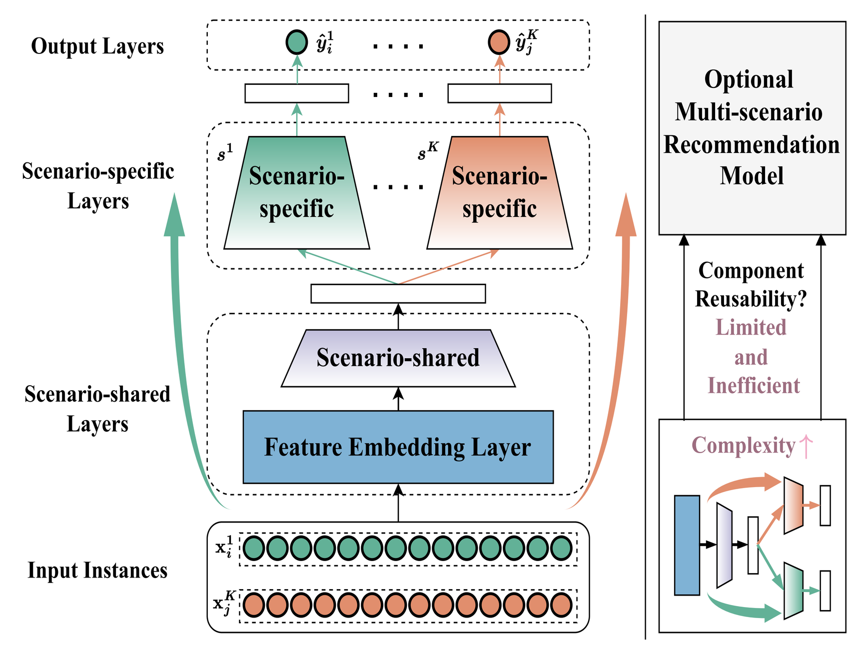
Scenario Shared Instance Modeling for Click-through Rate Prediction

Dugang Liu[1], Chaohua Yang[1], Yuwen Fu[2], Xing Tang[2], Gongfu Li[2]
Fuyuan Lyu[3], Xiuqiang He[4], Zhong Ming[4]

[1]Shenzhen University, China
[2]Financial Technology (FiT), Tencent, China
[3]McGill University, Canada
[4]Shenzhen Technology University, China

Background

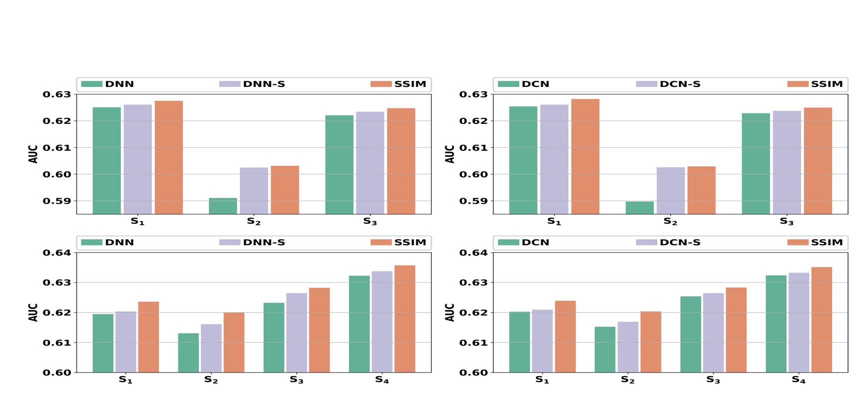
Multi-scenario recommendation (MSR) is a popular training paradigm in industrial platforms for uniformly integrating information from multiple scenarios and serving them simultaneously. A key challenge in MSR research is accurately identifying the commonalities and distinctive information between scenarios.



- Existing MSR methods focus on implicitly extracting this information from the architectural level.
- This continues to increase the complexity and training overhead of MSR.
- The implicit extractors in MSR methods are tightly coupled with specific architectures and hard to reuse.

Motivation

To this end, we argue that we can switch perspectives from implicit information extraction at the architecture level to explicit extraction at the data level. A potential solution is to directly share valuable instances across scenarios to enhance model training for all scenarios in MSR. The idea behind it is that some MSR instances carry key information and serve as hubs for propagation. Furthermore, since only a simple reuse operation of instances is required and the operation is model-free, and selected shared instances are easy to save and reuse.



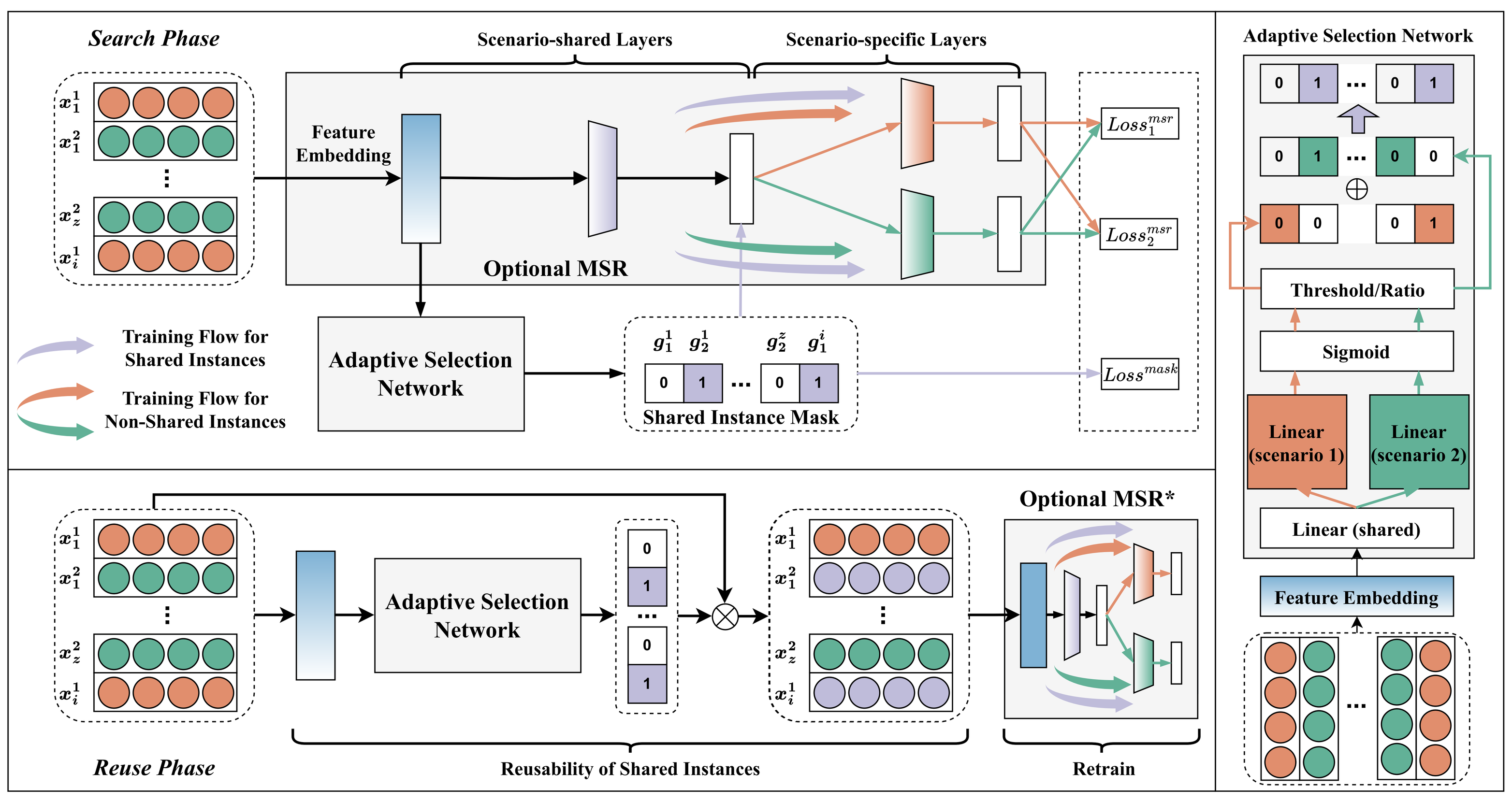
- We conduct a preliminary experiment to verify the feasibility of this potential idea.
- This result motivates us to develop a training framework to better select shared instances in MSR.

Problem Formulation

Based on the training instances set $\mathcal{D}$, multi-scenario learning aims to train a model $\hat{y}_i^k = f(\mathbf{x}_i^k \mid \theta, \{\theta_{s^k}\})$ to serve multiple scenarios. The cross-entropy function is usually used to optimize the model, $\mathcal{L} = \sum_{k=1}^{K}\sum_{i=1}^{|s^k|} l(y_i^k, \hat{y}_i^k)$. Then, training instances $\mathcal{D}$ are typically organized into n mini-batches containing instances from multiple scenarios, $\mathbf{B} = [B_1, B_2, \cdots, B_n]$, where the j-th mini-batch with m instances is denoted as $B_j = \{(\mathbf{x}_{j,i}^{\star}, y_{j,i}^{\star}, s^{\star})\}_{i=1}^{m}$. Hence, the shared instance selection problem can be defined as learning the shared instance mask operation $\mathbf{G}$ to select shared instances beneficial for all scenarios from $\mathbf{B}$,

$$\mathbf{B} \odot \mathbf{G} \rightarrow \mathbf{B}^{sh} = [B_1', B_2', \cdots, B_n'], \tag{1}$$

Architecture



The SSIM framework includes two phases: search and retrain, and two key components: the backbone MSR model and the adaptive selection network.

The Adaptive Selection Network

- The search phase aims to obtain an adaptive selection network that can efficiently select a set of scenario-shared instances.
- We propose an adaptive selection network with a hyper-network learning structure. After obtaining each instance's embedding $\mathbf{e}_{j,i}^k$, we can calculate a shared instance mask for each instance as,

$$\begin{aligned} \mathbf{h}_{j,i}^l &= \sigma\left(\mathbf{W}^l \mathbf{h}_{j,i}^{l-1} + \mathbf{b}^l\right), \quad l \in [1, L], \\ p_{j,i} &= \alpha\left(\mathbf{W}^k \mathbf{h}_{j,i}^L + \mathbf{b}^k\right) \leftarrow (\mathbf{x}_{j,i}^k, s^k), \\ g_{j,i} &= S(\mathbf{relu}(\mathbf{p}_{j,i} - \epsilon)), \end{aligned} \tag{2}$$

where σ, α and $S(\cdot)$ are $ReLu$, $Sigmoid$ and STE operation.

- Through $g_{j,i}$, we can obtain $\mathbf{B}^{sh}$. Then we need to introduce the required optimization constraints,

$$\begin{aligned} \mathcal{L}^{msr} &= \mathcal{L}(\mathcal{D} \setminus \mathbf{B}^{sh}) + \mathcal{L}^{sh}(\mathbf{B}^{sh}), \ \mathcal{L}^{mask} = D_{KL}(\mathbf{1} \| \mathbf{G}) \\ \min_{\Theta, \mathbf{G}} \mathcal{L}_{SSIM} &= \mathcal{L}^{msr} + \lambda \mathcal{L}^{mask}. \end{aligned} \tag{3}$$

The Shared Instance Reuse

- In the reuse phase, we can construct a "new training set" that includes the shared instances selected by the well-trained adaptive selection network and the remaining scenario-specific instances to train various MSR models.
- After determining the optimal selection ratio, assuming that the current shared instance is represented as $\overline{\mathbf{B}}^{sh}$, we constrain the newly trained MSR* to achieve an optimal performance.

$$\min_{\Theta} \mathcal{L}(\mathcal{D} \setminus \overline{\mathbf{B}}^{sh}) + \mathcal{L}^{sh}(\overline{\mathbf{B}}^{sh}). \tag{4}$$

Result 1

	Method	AliCCP						AliMama							
		AUC↑			Logloss↓			AUC↑				Logloss↓			
DNN	ST-Backbone	.6246	.6023	.6220	.1661	.1794	.1600	.6184	.6144	.6245	.6323	.2011	.2023	.1957	.1926
	Finetune	.6250	.6025	.6222	.1658	.1794	.1598	.6197	.6155	.6245	.6328	.2013	.2018	.1965	.1931
	MT-Backbone	.6250	.5910	.6221	**.1652**	.1797	.1604	.6195	.6131	.6232	.6323	.2006	.2019	.1960	**.1919**
	MT-Backbone-S	.6261	.6024	.6234	.1656	**.1789**	.1596	.6203	.6161	.6264	.6337	.2008	.2019	.1953	.1924
	SSIM	**.6275***	**.6031***	**.6247***	.1659	.1795	**.1595**	**.6236***	**.6199***	**.6282***	**.6352***	**.1999***	**.2009***	**.1947***	**.1919**
DeepFM	ST-Backbone	.6248	.6024	.6229	.1662	.1800	.1601	.6202	.6158	.6242	.6303	.2010	.2020	.1957	.1933
	Finetune	.6249	.6027	.6230	.1659	.1796	.1597	.6205	.6160	.6245	.6304	.2006	.2017	.1967	.1981
	MT-Backbone	.6252	.5931	.6227	**.1648**	.1837	.1595	.6202	.6152	.6255	.6323	.2001	.2011	.1946	.1928
	MT-Backbone-S	.6266	.6024	.6237	.1653	.1791	.1594	.6204	.6158	.6258	.6330	.2000	.2010	.1944	**.1919**
	SSIM	**.6279***	**.6031**	**.6254***	.1649	**.1785***	**.1591**	**.6234***	**.6199***	**.6282***	**.6352***	**.1996**	**.2007***	**.1943**	.1921
DCN	ST-Backbone	.6251	.6007	.6223	.1651	.1784	.1591	.6207	.6161	.6245	.6306	.2007	.2017	.1955	.1931
	Finetune	.6257	.6008	.6225	.1650	.1787	**.1590**	.6209	.6169	.6259	.6319	.2003	.2015	.1949	.1935
	MT-Backbone	.6254	.5897	.6228	.1651	.1797	.1599	.6202	.6152	.6254	.6324	.2000	.2011	.1946	.1928
	MT-Backbone-S	.6265	.6020	.6236	.1654	.1788	.1592	.6209	.6169	.6264	.6332	.2004	.2016	.1952	.1923
	SSIM	**.6281***	**.6029***	**.6249***	**.1649**	**.1782**	.1593	**.6239***	**.6203***	**.6283***	**.6351***	**.1994***	**.2003***	**.1942**	**.1921**
MSR	STAR	.6240	.5880	.6191	.1691	.1817	.1604	.6152	.6112	.6216	.6283	.2048	.2096	.1985	.1933
	HMoE	.6220	.5952	.6182	.1665	.1809	.1601	.6164	.6099	.6232	.6279	.2006	.2022	.1949	.1923
	DFFM	.6251	.6022	.6220	**.1657**	**.1795**	.1598	.6216	.6157	.6245	.6319	.2003	.2013	.1947	.1924
	STAR+SSIM	**.6277***	.5975	**.6260***	.1736	.3130	.1608	.6223	.6189	.6272	**.6352***	.2141	.2135	.2024	**.1916**
	HMoE+SSIM	.6262	**.6025**	.6232	**.1657**	.1796	**.1596**	**.6244***	**.6205***	**.6281***	.6350	**.1996***	**.2005***	**.1945**	.1919

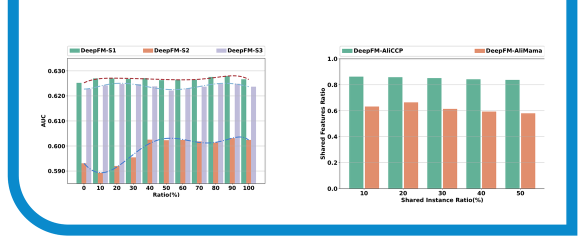
Result 2

Model	Methods	AliMama							
		AUC↑				Logloss↓			
DNN	SSIM-R	.6203	.6155	.6232	.6323	.2000	.2021	.1960	**.1919**
	SSIM-P	.6193	.6135	.6225	.6331	.2363	.2345	.2241	.2166
	SSIM-NK	.6188	.6147	.6239	.6321	.2005	.2023	.1958	.1923
	SSIM-ND	.6200	.6158	.6238	.6240	.2006	.2012	.1951	.1931
	SSIM	**.6236**	**.6199**	**.6282**	**.6352**	**.1999**	**.2009**	**.1947**	**.1919**

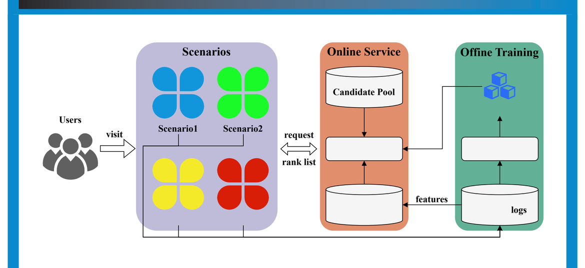
Result 3

Target	Source	AliMama							
		AUC↑				Logloss↓			
STAR	DNN	**.6237**	.6195	.6281	**.6353**	.2138	.2127	.2020	.1916
	DeepFM	.6226	.6195	.6278	.6352	**.2109**	**.2105**	**.2007**	.1916
	DCN	.6231	**.6201**	**.6283**	.6349	.2160	.2147	.2034	.1916
HMoE	DNN	**.6244**	.6202	.6279	.6350	**.1996**	.2006	.1946	**.1919**
	DeepFM	.6241	.6202	.6279	**.6351**	.1997	.2006	.1946	.1920
	DCN	**.6244**	**.6204**	.6279	.6349	**.1996**	.2006	**.1945**	**.1919**

Result 4



Result 5



Metrics	Scenario1	Scenario2	Scenario3	Scenario4
CTR	3.15%	2.08%	0.93%	0.92%
SA	3.07%	1.92%	7.71%	14.81%