Optimizing Feature Set for Click-Through Rate Prediction

Fuyuan Lyu*
McGill University
Montreal, Canada
fuyuan.lyu@mail.mcgill.ca

Xing Tang*
FiT, Tencent
Shenzhen, China
shawntang@tencent.com

Dugang Liu†‡
Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ)
Shenzhen, China
dugang.ldg@gmail.com

Liang Chen
FiT, Tencent
Shenzhen, China
leocchen@tencent.com

Xiuqiang He‡
FiT, Tencent
Shenzhen, China
xiuqianghe@tencent.com

Xue Liu
McGill University
Montreal, Canada
xueliu@cs.mcgill.ca

WWW 2023

Code: https://github.com/fuyuanlyu/OptFS

Reported by liang li

Motivation

Details：

Most previous works focus on either feature field selection or only select feature interaction based on the fixed feature set to produce the feature.

- The former restricts search space to the feature field, which is too coarse to determine subtle features. They also do not filter useless feature interactions, leading to higher computation costs and degraded model performance.
- The latter identifies useful feature interaction from all available features, resulting in many redundant features in the feature set.

Problem Statement

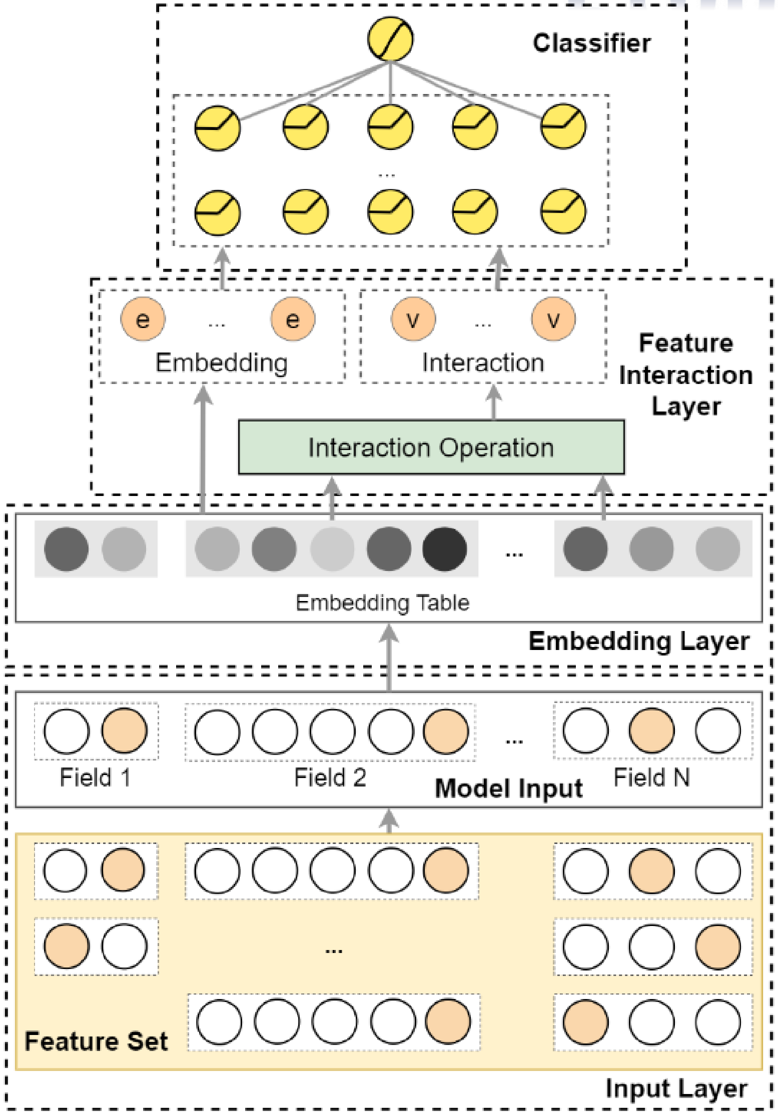


Figure 1: Overview of the general CTR framework.

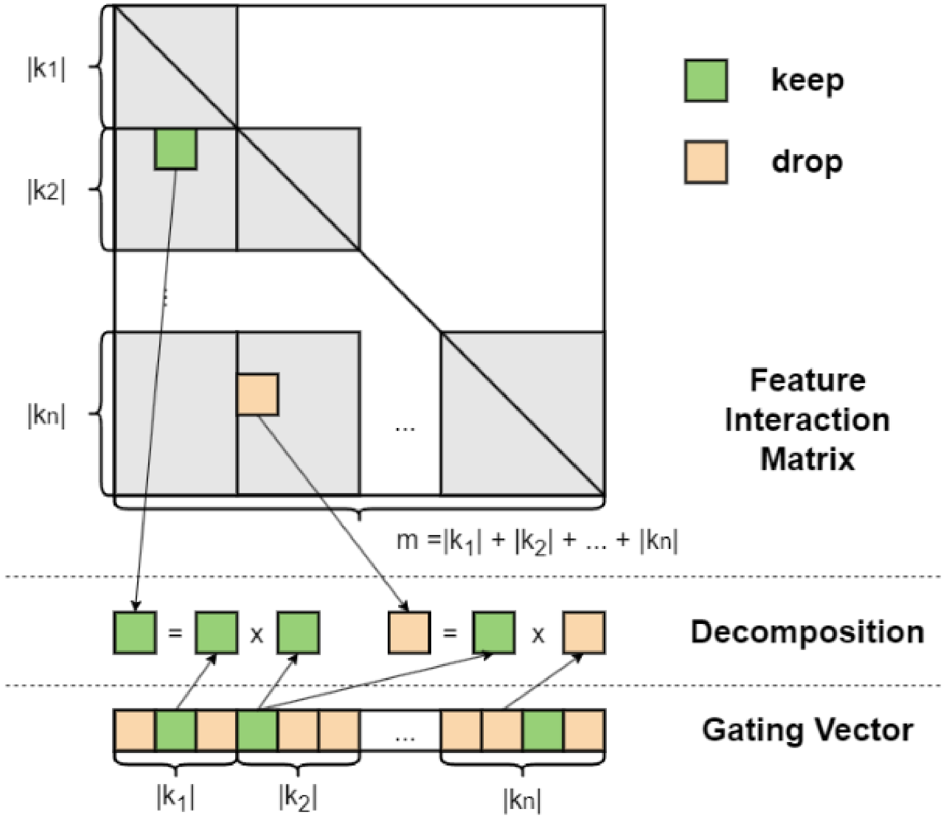


Figure 2: The Overview of OptFS.

$$X = \{x_1, x_2, \cdots, x_m\}$$

Method

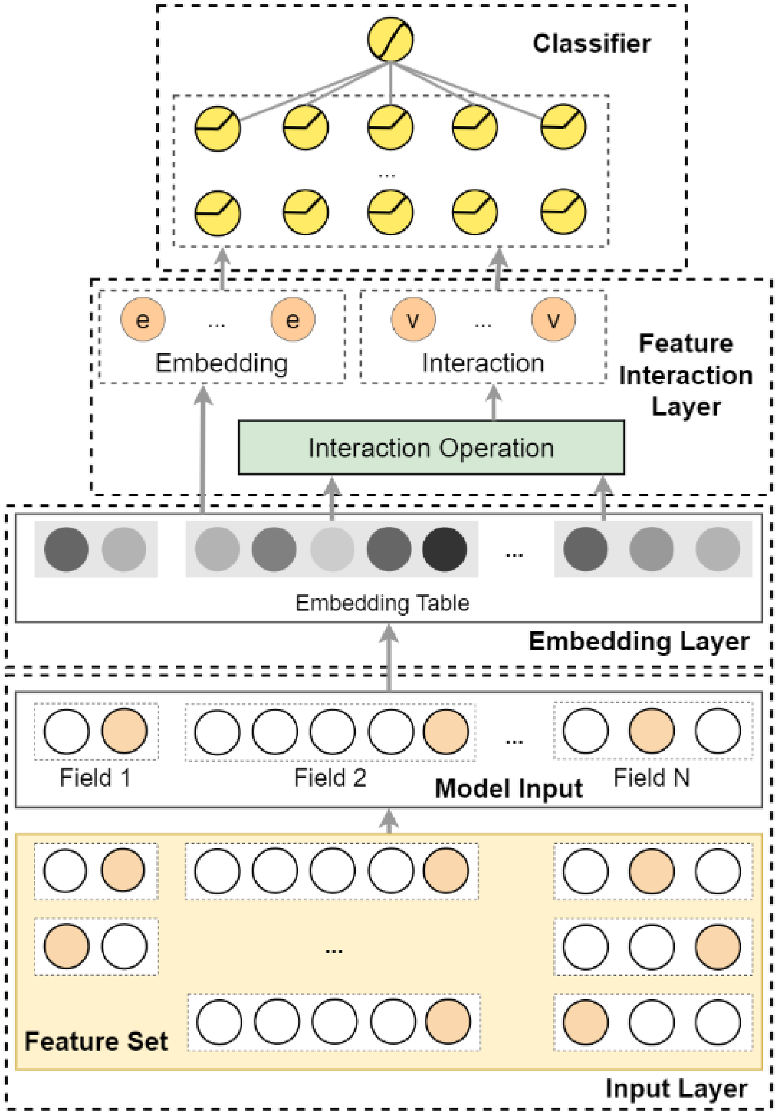


Figure 1: Overview of the general CTR framework.

$$\min_{\mathbf{W}} \mathcal{L}(\mathcal{D}|\mathbf{W}),\ \mathcal{D} = \{\mathbf{X}^g, \mathbf{Y}\},$$
$$s.t. \forall \mathbf{x} \in \mathbf{X}^g, \mathcal{L}(\mathbf{X}^g) > \mathcal{L}(\mathbf{X}^g - \{\mathbf{x}\}), \tag{1}$$
$$\forall \mathbf{x} \notin \mathbf{X}^g, \mathcal{L}(\mathbf{X}^g) \geq \mathcal{L}(\mathbf{X}^g + \{\mathbf{x}\}),$$

$$\mathbf{z}_i = \{\mathbf{x}_{k_i}\},\ 1 \leq k_i \leq m, \tag{2}$$

$$\mathbf{z} = [\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_n] = [\mathbf{x}_{k_1}, \mathbf{x}_{k_2}, \cdots, \mathbf{x}_{k_n}], \tag{3}$$

$$\mathbf{e}^g_{k_i} = \mathbf{g}_{k_i} \odot \mathbf{e}_{k_i} = \mathbf{g}_{k_i} \odot (\mathbf{E} \times \mathbf{x}_{k_i}). \tag{4}$$

$$\mathbf{g}_{k_i} \in \{0, 1\}$$

Method

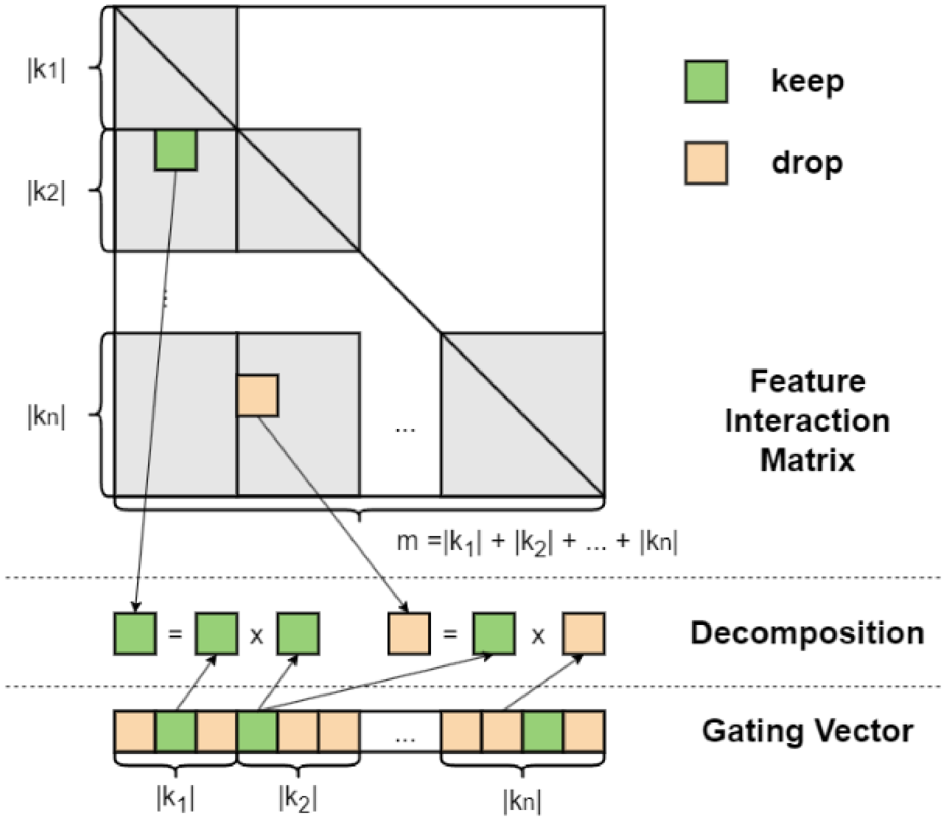


Figure 2: The Overview of OptFS.

$$\hat{y} = \mathcal{F}(\mathbf{g} \odot \mathbf{E} \times \mathbf{x} | \mathbf{W}) = \mathcal{F}(\mathbf{E}^{\mathbf{g}} \times \mathbf{x} | \mathbf{W}), \tag{5}$$

$$\mathbf{g} \in \{0, 1\}^m$$

$$\mathbf{v}_{(i,j)} = \mathcal{O}(\mathbf{e}_i, \mathbf{e}_j), \tag{6}$$

$$\hat{y} = \mathcal{H}((\mathbf{g}' \odot \mathbf{v}) \oplus \mathcal{G}(\mathbf{e}^{\mathbf{g}})) = \mathcal{H}(\mathbf{v}^{\mathbf{g}'} \oplus \mathcal{G}(\mathbf{e}^{\mathbf{g}})), \tag{7}$$

$$\mathbf{g}'_{(k_i,k_j)} = \mathbf{g}_{k_i} \times \mathbf{g}_{k_j}, \tag{8}$$

$$\hat{y} = \mathcal{H}((\mathbf{g} \times \mathbf{g} \odot \mathbf{v}) \oplus \mathcal{G}(\mathbf{g} \odot \mathbf{e})), \tag{9}$$

$$\mathbf{g} = \frac{\sigma(\mathbf{g}_c \times \tau)}{\sigma(\mathbf{g}_c^{(0)})}, \; \tau = \gamma^{t/T} \tag{10}$$

$\{2e+2, 5e+2, 1e+3, 2e+3, 5e+3, 1e+4\}$.

Method

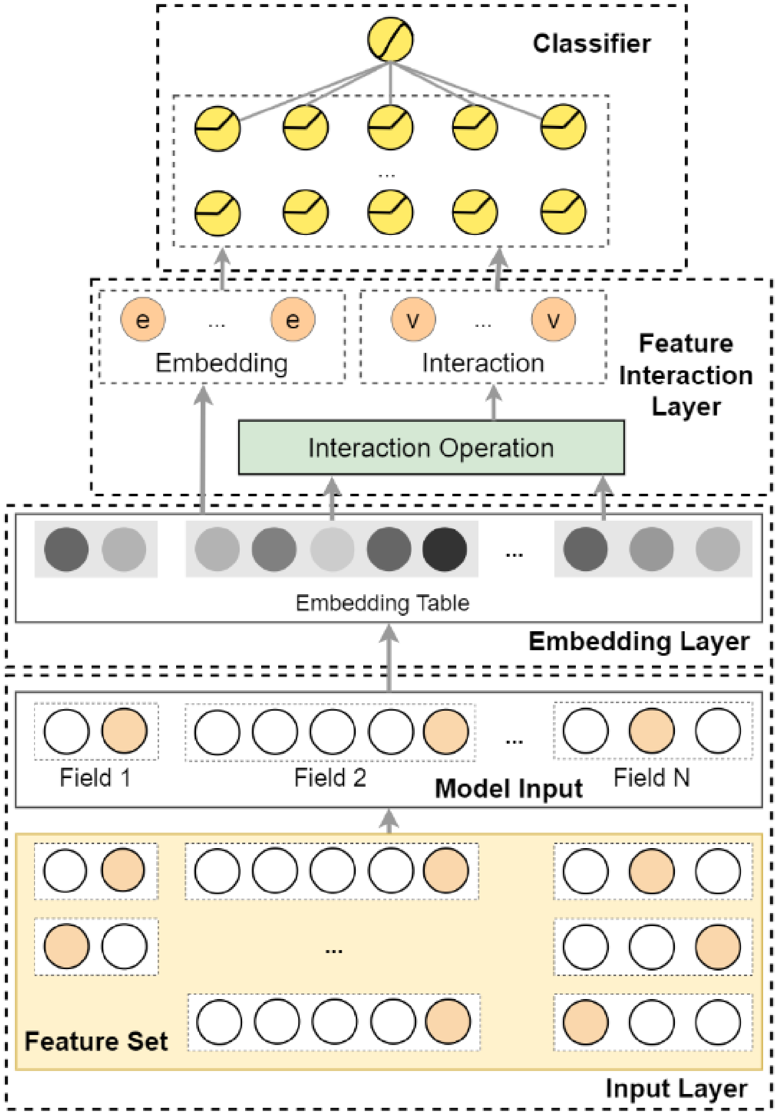


Figure 1: Overview of the general CTR framework.

$$\mathrm{CE}(y, \hat{y}) = y \log(\hat{y}) + (1 - y) \log(1 - \hat{y}), \tag{11}$$

$$\mathcal{L}_{\mathrm{CE}}(\mathcal{D}|\{\mathbf{E}, \mathbf{W}\}) = -\frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}, y) \in \mathcal{D}} \mathrm{CE}(y, \mathcal{F}(\mathbf{E} \times \mathbf{x}|\mathbf{W})), \tag{12}$$

$$\min_{\mathbf{g}_c, \mathbf{E}, \mathbf{W}} \mathcal{L}_{\mathrm{CE}}(\mathcal{D}|\{\mathbf{g}_c \odot \mathbf{E}, \mathbf{W}\}) + \lambda \|\mathbf{g}\|_1, \tag{13}$$

$$\mathbf{g} = \begin{cases} 0, & \mathbf{g}_c \leq 0 \\ 1, & \text{otherwise} \end{cases}. \tag{14}$$

$$\min_{\mathbf{E}, \mathbf{W}} \mathcal{L}_{\mathrm{CE}}(\mathcal{D}|\{\mathbf{g} \odot \mathbf{E}, \mathbf{W}\}). \tag{15}$$

Experiments

Table 1: Summary of $\mathcal{G}(\cdot)$, $\mathcal{O}(\cdot)$ and $\mathcal{H}(\cdot)$ in mainstream models

Model	$\mathcal{G}(\cdot)$	$\mathcal{O}(\cdot)$	$\mathcal{H}(\cdot)$
FM [26]	null	inner product	null
DeepFM [7]	MLP	inner product	average
DCN [31]	MLP	cross network	average
IPNN [24]	null	inner product	MLP
OPNN [24]	null	outer product	MLP
PIN [25]	null	MLP	MLP

Experiments

Table 2: Performance Comparison Between OptFS and Feature Selection Methods.

	Method	FM			DeepFM			DCN			IPNN		
		AUC↑	Logloss↓	Ratio↓	AUC↑	Logloss↓	Ratio↓	AUC↑	Logloss↓	Ratio↓	AUC↑	Logloss↓	Ratio↓
Criteo	Backbone	0.8055	0.4457	1.0000	0.8089	0.4426	1.0000	0.8107	0.4410	1.0000	0.8110	0.4407	1.0000
	LPFS	0.7888	0.4604	0.0157	0.7915	0.4579	0.2415	0.7802	0.4743	0.1177	0.7789	0.4705	0.3457
	AutoField	0.7932	0.4567	**0.0008**	0.8072	0.4439	0.3811	**0.8113**	**0.4402**	0.5900	0.8115	**0.4401**	0.9997
	AdaFS	0.7897	0.4597	1.0000	0.8005	0.4501	1.0000	0.8053	0.4472	1.0000	0.8065	0.4448	1.0000
	OptFS	**0.8060**	**0.4454**	0.1387	**0.8100***	**0.4415***	**0.0422**	0.8111	0.4405	**0.0802**	**0.8116**	**0.4401**	**0.0719**
Avazu	Backbone	0.7838	0.3788	1.0000	0.7901	0.3757	1.0000	0.7899	0.3755	1.0000	0.7913	0.3744	1.0000
	LPFS	0.7408	0.4029	0.7735	0.7635	0.3942	0.9975	0.7675	0.3889	0.9967	0.7685	0.3883	0.9967
	AutoField	0.7680	0.3862	**0.0061**	0.7870	0.3773	1.0000	0.7836	0.3782	0.9992	0.7865	0.3770	0.9992
	AdaFS	0.7596	0.3913	1.0000	0.7797	0.3837	1.0000	0.7693	0.3954	1.0000	0.7818	0.3833	1.0000
	OptFS	**0.7839**	**0.3784**	0.8096	**0.7946***	**0.3712***	**0.8686**	**0.7932***	**0.3718***	**0.8665**	**0.7950***	**0.3709***	**0.9118**
KDD12	Backbone	0.7783	0.1566	1.0000	0.7967	0.1531	1.0000	0.7974	0.1531	1.0000	0.7966	0.1532	1.0000
	LPFS	0.7725	0.1578	1.0000	0.7964	0.1532	1.0000	0.7970	**0.1530**	1.0000	0.7967	0.1532	1.0000
	AutoField	0.7411	0.1634	**0.0040**	0.7919	0.1542	0.9962	0.7943	0.1536	**0.8249**	0.7926	0.1541	0.8761
	AdaFS	0.7418	0.1644	1.0000	0.7917	0.1543	1.0000	0.7939	0.1538	1.0000	0.7936	0.1539	1.0000
	OptFS	**0.7811***	**0.1560***	0.5773	**0.7988***	**0.1527***	**0.9046**	**0.7981***	**0.1530**	0.8942	**0.7975**	**0.1530**	**0.8729**

Here $*$ denotes statistically significant improvement (measured by a two-sided t-test with p-value < 0.05) over the best baseline. **Bold** font indicates the best-performed method.

$$\text{Ratio} = \#\text{Remaining Features}/m. \qquad (16)$$

Experiments

Table 3: Performance Comparison Between OptFS and Feature Interaction Selection Method.

	Model	Method	Metrics AUC↑	Logloss↓	Ratio↓
Criteo	FM	Backbone	0.8055	0.4457	1.0000
		AutoFIS	**0.8063**	**0.4449**	1.0000
		OptFS	0.8060	0.4454	**0.1387**
	DeepFM	Backbone	0.8089	0.4426	1.0000
		AutoFIS	0.8097	0.4418	1.0000
		OptFS	**0.8100**	**0.4415**	**0.0422**
Avazu	FM	Backbone	0.7838	0.3788	1.0000
		AutoFIS	**0.7843**	0.3785	1.0000
		OptFS	0.7839	**0.3784**	**0.8096**
	DeepFM	Backbone	0.7901	0.3757	1.0000
		AutoFIS	0.7928	0.3721	1.0000
		OptFS	**0.7946***	**0.3712***	**0.8686**

Here ∗ denotes statistically significant improvement (measured by a two-sided t-test with p-value < 0.05) over the best baseline. **Bold** font indicates the best-performed method.

Table 4: Transferability Analysis on Criteo and Avazu.

	Target	Source	Metrics AUC↑	Logloss↓	Ratio↓
Criteo	DeepFM	DeepFM	**0.8100**	**0.4415**	**0.0422**
		DCN	0.8097	0.4419	0.0802
		IPNN	0.8097	0.4418	0.0719
	DCN	DCN	**0.8111**	**0.4405**	0.0802
		DeepFM	0.8106	0.4410	**0.0422**
		IPNN	0.8107	0.4410	0.0719
	IPNN	IPNN	**0.8116**	**0.4401**	0.0719
		DCN	0.8113	0.4404	0.0802
		DeepFM	0.8114	0.4403	**0.0422**
Avazu	DeepFM	DeepFM	**0.7946***	**0.3712***	0.8686
		DCN	0.7873	0.3754	**0.8665**
		IPNN	0.7872	0.3755	0.9118
	DCN	DCN	**0.7932***	**0.3718***	**0.8665**
		DeepFM	0.7879	0.3784	0.8686
		IPNN	0.7860	0.3762	0.9118
	IPNN	IPNN	**0.7950***	**0.3709***	0.9118
		DCN	0.7907	0.3747	**0.8665**
		DeepFM	0.7908	0.3748	0.8686

Here ∗ denotes statistically significant improvement (measured by a two-sided t-test with p-value < 0.05) over the best baseline. **Bold** font indicates the best-performed method.

Experiments

Table 5: Ablation Study Regarding the Re-training Stage.

	Model	Metrics	Methods: w.o.	r.i.	l.t.h.	c.i.
Criteo	DeepFM	AUC↑	0.8012	**0.8100**	**0.8100**	**0.8100**
		Logloss↓	0.4686	0.4416	**0.4415**	**0.4415**
	DCN	AUC↑	0.8077	0.8109	0.8108	**0.8111**
		Logloss↓	0.4522	0.4407	0.4408	**0.4405**
	IPNN	AUC↑	0.7757	0.8113	0.8114	**0.8116**
		Logloss↓	0.4998	0.4404	0.4403	**0.4401**
Avazu	DeepFM	AUC↑	0.6972	0.7873	0.7883	**0.7946***
		Logloss↓	0.5017	0.3754	0.3790	**0.3712***
	DCN	AUC↑	0.7122	0.7870	0.7858	**0.7932***
		Logloss↓	0.4736	0.3801	0.3764	**0.3718***
	IPNN	AUC↑	0.7560	0.7912	0.7910	**0.7950***
		Logloss↓	0.4411	0.3745	0.3745	**0.3709***

Here ∗ denotes statistically significant improvement (measured by a two-sided t-test with p-value < 0.05) over the best baseline. **Bold** font indicates the best-performed method.
Here *w.o.* stands for without re-training, *r.i.* stands for re-training with random initialization, *l.t.h.* stands for initialization using lottery ticket hypothesis [4], *c.i.* stands for re-training with customized initialization, as previously discussed in Section 2.4.

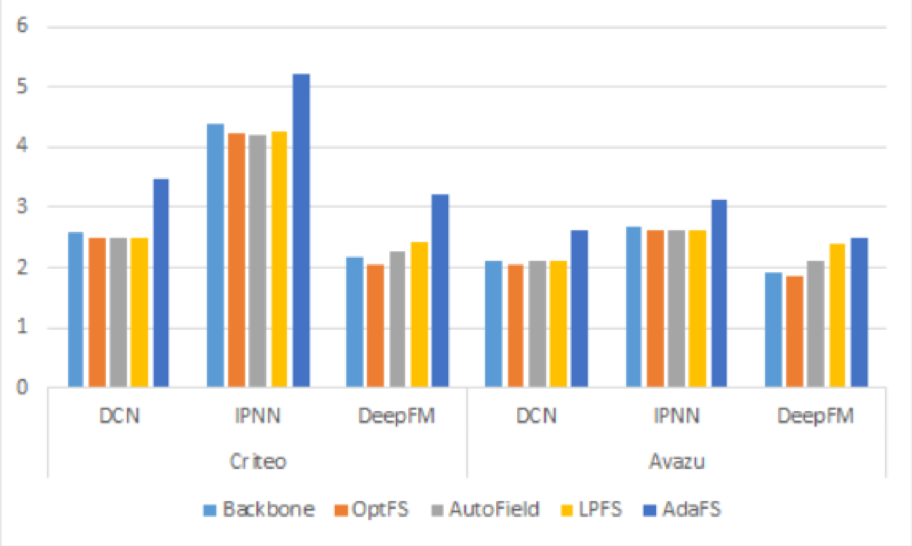
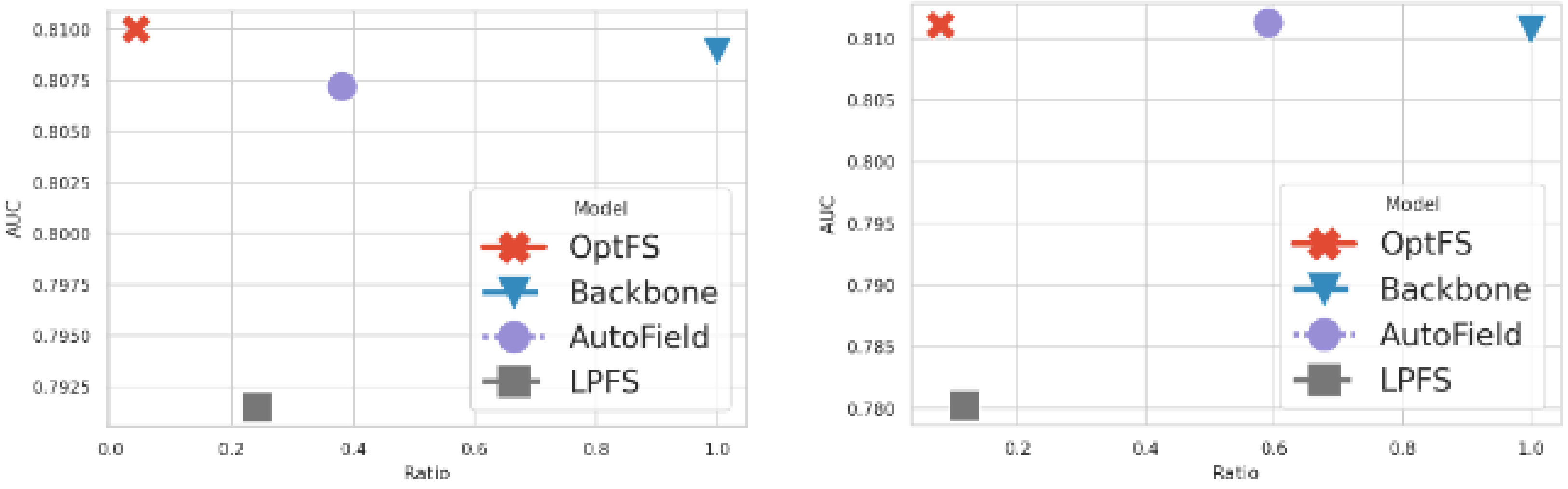


Figure 4: Inference Time of Different Models on Criteo and Avazu Dataset. The Y-axis represents the influence time, measured by ms

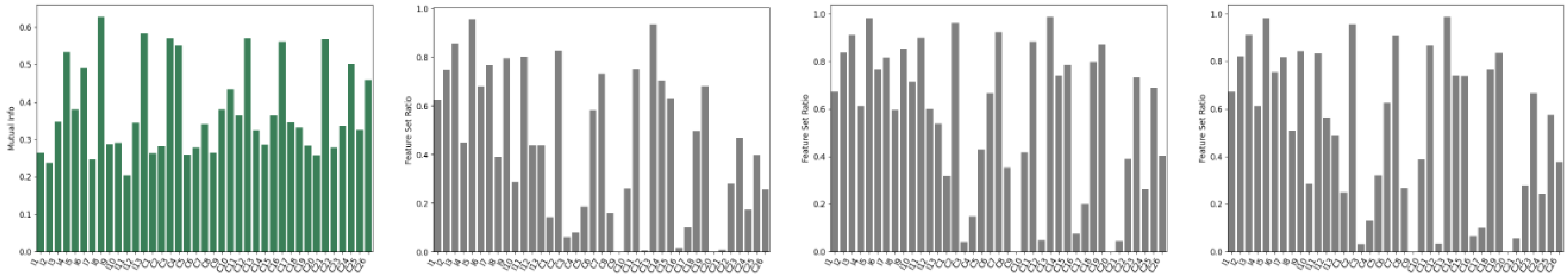
Experiments



(a) DeepFM (b) DCN

Figure 5: Visualization of efficiency-effectiveness trade-off on Criteo datasets. The closer to the top-left the better.

Experiments



(a) Mutual Info (b) DeepFM (c) DCN (d) IPNN

Figure 6: A Case Study of OptFS output on Criteo. In all subfigures, the X-axis indicates the field identifiers. Subfigure (a) plots the mutual information scores, while subfigures (b), (c) and (d) plot the feature set ratio of OptFS on DeepFM, DCN and IPNN.

Thanks