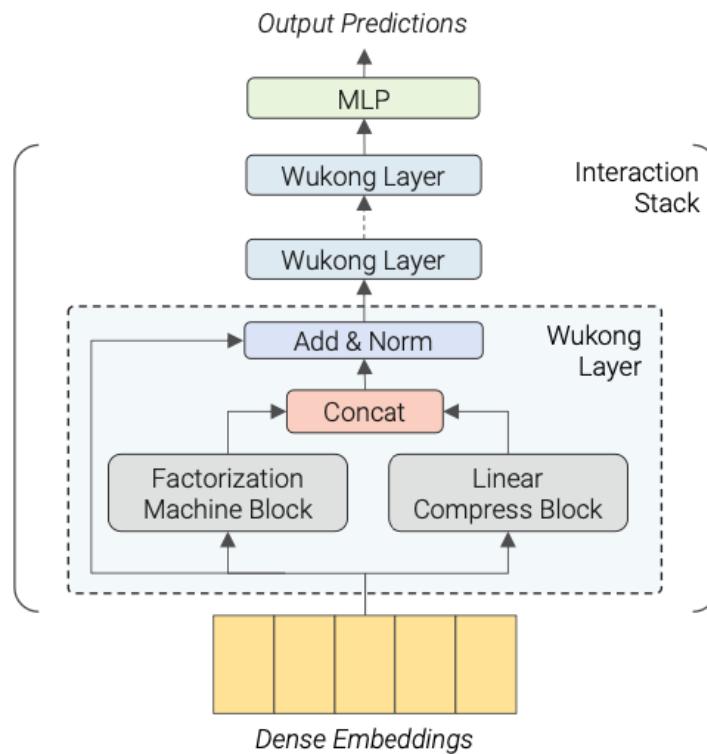
$$Y(X) = f_2 \left(\text{Norm} (A(X)V(X)) \odot U(X) \right) \quad (3)$$

Can be interpreted as a variant of **SwiGLU**, with a bilinear term $V(X) \odot U(X)$, which is similar with the **target-aware representation** in TIN.

$$\sum_{X_i \in \mathcal{H}} \underbrace{\alpha(\tilde{e}_i, \tilde{v}_t)}_{\text{TA}} \cdot \underbrace{(\tilde{e}_i \odot \tilde{v}_t)}_{\text{TR}}$$

Wukong

Each layer in the Interaction Stack consists of a **Factorization Machine Block** and a **Linear Compression Block**.



- The linear compression block helps **mitigate** the **dimensional collapse**.
- **Scale up** (HSTU, Wukong) v.s. **scale-out**

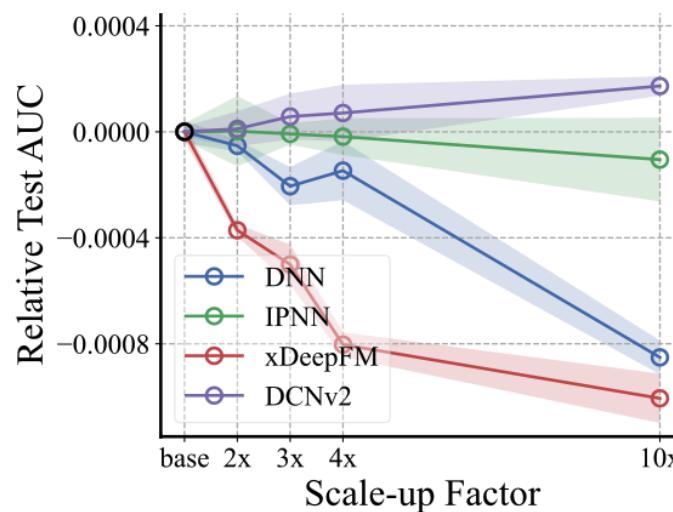
Dimensional Collapse

- Embeddings **dominates** the parameters in RecSys. A straightforward way to scale is to **enlarge embedding dimensions K** .
- However, the performance **deteriorates** when we increase K .
- Through singular spectrum analysis, we found severe **dimensional collapse**.

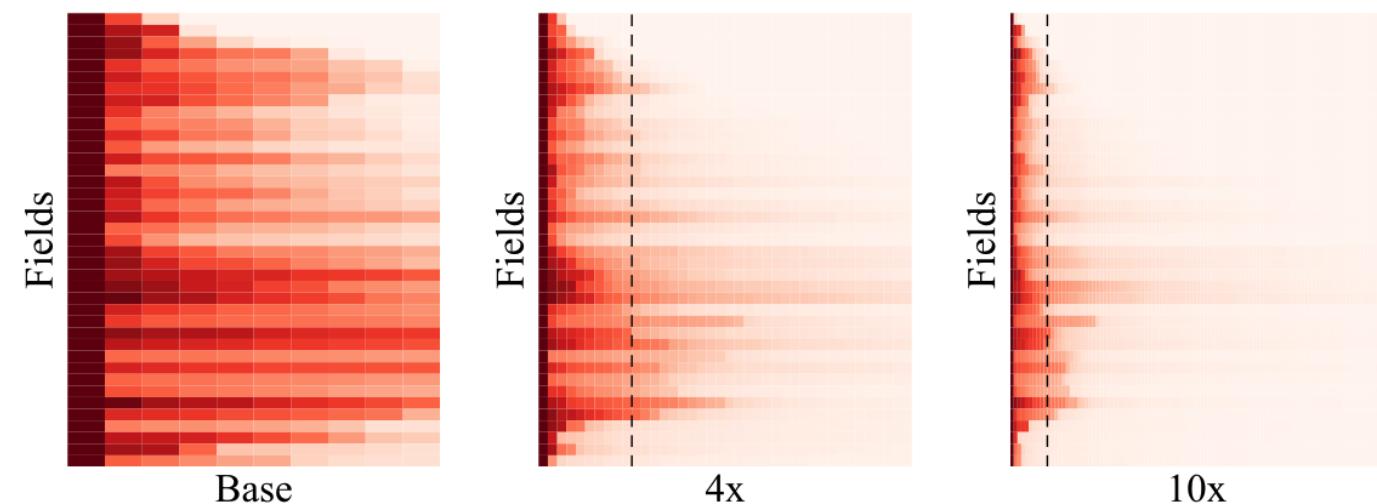
Information Abundance

$$IA = \frac{|\Sigma|_1}{|\Sigma|_\infty}$$

Sum of all singular values divided by the maximum one

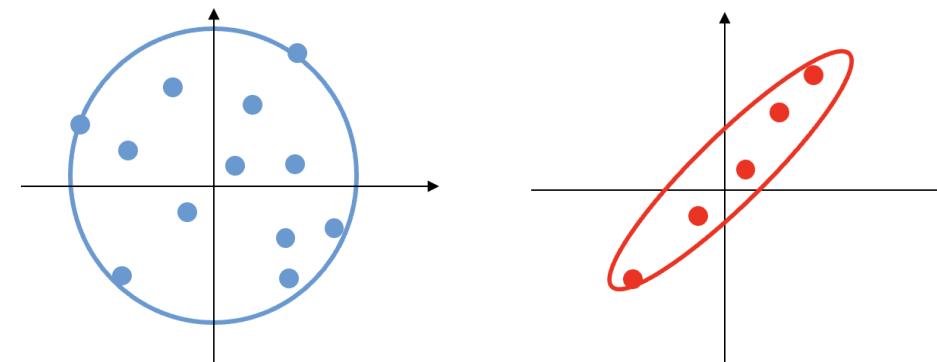


(a) Performance when scaling up recommendation models

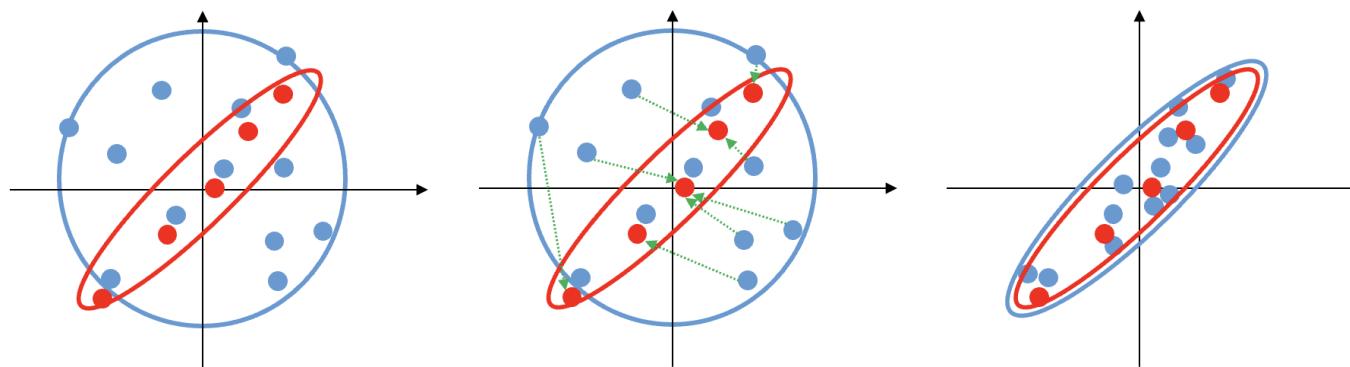


(b) Singular values of DCNv2 under different model size, with the dashed lines corresponding to the base size.

Interaction Collapse Theory

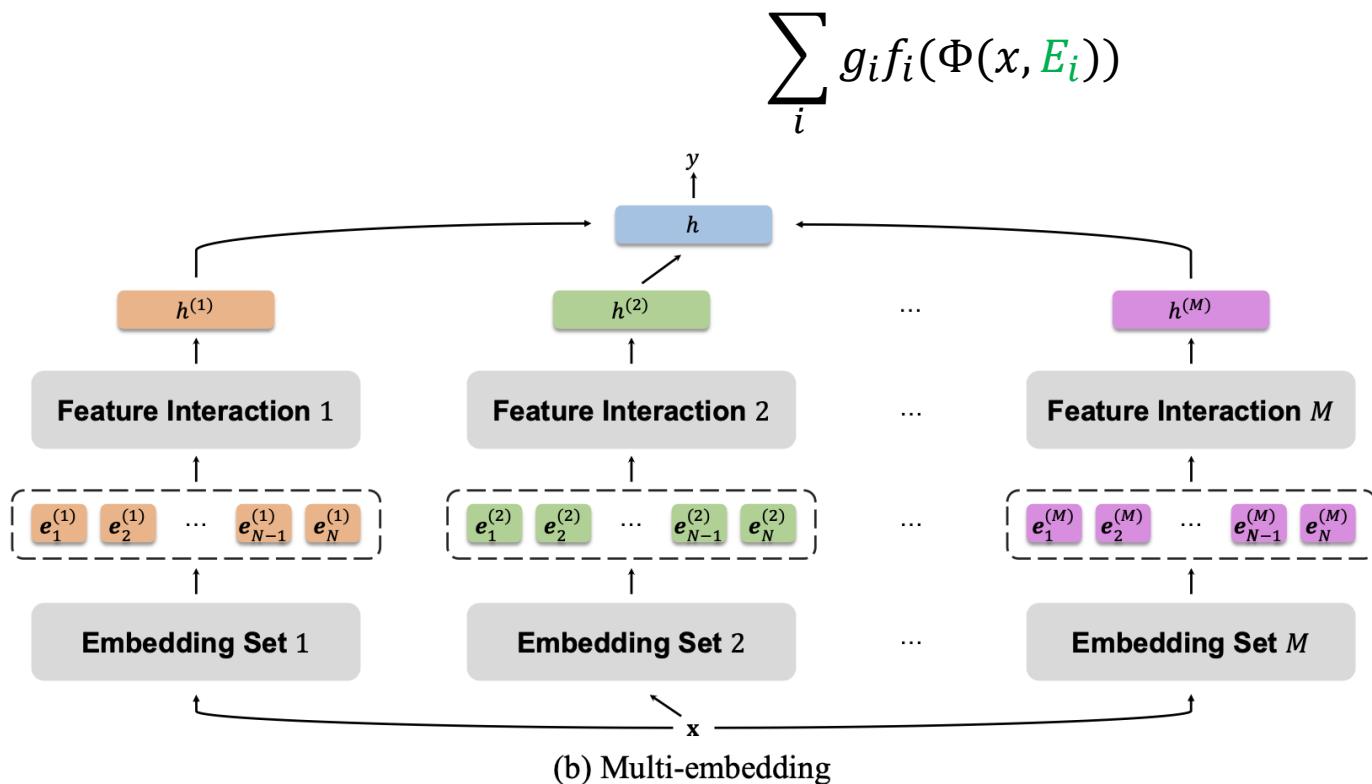


High-cardinality features (item ID) Low-cardinality features(Gender)

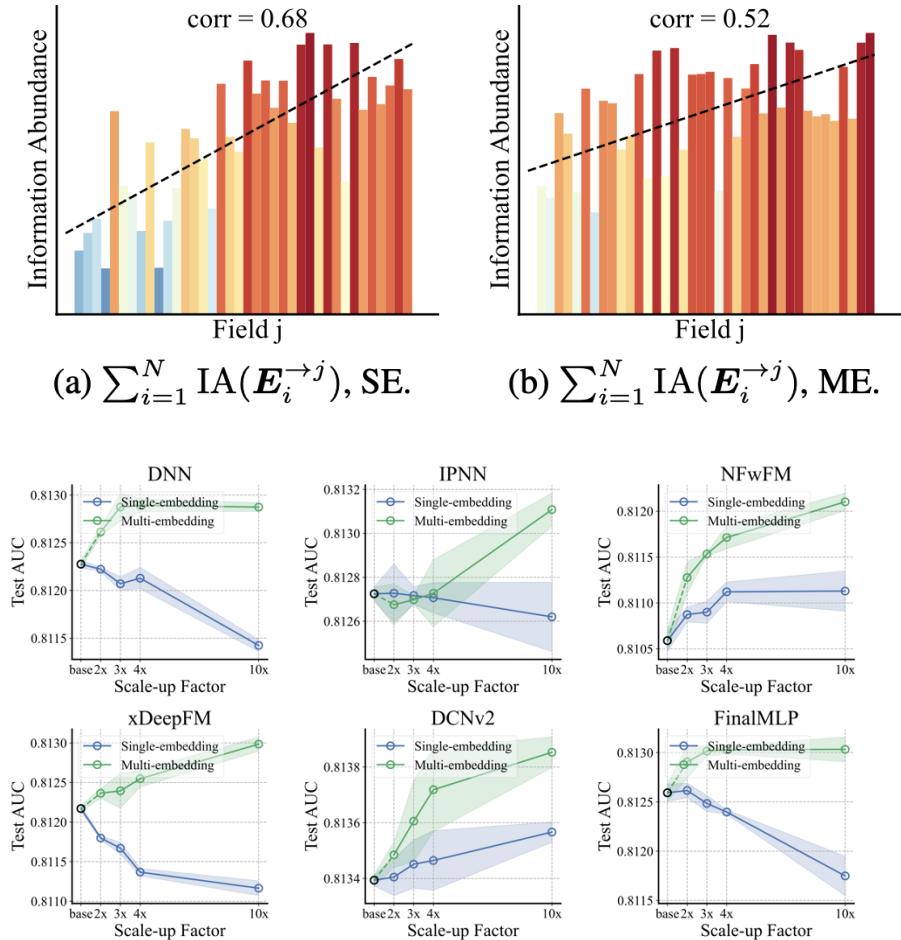


In feature interaction of recommendation models, fields with low-information-abundance (low cardinality) embeddings constrain the information abundance of other fields, resulting in collapsed embedding matrices.

Multi-Embedding Paradigm

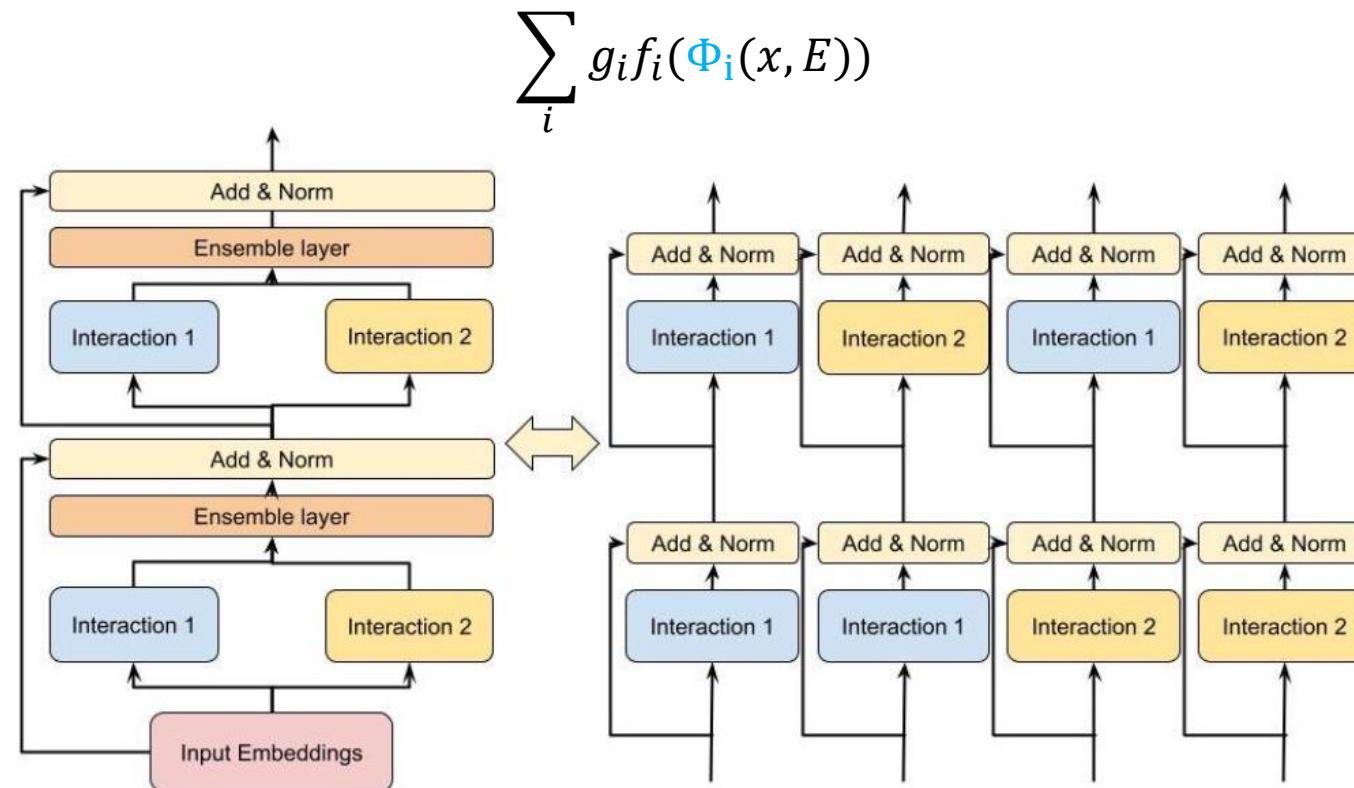


- Both ME and FFM learns **multiple embeddings** for each feature.
- Different motivation than STEM and Crocodile for Multi-task learning.



DHEN (Deep and Hierarchical Ensemble Network)

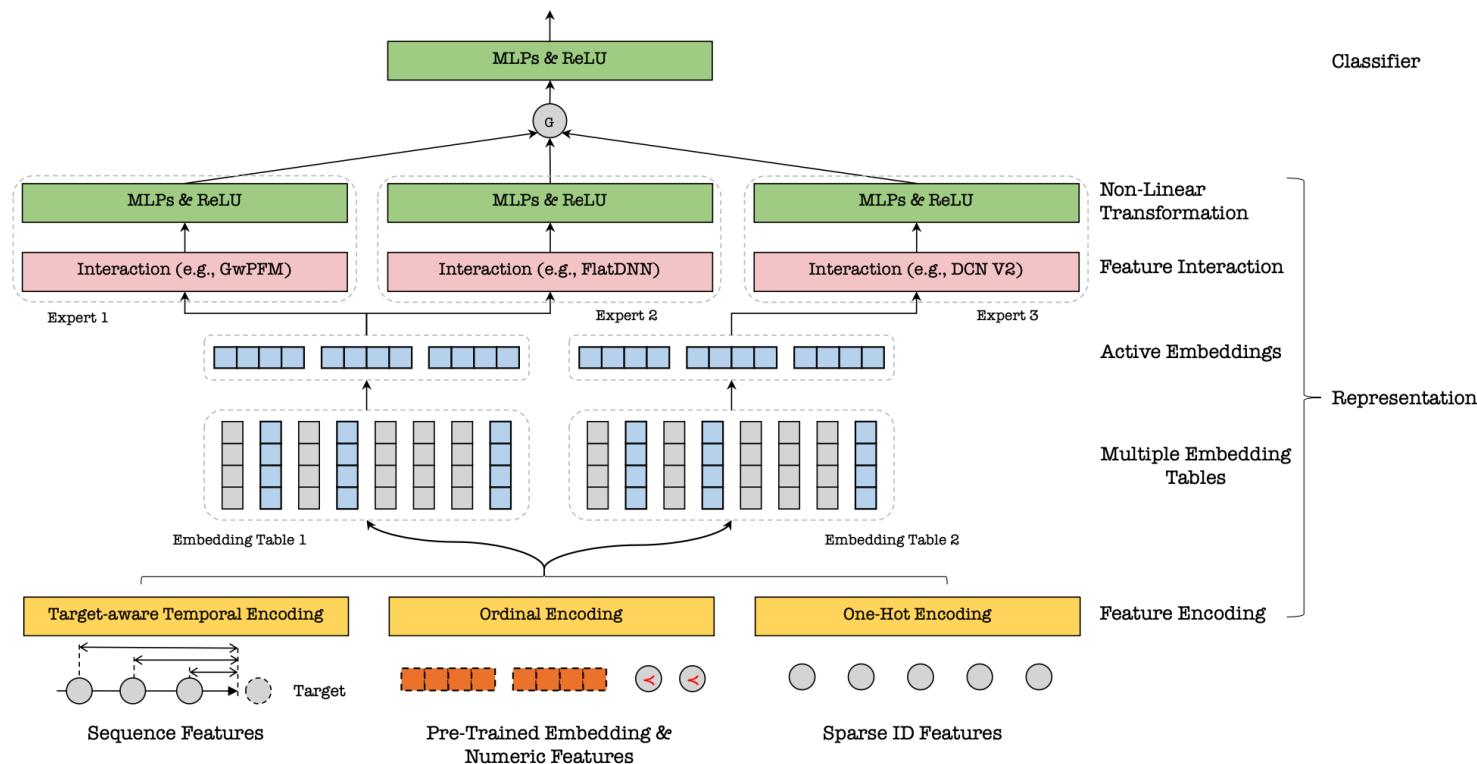
Heterogeneous interaction experts.



Hetero-ME (Heterogeneous Multi-Embedding)

Heterogeneous interaction experts, with multiple independent embeddings.

$$\sum_i g_i f_i(\Phi_i(x, E_i))$$

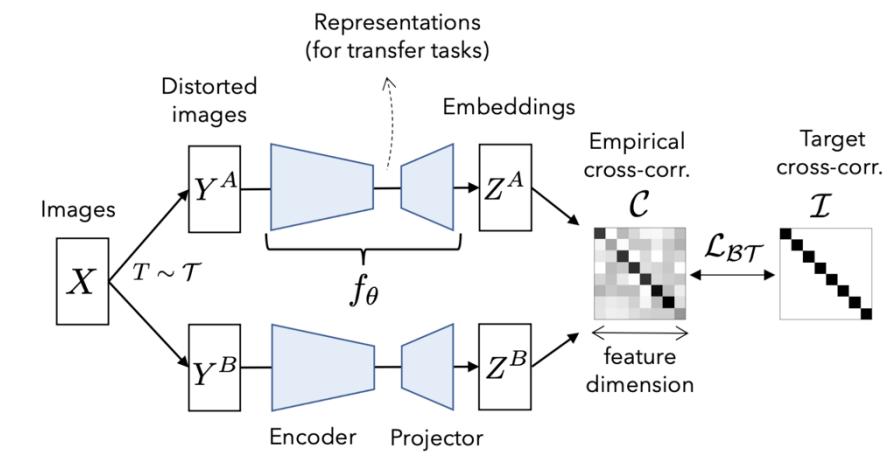
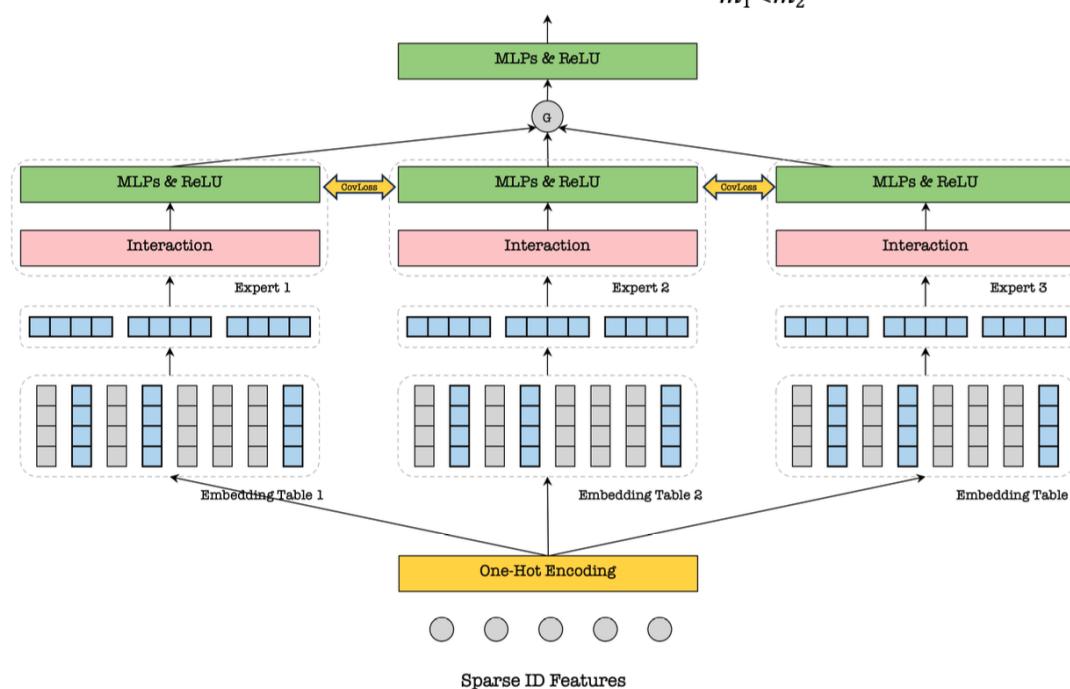


D-MoE (De-Correlated MoE)

Besides adopting heterogeneous architectures (DHEN), or different embedding dimensions, we can directly de-correlate the experts by an **Inter-Expert De-Correlation Loss**

$$L_{BCE} \left(\sum_i g_i f_i(\Phi_i(x, E_i)), y \right) + \sum_i \sum_j L_{Corr}(\Phi_i(x, E_i), \Phi_j(x, E_j))$$

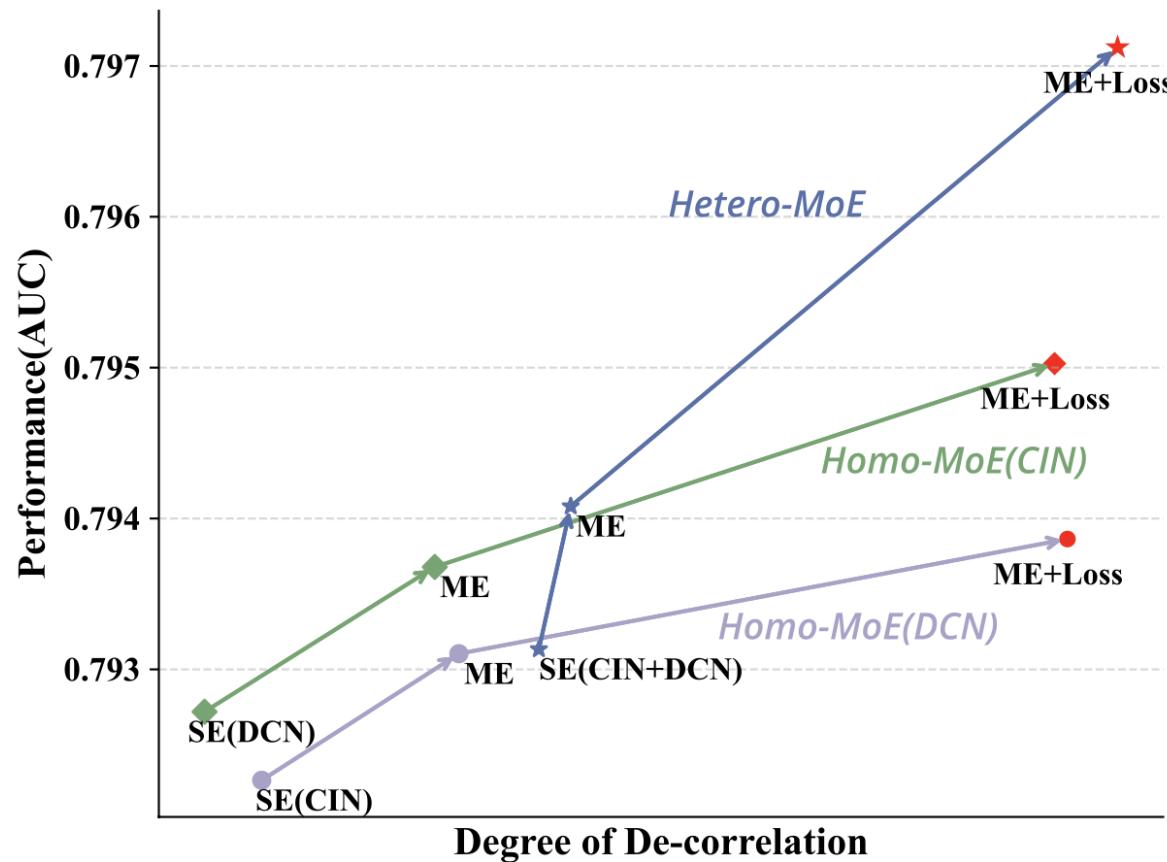
$$\mathcal{L}_{Corr} = \frac{1}{d^2} \sum_{\substack{m_1, m_2 \in \{1, \dots, M\} \\ m_1 < m_2}} \left\| \left[\frac{\mathbf{O}^{(m_1)} - \bar{\mathbf{O}}^{(m_1)}}{\sigma(\mathbf{O}^{(m_1)})} \right]^T \left[\frac{\mathbf{O}^{(m_2)} - \bar{\mathbf{O}}^{(m_2)}}{\sigma(\mathbf{O}^{(m_2)})} \right] \right\|_2$$



Barlow-Twins

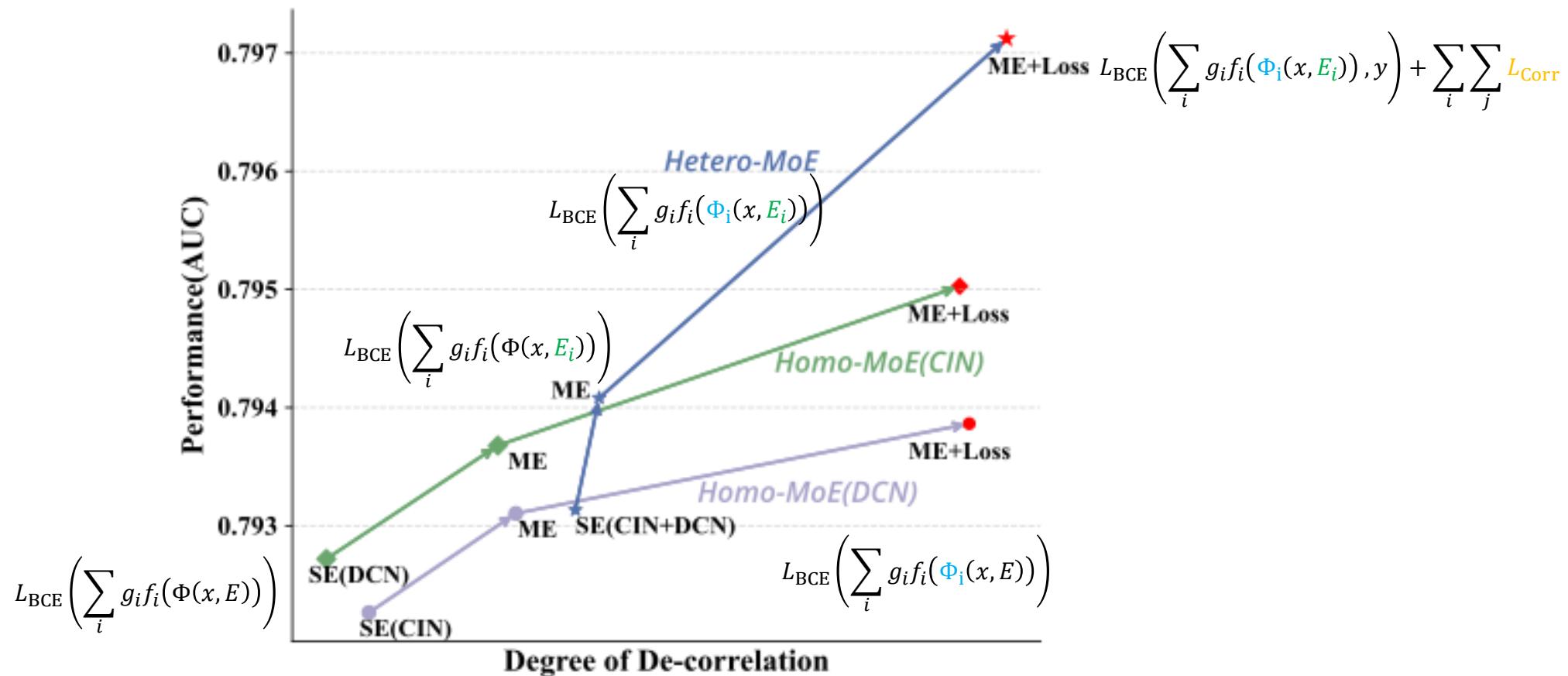
D-MoE - Redundancy Reduction

Applying the **Inter-Expert De-Correlation Loss** on ME-MoE or ME-Hetero-MoE can further reduce the correlation between behaviors.

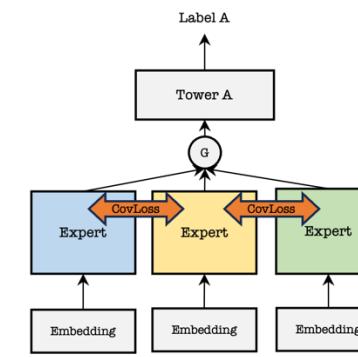
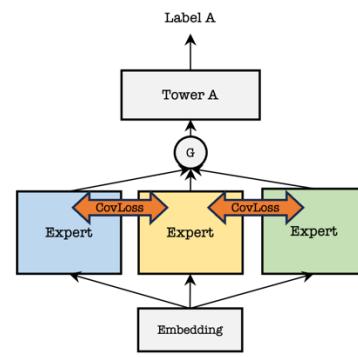
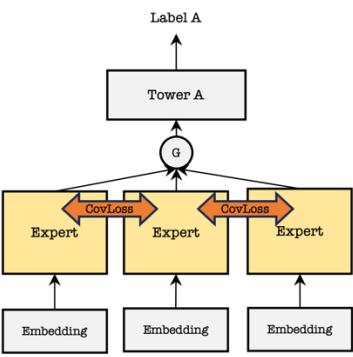
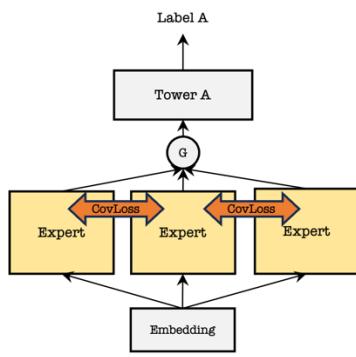
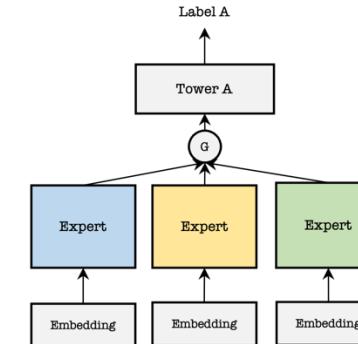
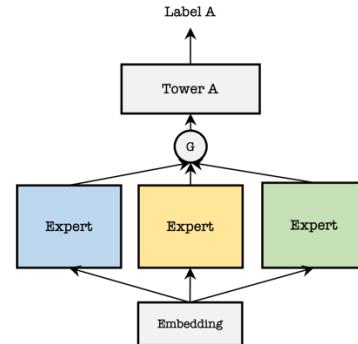
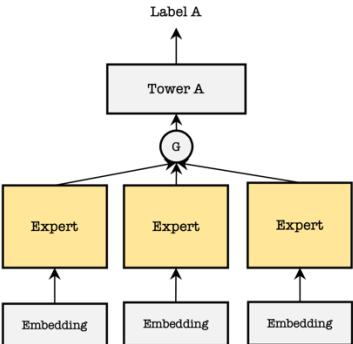
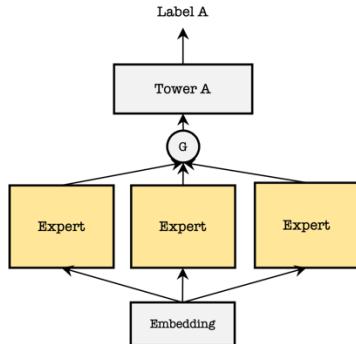


Expert De-Correlation Principle

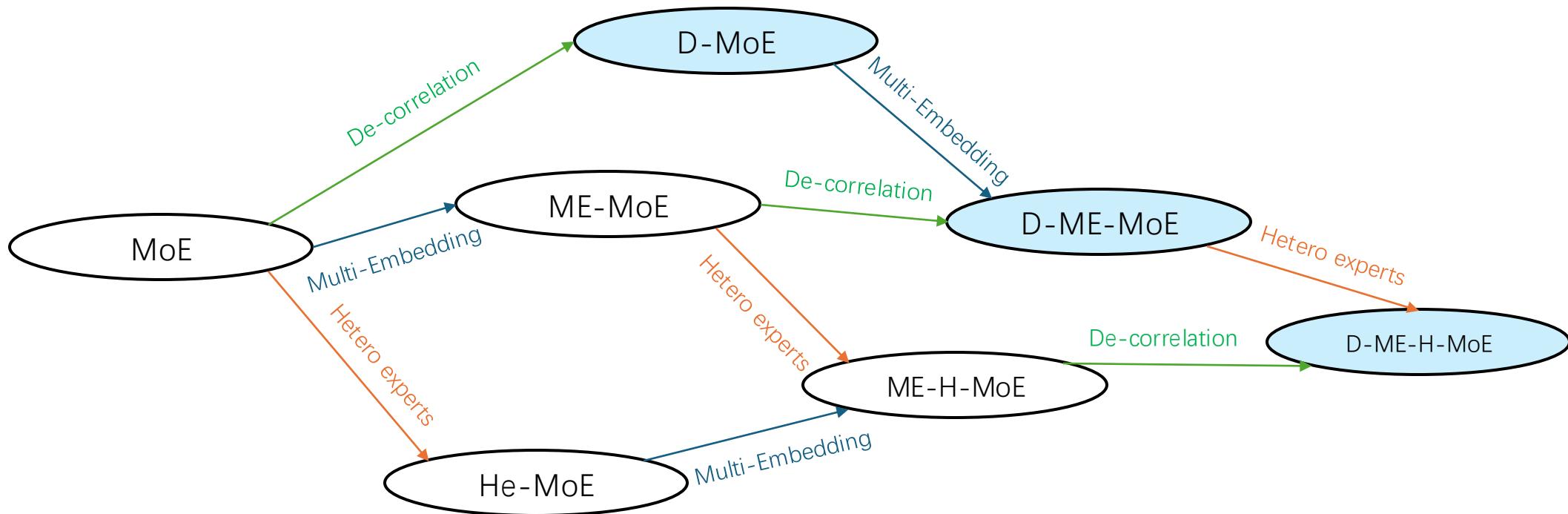
By combining the **Heterogeneous MoE**, **Multi-Embedding**, **Inter-Expert De-correlation**, we can progressively de-correlate the experts, while achieving performance lift.



MoE Model Zoo



MoE Evolution Map



- Part II, Prediction
 - Perspectives
 - Feature Interaction
 - Sequential Models
 - Multi-Task and Multi-Domain Learning
 - Large Recommendation Models
 - **LLM4Rec**

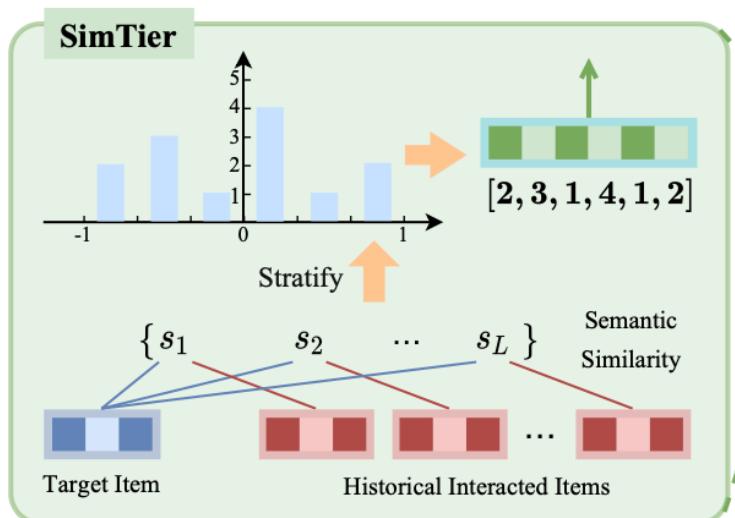
LLM4Rec

How to transfer the knowledge from LLM to RecSys?

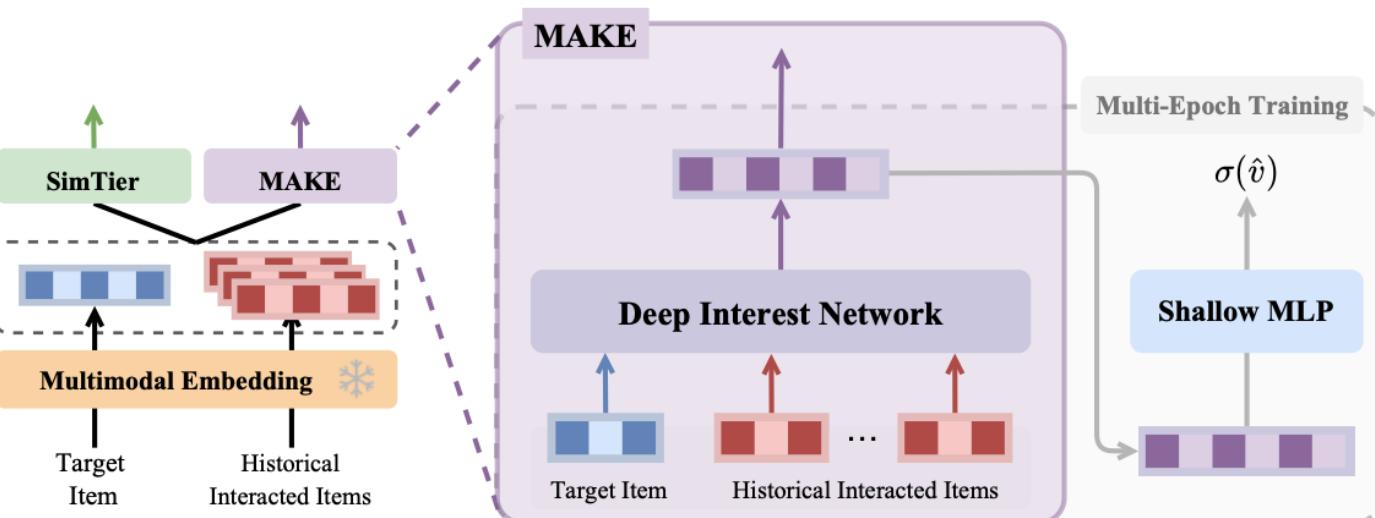
- Distance Transfer
- Embedding Alignment
- Semantic IDs

SimTier

- Calculate the **similarity score** between multi-modal representation of behaviors and the target.
- **Histogram** of the scores in a N pre-defined buckets.
- Use the N-dimension vector as a new representation.



(a). SimTier



(b). MAKE

MNSE (Multiple Numeric Systems Encoding)

- Calculate the **distance** between two features based on the source embedding.
- Encoding the distance with **n-ary**
- Train the encoded embeddings in the target task, together with other embeddings in the target space.

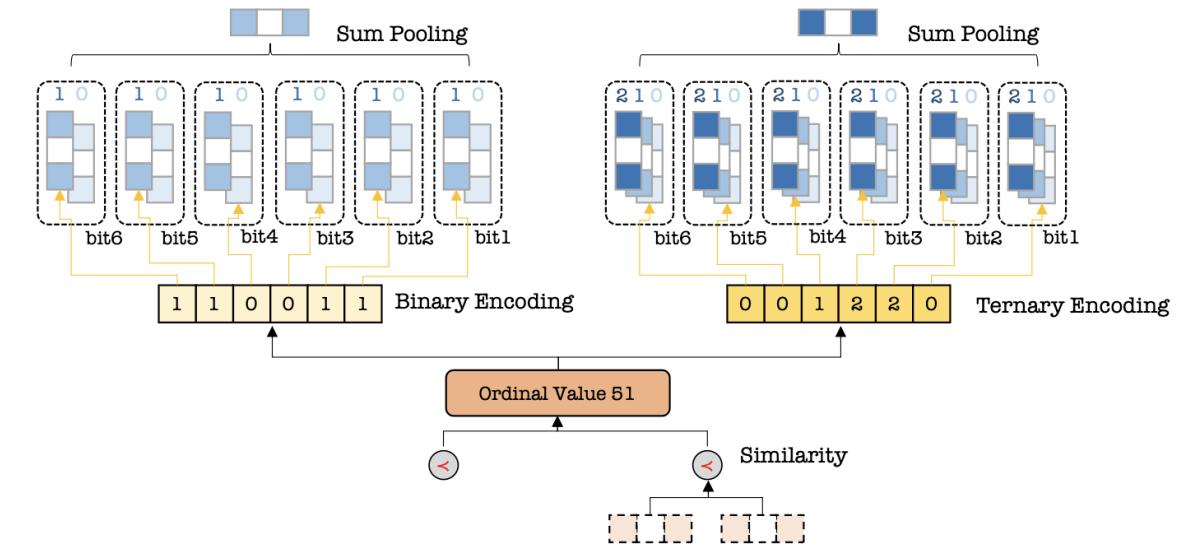
$$f_{\text{MNS}}(v) = [\sum_{k=1}^{K_2} \mathbf{X}_{2k+\mathbb{B}_k}^{(2)}, \sum_{k=1}^{K_3} \mathbf{X}_{3k+\mathbb{C}_k}^{(3)}, \dots, \sum_{k=1}^{K_n} \mathbf{X}_{nk+\mathbb{N}_k}^{(n)}]$$

$\mathbb{B} = \text{func_binary}(v), \mathbb{C} = \text{func_ternary}(v), \dots$

Numerical Feature (Decimal)	Binary	Ternary
45	0000101101	
46	0000101110	
957	<u>1110111101</u>	

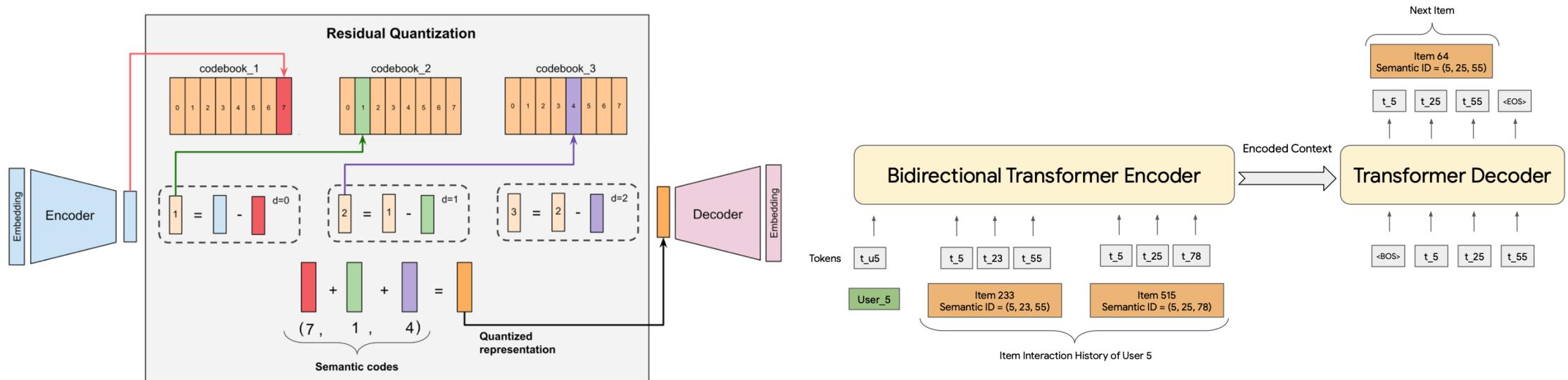
Numerical Feature (Decimal)	Binary	Ternary
63	0000111111	
64	000 <u>1000000</u>	
575	<u>1000111111</u>	<u>000210022</u>

Key properties of n-ary: continuity, discriminability



Tiger

- Use **RQ-VAE** on the LLM representation to get **semantic IDs**.
- Use semantic IDs in the downstream recommendation models.

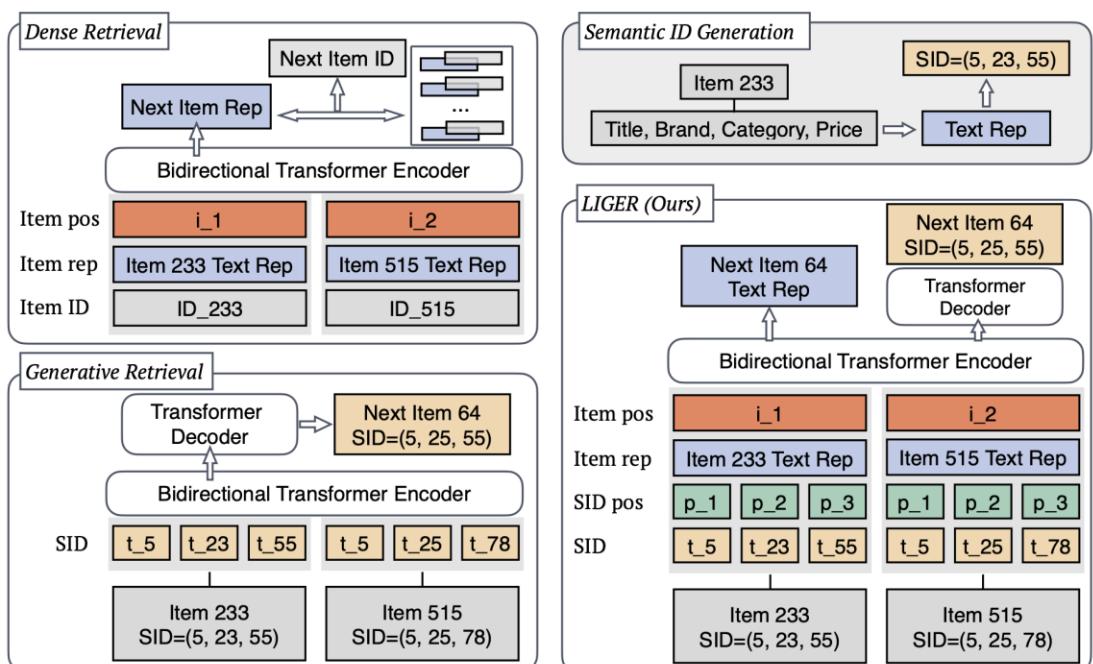
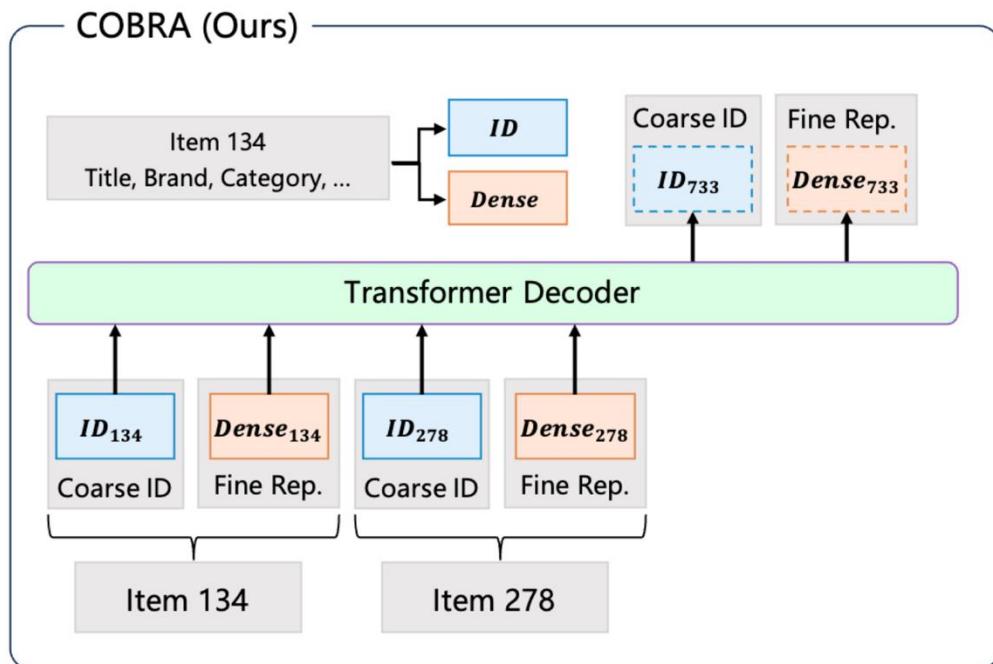


Why it works?

- **RQ-VAE** and **Semantic IDs** to capture the **source** space structures.
- **Semantic ID Embeddings** to align with the **target** space.

COBRA, LIGER

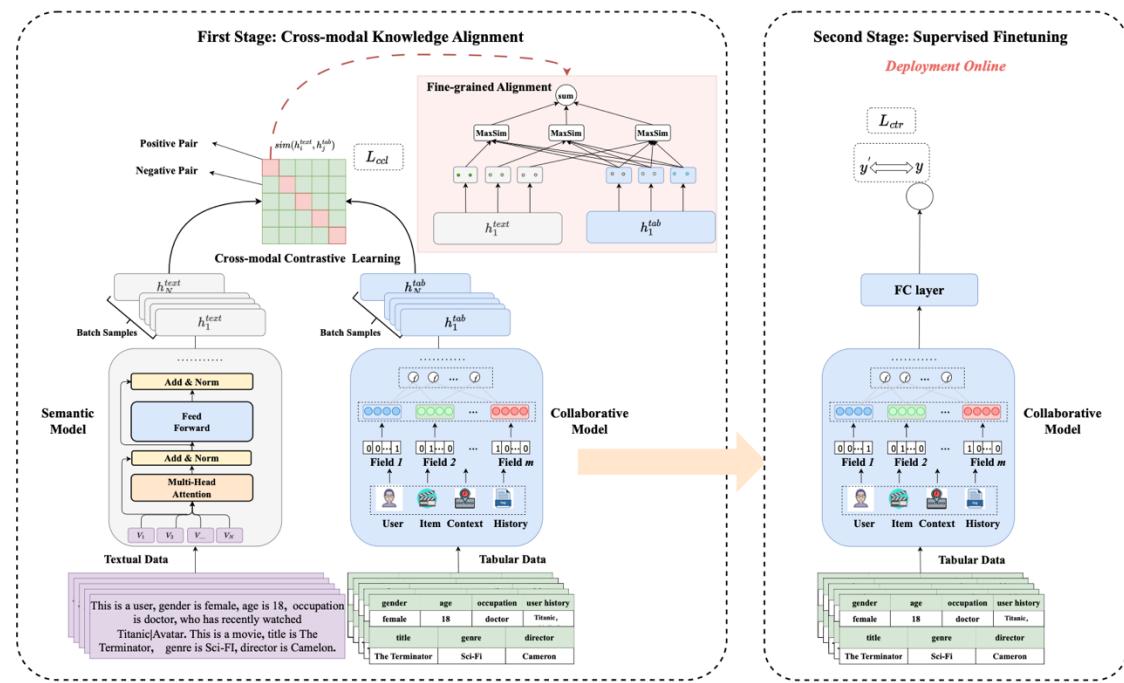
- Utilize both **sparse** (Semantic IDs through **RQ-VAE**) and **dense** representations to extract knowledge from the LLM.



CTRL

Taxtual-to-Tabular **contrastive loss**

$$\mathcal{L}^{textual2tabular} = -\frac{1}{N} \sum_{k=1}^N \log \frac{\exp(\text{sim}(\mathbf{h}_k^{text}, \mathbf{h}_k^{tab})/\tau)}{\sum_{j=1}^N \exp(\text{sim}(\mathbf{h}_k^{text}, \mathbf{h}_j^{tab})/\tau)},$$



PAD (Pre-train, Align and Disentangle)

Adopt **MK-MMD** as the alignment loss to capture all information about the distribution.
 Combine the alignment with the BCE loss to avoid **catastrophic forgetting**.

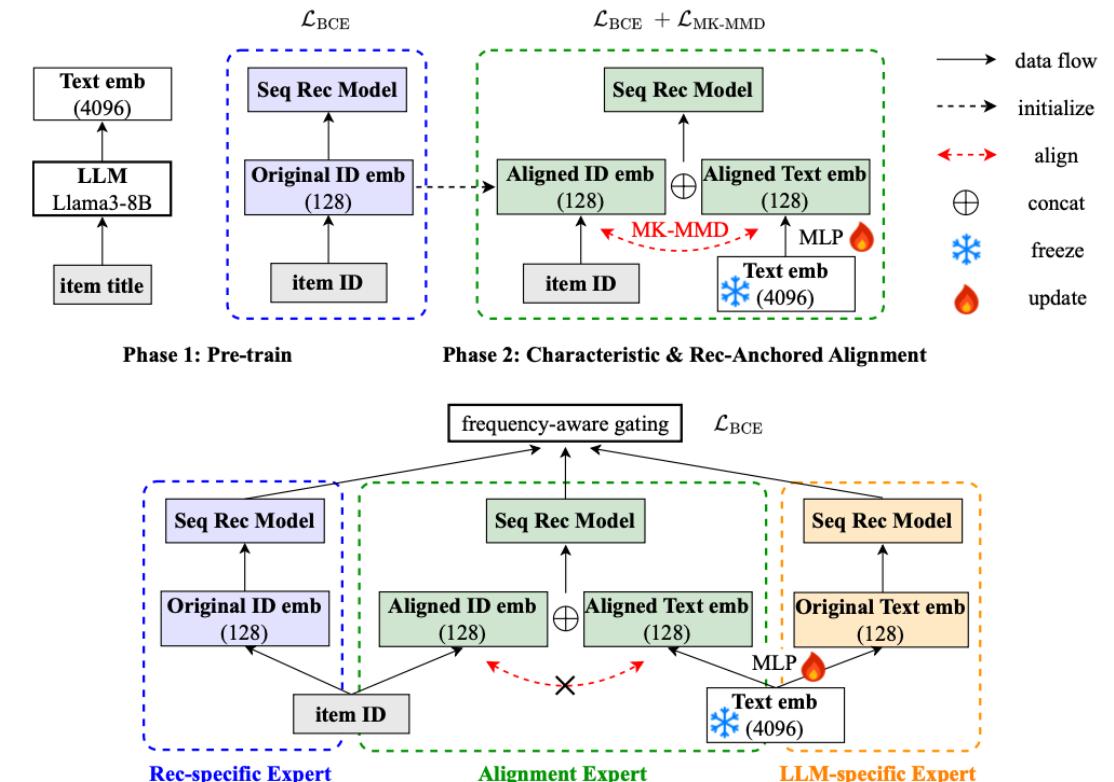
Avoid **catastrophic forgetting** **Space Alignment**

$$\mathcal{L} = \boxed{\mathcal{L}_{\text{REC}}} + \gamma \cdot \boxed{\mathcal{L}_{\text{MK-MMD}}}$$

$$\mathcal{L}_{\text{MK-MMD}} = D_k^2(f_{\text{MLP}}(\text{SG}(\mathcal{D}_{\text{text}}), \mathbf{w}), \mathcal{D}_{\text{rec}})$$

$$\mathcal{L}_{\text{REC}} = \frac{1}{n} \sum_{i=1}^n \text{BCE}\left(f_{\theta}\left(\{\mathbf{h}_i^s\}, \{\mathbf{h}_i^c\}, \mathbf{x}_i^s, \mathbf{x}_i^c\right), y_i\right)$$

$$\text{MK-MMD}^2(X_s, X_t) = \left\| \frac{1}{n} \sum_{i=1}^n \phi_k(x_s^i) - \frac{1}{m} \sum_{j=1}^m \phi_k(x_t^j) \right\|_{\mathcal{H}_k}^2$$



Summary

- Perspectives
- Feature Interaction Models
- Sequential Models
- Multi-task/domain Learning
- Large Recommendation Models
- LLM4Rec

References, |

- Factorization Machines. ICDM 2010.
- Field-aware Factorization Machines. RecSys 2016.
- **Field-weighted Factorization Machines for Click-Through Rate Prediction in Display Advertising.** WWW 2018.
- **FmFM: Field-matrixed Factorization Machines for CTR Prediction.** WWW 2021.
- **Towards Unifying Feature Interaction Models for Click-Through Rate Prediction.** 2024.
- Wide & Deep Learning for Recommender Systems. 2016.
- DeepFM: A Factorization-Machine based Neural Network for CTR Prediction. 2017.
- xDeepFM: Combining Explicit and Implicit Feature Interactions for Recommender Systems. KDD 2018.
- DCN v2- Improved Deep & Cross Network and Practical Lessons for Web-scale Learning to Rank Systems.
- Neural Collaborative Filtering vs. Matrix Factorization Revisited. RecSys 2020.
- **Understanding DNNs in Feature Interaction Models - A Dimensional Collapse Perspective.** 2025.
- **From Feature Interaction to Feature Generation: A Generative Paradigm of CTR Prediction Models.** Under review.
- **Deep Interest Network for Click-Through Rate Prediction.** KDD 2018.
- **Deep interest evolution network for click-through rate prediction.** AAAI 2019.

References, ||

- Deep Session Interest Network for Click-Through Rate Prediction. 2019.
- Temporal Interest Network for User Response Prediction. WWW 2024.
- Search-based User Interest Modeling with Lifelong Sequential Behavior Data for Click-Through Rate Prediction. CIKM 2020.
- Sampling is all you need on modeling long-term user behaviors for CTR prediction. CIKM 2022.
- ETA: End-to-End User Behavior Retrieval in Click-Through Rate Prediction Model. 2021.
- TWIN: Two-stage Interest Network for Lifelong User Behavior Modeling in CTR Prediction at Kuaishou.
- TWIN V2: Scaling Ultra-Long User Behavior Sequence Modeling for Enhanced CTR Prediction at Kuaishou. CIKM 2024.
- Long-Sequence Recommendation Models need Decoupled Embeddings. ICLR 2025.
- Lighter and Better: Low-Rank Decomposed Self-Attention Networks for Next-Item Recommendation. 2021.
- Lifelong Sequential Modeling with Personalized Memorization for User Response Prediction. SIGIR 2019.
- Practice on Long Sequential User Behavior Modeling for Click-Through Rate Prediction. KDD 2019.
- Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-Experts. KDD 2018.
- Progressive Layered Extraction (PLE): A Novel Multi-Task Learning (MTL) Model for Personalized Recommendations. RecSys, 2020.
- PEPNet: Parameter and Embedding Personalized Network for Infusing with Personalized Prior Information. 2023.

References, III

- STEM: Unleashing the Power of Embeddings for Multi-Task Recommendation. AAAI 2024.
- Ads Recommendation in a Collapsed and Entangled World. KDD 2024.
- Crocodile - Cross Experts Covariance for Disentangled Learning in Multi-Domain Recommendation. Under review.
- Actions Speak Louder than Words: Trillion-Parameter Sequential Transducers for Generative Recommendations. ICML 2024.
- Wukong: Towards a Scaling Law for Large-Scale Recommendation. ICML 2024.
- On the Embedding Collapse When Scaling Up Recommendation Models. ICML 2024.
- DHEN: A Deep and Hierarchical Ensemble Network for Large-Scale Click-Through Rate Prediction. 2022.
- Towards De-correlated Mixture-of-Experts for CTR Prediction. Under review.
- Enhancing Taobao Display Advertising with Multimodal Representations: Challenges, Approaches and Insights.
- Recommender Systems with Generative Retrieval. NIPS 2023.
- Sparse Meets Dense: Unified Generative Recommendations with Cascaded Sparse-Dense Representations. 2025.
- CTRL: Connect Collaborative and Language Model for CTR Prediction.
- Pre-train, Align and Disentangle: Empowering Sequential Recommendation with Large Language Models. SIGIR 2025.

Thanks!

Q & A