计 算 机 研 究 与 发 展
Journal of Computer Research and Development
DOI: 10.7544/issn1000-1239.202220943
61(5): 1290−1298, 2024

基于多粒度特征交叉剪枝的点击率预测模型

白 婷[1] 刘轩宁[1] 吴 斌[1] 张梓滨[2] 徐志远[2] 林康熠[2]
[1](北京邮电大学计算机学院(国家示范性软件学院) 北京 100876)
[2](微信事业群开放平台基础部 广州 510220)
(baiting@bupt.edu.cn)

Multi-Granularity Based Feature Interaction Pruning Model for CTR Prediction

Bai Ting[1], Liu Xuanning[1], Wu Bin[1], Zhang Zibin[2], Xu Zhiyuan[2], and Lin Kangyi[2]
[1](*School of Computer Science* (*National Pilot Software Engineering School*), *Beijing University of Posts and Telecommunications*, *Beijing* 100876)
[2](*Weixin Open Platform*, *Tencent*, *Guangzhou* 510220)

Abstract Learning effective high-order feature interactions is crucial for click through rate (CTR) prediction in recommender systems. Existing methods that learn meaningful high-order feature combinations by reassembling low-order feature combinations, i.e., 2-order feature interaction, suffer from high computational costs to calculate the interaction weight of all pairwise feature interactions. Some deep neural network-based methods can be seen as universal function approximators to potentially learn all kinds of feature interactions. However, it had been proved to be inefficient to approximate the low-order interactions, i.e., 2-order or 3rd-order feature interactions, which may influence the accuracy of CTR prediction task. Based on the above consideration, we propose a multi-granularity based feature interaction pruning network (FeatNet) for CTR prediction task. Firstly, FeatNet generates different subsets with a threshold pruning operation to select the meaningful feature combinations on the explicit feature granularity, which enables FeatNet to keep the diversity of different feature combinations, and reduce the complexity of high-order feature interactions. Based on the pruned feature subsets, implicit high-order feature interactions are further conducted on the granularity of feature elements, which automatically filters out the invalid feature interactions. Extensive experiments are conducted on two real-world datasets, showing the superiority of FeatNet in CTR prediction.

Key words CTR prediction; high-order feature interaction; multi-granularity; feature pruning; feature denoising

摘 要 在推荐系统中,学习有效的高阶特征交互是提升点击率预测的关键.现有的研究将低阶特征进行组合来学习高阶交叉特征表示,导致模型的时间复杂度随着特征维度的增加呈指数型增长;而基于深度神经网络的高阶特征交叉模型也无法很好地拟合低阶特征交叉,影响预测的准确率.针对这些问题,提出了基于多粒度特征交叉剪枝的点击率预测模型 FeatNet.该模型首先在显式的特征粒度上,通过特征剪枝生成有效的特征集合,保持了不同特征组合的多样性,也降低了高阶特征交叉的复杂度;基于剪枝后的特征集合,在特征元素粒度上进一步进行隐式高阶特征交叉,通过滤波器自动过滤无效的特征交叉.在 2 个真实的数据集上进行了大量的实验,FeatNet 都取得了最优的点击率预测效果.

收稿日期: 2022-11-11; **修回日期**: 2023-03-09
基金项目: 国家自然科学基金项目(62102038, 61972047);腾讯微信开放平台项目(S2021120)
This work was supported by the National Natural Science Foundation of China (62102038, 61972047) and the Project of Tencent Weixin Open Platform (S2021120).
通信作者: 吴斌(wubin@bupt.edu.cn)

关键词 点击率预测；高阶特征交叉；多粒度；特征剪枝；特征降噪

中图法分类号 TP391

点击率(click through rate, CTR)预测是推荐系统中的一个重要任务[1-4]，它的目标是预测用户对候选物品的点击概率，并根据预测概率对用户进行物品的个性化推荐. 特征交叉是CTR预测任务中非常重要的一环，学习有效的高阶特征交叉可以提高模型的性能[5-8]. 目前学习高阶交叉特征的方法主要基于低阶特征组合，通过不同的组合方式构造高阶的特征交叉. 但当原始特征的数量增加时，特征组合的数量会呈指数增长，带来巨大的时间开销. 而在实际应用中，原始输入特征往往是上万维的稀疏特征：例如用户或产品的ID字段，在进行one-hot编码后会产生一个非常稀疏的向量，基于高维稀疏向量的特征交叉会带来极大的开销，而且不可避免地会导致模型过拟合问题.

为了解决上述问题，基于特征交叉的研究工作分为3类：1)基于因子分解机(factorization machines, FM)的模型，例如FM[9]，DeepFM[10]，xDeepFM[11]，FiBiNet[12]；2)基于自注意力(self-attention)机制的模型，例如AutoInt[13]，AFM[14]，HoAFM[15]；3)基于神经网络的特征交叉模型，例如DCN[16]及其改进版本DCN-V2[17]，这类方法通过组合低阶特征交叉的方式来产生高阶特征交叉. 考虑到枚举所有的2阶交叉特征非常耗时，并且可能产生不相关的特征组合，使得模型引入噪声影响模型性能. 一些方法[10,18]尝试利用深度神经网络(deep neural network, DNN)来减小低阶特征组合的搜索空间，但其被证明不能很好地学习低阶的特征交叉信息[1,16]，影响模型的预测性能.

为了能够同时学习有效的低阶和高阶的特征交叉，本文提出了一种基于多粒度特征交叉剪枝的模型FeatNet(feature interaction pruning network). 该模型包括显式特征子集生成网络(feature subset generation network, FSGN)和隐式特征交叉滤波网络(feature interaction filtering network, FIFN)这2个部分. 其中，显式特征子集生成网络通过软阈值剪枝策略生成有效的特征集合，既保持了不同特征组合的多样性，又降低了高阶特征交叉的复杂度；隐式特征交叉滤波网络是基于剪枝后的特征集合，在特征元素粒度上进一步进行隐式高阶特征交叉，通过滤波器自动过滤无效的特征交叉信息，提升了模型性能.

相比于其他高阶特征交叉模型，在本文提出的基于多粒度特征交叉剪枝的CTR预测模型FeatNet中，高阶特征交叉的生成来源于不同组合的低阶特征集合，在显式特征粒度筛除了噪声特征，为后续的隐式高阶特征交叉提供了更加丰富的特征组合信息，提高了模型的预测性能. 本文的贡献总结为3个方面：

1)提出了一种基于多粒度特征交叉剪枝的CTR预测模型FeatNet，该模型能够在显式的特征粒度和隐式的特征交叉粒度过滤噪声交叉信息，同时保留有效低阶和高阶特征交叉信息.

2)提出了一种基于软阈值特征剪枝的策略，能够自动生成带有不同信息的特征集合，并对不同剪枝阈值进行分析，验证了降低子集之间的相似度、生成差异性的特征组合，可以提升模型CTR预测的准确率；

3)在MovieLens和WeChat数据集上进行了大量的实验，验证FeatNet模型能够很好地处理高维输入特征，取得最优的点击率预测效果.

1 相关工作

CTR预测的研究[19-23]通常分为2种：一种主流算法侧重于如何捕获高阶特征交互；另一种则试图找到更好的方法来学习特征交互的重要性，以获得更好的特征组合.

本文将从高阶特征交叉建模、特征重要性建模2个方面来梳理点击率预测的相关研究，并总结常用的剪枝策略.

1.1 高阶特征交叉建模

FM[9]是一种经典的特征交叉方法，它枚举了所有2阶特征组合，通过点积的方式来产生2阶交叉特征. FM模型在构建K阶交叉特征时，需要枚举所有K阶特征的全部组合，会产生非常高的开销. Blondel等人[24]提出了一种高阶特征交叉的方法HOFMs，通过引入核函数，构建了线性时间复杂度的动态规划算法，可以加快高阶特征交叉的速度. DeepFM[10]采用把FM和DNN相结合的方式，使用FM构造2阶交叉特征，利用深度神经网络构造隐式高阶交叉特征，减小了计算高阶交叉特征时的开销. xDeepFM[11]在DeepFM的基础上进行了改进，增加了压缩交叉网络(compressed interaction network, CIN)来显式枚举构造特征交叉，并利用求和池化操作对特征矩阵进行压缩来降低特征维度. 虽然CIN可以降低高阶交叉

特征的时间开销，但是仍需要对所有特征进行两两组合. 为了能够让模型自动地学习特征的组合方式，DCN[16]及其改进版本 DCN-V2[17]设计了 Cross Net 使用跨层信息来构造交叉特征，在学习隐式特征交叉方面取得更好的性能. AutoInt[13]模型基于 Transformer[25]结构，利用多头自注意力模块来构建隐式高阶特征交叉，采用 Query 和 Key 来评估不同特征维度之间的相似度，然后将带有注意力权重的特征相加得到高阶特征交叉信息.

1.2 特征权重建模

在 CTR 预测任务中，除了有效建模高阶特征交叉，建模交叉特征的重要性权重也非常重要，可以用来去除无关的噪声信息，提升预测的准确率. 例如，FwFM[26]通过为每个交叉对分配可学习的权重来改进 FM；IAFM[27]考虑特征和域信息来建模不同粒度的特征交叉权重；FiBiNet[28]使用 Squeeze-Excitation 网络通过双线性层来构建特征交叉并学习对应的权重值；AutoFIS[29]通过使用正则优化器来优化参数搜索空间，能够利用较少的资源自动识别重要的交叉特征.

1.3 剪枝策略

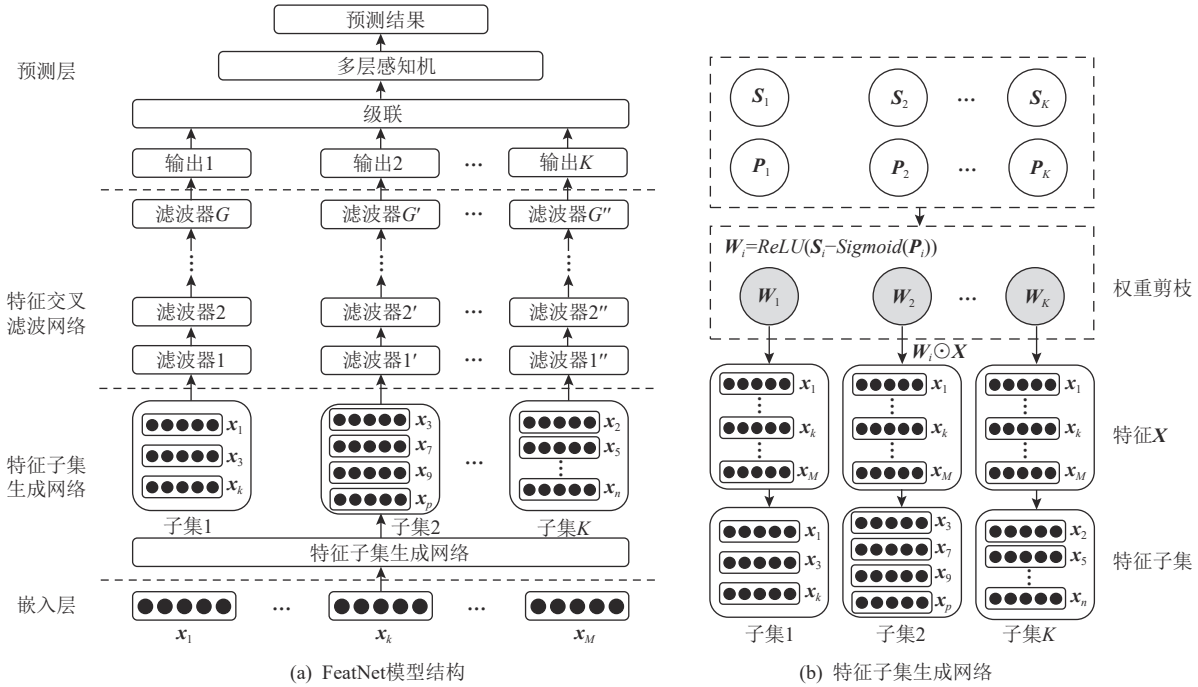
为了生成带有不同信息的特征子集，本文采用软阈值剪枝策略[30-32]来对特征进行选择. 目前，特征剪枝方法包括：Louizos 等人[33]提出在神经网络中加入L_0范数正则化，通过训练参数权重，并根据权重值来判定是否对网络进行剪枝. L_0-SIGN[34]使用 DNN 不同交叉特征的重要性，并采用L_0范数正则化方法来产生有意义的交叉特征；Kusupati 等人[30]提出了一种更简单的软阈值方法 STR，该方法从稀疏数据中学习，并使用软阈值权重来更新学习到的参数，当神经元的权重小于阈值时，它的权重被置为 0. 该方法可以在保持高准确率的同时减少 95% 以上的嵌入参数[35].

2 模型的提出

本文提出了基于多粒度特征交叉剪枝的点击率预测模型 FeatNet. 该模型首先基于显式特征粒度，通过特征剪枝生成有效的特征集合，保持了不同特征组合的多样性，也降低了高阶特征交叉的复杂度；基于剪枝后的特征集合，通过滤波器在特征元素粒度上进一步进行隐式高阶特征交叉.

2.1 模型结构

如图 1(a)所示，FeatNet 由 4 部分组成：嵌入层、特征子集生成网络、特征交叉滤波网络和预测层. 原始特征在嵌入层转化为稠密向量后，特征子集生成网络在特征级别(feature-wise)对特征进行显式地选择，得到若干个特征子集，并保证每个子集中特征组合的多样性；特征交叉滤波网络对每个子集中的特征进行滤波降噪，并在元素级别(bit-wise)对特征进行隐式特征交叉；预测层输出最终预测结果.



(a) FeatNet模型结构 (b) 特征子集生成网络

Fig. 1 Illustration of FeatNet and FSGN

图 1 FeatNet 和特征子集生成网络示意图

2.2 嵌入层

嵌入层将原始的输入特征转化为稠密向量. 一般来说, 输入的特征可以分为 3 类: 数值类型、类别特征和向量. 数值类型数据, 例如年龄, 可以直接用数字来表示; 类别特征数据, 例如国籍、ID 等, 需要转化为 one-hot 编码, 当类别特征较多时, 产生的 one-hot 编码是一个非常稀疏的向量; 向量数据一般是由上游任务产生的, 例如某个用户的浏览序列数据或者多模态数据(如图像的视觉特征向量)等. 模型通过嵌入层把所有的原始特征都映射到低维的空间中, 并把所有低维向量进行拼接得到最终的低维稠密特征向量的表示.

记 M 为稠密向量的个数, $\boldsymbol{x}_i$ 为第 i 个稠密向量, 则最终的输入矩阵 $\boldsymbol{X}$ 可以表示为

$$\boldsymbol{X}=(\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_M). \tag{1}$$

2.3 特征子集生成网络

特征子集生成网络能够在特征层面进行显式的特征自动选择, 产生若干个不同的特征子集. 每一个特征子集为特征交叉提供了不同的搜索空间, 在每个子集中分别进行高阶特征交叉, 不仅减少了高阶特征交叉的搜索空间, 还能使得模型能够学习到带有不同高阶信息的特征交叉组合. 通过基于软阈值的特征剪枝策略, 每个子集中的特征种类和数目保持差异化, 使得特征交叉滤波网络能够从信息更加多样的特征组合中学习丰富的交叉特征, 从而提高模型的表达能力. 基于软阈值剪枝策略的特征子集生成网络结构如图 1(b)所示.

假设特征子集生成网络有 K 个子集, 每个子集都有一组权重参数 set param 和剪枝参数 pruning param, 分别记为 $\boldsymbol{S}$ 和 $\boldsymbol{P}$. 权重参数 $\boldsymbol{S}$ 决定了在该子集中某个特征的重要程度, 剪枝参数 $\boldsymbol{P}$ 决定了在该子集中某个特征是否应该保留. 通过软阈值剪枝操作, 2 组参数共同决定了这个子集中每个特征剪枝后的特征权重值 $\boldsymbol{W}$:

$$\boldsymbol{W}=ReLU(\boldsymbol{S}-Sigmoid(\boldsymbol{P})), \tag{2}$$

其中, 激活函数 $Sigmoid$ 保证剪枝参数 $\boldsymbol{P}$ 都分布在 (0, 1) 之间, 且剪枝参数不会太大. 如果剪枝参数太大, 就会导致过多的特征被去除, 影响模型的性能; 而太小的剪枝参数, 又会使得剪枝力度过小, 过滤不掉噪声特征. 对权重参数 $\boldsymbol{S}$ 进行 [0, 1] 区间内的随机初始化, 并通过改变剪枝参数 $\boldsymbol{P}$ 的初始化范围来观察各个子集之间的差异性变化. 用权重参数 $\boldsymbol{S}$ 减去激活后的剪枝参数 $\boldsymbol{P}$ 得到剪枝权重, 并通过激活函数 $ReLU$ 把剪枝权重小于 0 的值置为 0, 得到最终的特征权重 $\boldsymbol{W}$.

最后, 把特征权重 $\boldsymbol{W}$ 和输入的稠密特征 $\boldsymbol{X}$ 相乘, 权重为 0 的特征就从集合中被自动过滤, 权重不为 0 的特征以加权的方式保留在集合中, 生成 K 个具有不同特征组合的特征子集 $\boldsymbol{X}'$, 计算公式为:

$$\boldsymbol{X}_i'=\boldsymbol{W}_i\odot\boldsymbol{X}, \tag{3}$$

$$\boldsymbol{X}'=(\boldsymbol{X}_1',\boldsymbol{X}_2',\cdots,\boldsymbol{X}_K'). \tag{4}$$

其中⊙表示哈达玛积.

2.4 特征交叉滤波网络

考虑到在序列推荐工作[36]中, 滤波器能够有效为原始数据进行滤波降噪, 学习数据的序列关联信息, 我们将滤波器和隐式特征交叉结合起来, 提出了一个结构简单且有效的特征交叉滤波网络来捕捉高阶隐式特征交叉. 特征交叉滤波网络基于特征子集生成网络的输出, 把特征子集输入到多个滤波器(filter)的串行结构中, 对特征进行滤波降噪, 并在元素级别对特征进行隐式高阶交叉. 滤波器的结构如图 2 所示.

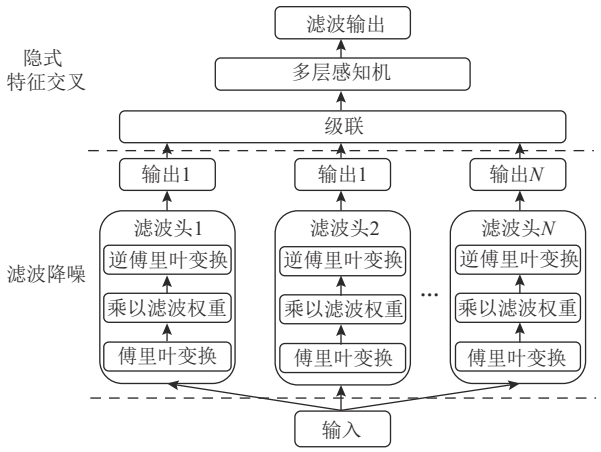


Fig. 2 Illustration of filter

图 2 滤波器示意图

借鉴序列推荐工作[36]中滤波网络的模型结构, 在滤波交叉网络中, 输入的特征子集被分别送到多个滤波头中, 对于每一个特征子集 $\boldsymbol{X}_i'$, 滤波头通过快速傅里叶变换把特征转换到频率域中, 利用可学习的滤波权重和其相乘进行滤波降噪, 再通过逆快速傅里叶变换把特征转换回原始的分布空间以完成滤波操作, 得到滤波后的特征子集 $\boldsymbol{X}_i''$. 记傅里叶变换为 $F(\cdot)$, 逆傅里叶变换为 $IF(\cdot)$, 第 i 个滤波头的滤波权重为 $\boldsymbol{w}_i$, 则第 i 个滤波头将 $\boldsymbol{X}_i'$ 转化为 $\boldsymbol{X}_i''$ 的方式为

$$\boldsymbol{X}_i''=IF(F(\boldsymbol{X}_i')\odot\boldsymbol{w}_i). \tag{5}$$

最后将多个滤波头的输出进行拼接, 输入到多层感知机中进行隐式特征交叉, 得到交叉后的高阶

特征向量. 多个滤波器的串行叠加，保证了模型能够学习到更丰富的高阶特征交叉信息.

2.5 预测层

预测层把特征交叉滤波网络输出的所有交叉特征进行拼接，得到特征的最终表示 $\boldsymbol{X}''=(\boldsymbol{X}_1'',\boldsymbol{X}_2'',\cdots,\boldsymbol{X}_K'')$. 然后把 $\boldsymbol{X}''$ 输入到一个多层感知机中，输出预测结果. 记初始状态为 $\boldsymbol{a}^{(0)}=\boldsymbol{X}''$，点击率预测概率 $\hat{y}$ 为

$$\hat{y}=Sigmoid(\boldsymbol{H}^{(l)}\boldsymbol{a}^{(l-1)}+\boldsymbol{b}^{(l)}), \tag{6}$$

其中 $\boldsymbol{H}$ 和 $\boldsymbol{b}$ 为转换矩阵的权重和偏置，l 为多层感知机的层数.

2.6 损失函数

在 CTR 预测任务中，最常使用 LogLoss 损失函数，它度量了样本真实标签与预测标签分布的 KL 散度：

$$E=-\frac{1}{N}\sum_{i=1}^{N}(y_i\times\ln(\hat{y}_i)+(1-y_i)\times(1-\ln(1-\hat{y}_i))), \tag{7}$$

其中 y_i 是样本 x_i 的真实标签值，$\hat{y}_i$ 是模型预测的标签，E 是数据集中每个样本的平均 LogLoss 损失.

3 实　　验

为了验证模型的有效性，本文选取了 CTR 预测模型中的 4 个代表性模型，在 2 个真实数据集上进行试验，通过 AUC(area under curve)和 LogLoss 来评估模型的推荐效果. AUC 值可以评估随机抽样时正样本排在负样本前的概率[37]，LogLoss 能够度量样本的真实标签与预测标签的分布距离，这 2 个指标在推荐场景中都有重要的意义.

3.1 实验设置

3.1.1 数据集

本文在 2 个数据集上进行实验，包括公开数据集 MovieLens，以及一个来自微信的企业数据集 WeChat. 如表 1 所示.

Table 1 Dataset Statistics

表 1 数据集统计信息

数据集	特征数	数据量
MovieLens	12	1 000 000
WeChat	261	10 000 000

MovieLens 数据集是在个性化推荐中常用的数据集，其中包含了电影的题材、标题以及用户的性别、年龄、职业和对电影的评分等信息；WeChat 数据集中以字典的方式存放了用户和推文的特征，特征的 ID 索引值为 190 万，每个样本取其中 261 个作为样本特征，数据非常稀疏.

3.1.2 对比方法

实验选取 CTR 预测模型中 4 个代表性模型 DeepFM, xDeepFM, AutoInt, DCN-V2 作为对比.

1)DeepFM[9] 用因子分解机学习低阶交叉特征，并通过深度神经网络产生高阶交叉特征；

2)xDeepFM[10] 在 DeepFM 基础上新增了 CIN 这一新的模块，实现了特征级别的显式交叉；

3)AutoInt[12] 采用 Query 和 Key 来评估不同特征维度之间的相似度，然后将带有注意力权重的特征相加得到高阶特征交叉信息；

4)DCN-V2[16] 是基于深度神经网络的模型，采用 CrossNet 来学习跨层特征交叉信息，能够较好地捕捉隐式特征交叉.

3.1.3 参数设置

在实验中，数据集以 8∶1∶1 的比例划分为训练集、验证集和测试集. 采用网格搜索的方式来调整各个模型中的超参数并记录不同超参数下各个模型的最优结果. FeatNet 的子集数 K 在 [2, 12] 之间搜索，滤波头个数 F 在 [1, 5] 之间搜索，学习率 lr 设置在 {1E−4, 1E−5, 5E−6, 1E−6}搜索. 所有实验均在 Python 3.6.13, Pytorch 1.10.1 环境下完成.

调参实验后，FeatNet 在 MovieLens 数据集上的最优配置为：学习率 lr =1E−4，滤波头个数 F=4，子集数 K=5，权重参数 $\boldsymbol{S}$ 的初始化区间为 [0.2,1]，剪枝参数 $\boldsymbol{P}$ 的初始化区间为 [−4, 1]. 在 WeChat 数据集上的最优配置为：学习率 lr =1E−6，滤波头个数 F=[2,2](2 个滤波器串行叠加，每个滤波器有 2 个滤波头)，子集数 K=10，权重参数 $\boldsymbol{S}$ 的初始化区间为 [0.4, 1]，剪枝参数 $\boldsymbol{P}$ 的初始化区间为 [−4, −2].

3.2 实验结果及分析

模型结果如表 2 所示，可以得出 4 个结论：

1)DeepFM 和 xDeepFM 在 2 个数据集上的表现

Table 2 Comparative Results on MovieLens and WeChat Datasets

表 2 在 MovieLens 和 WeChat 数据集上的对比结果

模型	MovieLens		WeChat	
	AUC	LogLoss	AUC	LogLoss
DeepFM	0.888 4	0.331 03	0.657 1	0.273 73
xDeepFM	0.888 0	0.332 52	0.657 1	0.273 80
AutoInt	0.891 1	0.330 71	0.658 5	0.273 81
DCN-V2	0.892 0	0.329 48	0.657 8	0.274 44
FeatNet（本文方法）	**0.892 3**	**0.326 13**	**0.665 1**	**0.272 90**

注：已有工作证明，在 CTR 预测中，AUC 在 0.000 1 级别的提升为显著的提升. 加粗和下划线数字分别表示最优结果、次优结果.

最差，说明这 2 个模型构造低阶和高阶特征交叉的能力较弱，不能把 2 阶噪声特征（把 2 个不相关的特征组合到一起）去除，在高维稀疏数据集上可能会拟合噪声数据，影响预测结果，因此这 2 个模型在 2 个数据集上效果较差.

2）AutoInt 模型在 WeChat 数据集上的效果仅次于 FeatNet，表明 self-attention 机制能够很好地捕捉高阶交叉特征的信息. 但是在 MovieLens 数据集上的效果一般，原因可能是 self-attention 机制对于低阶特征不能有效地学习，限制了模型的建模能力.

3）DCN-V2 在 MovieLens 数据集上的 AUC 值仅次于 FeatNet，说明 DCN-V2 组合低阶特征的能力强，能够产生有意义的高阶特征交叉. 但是 DCN-V2 在 WeChat 数据集上的表现一般，原因是在处理高维稀疏向量时，模型没有对输入特征进行降噪，会拟合噪声数据，导致模型效果下降.

4）FeatNet 模型在 2 个数据集上都取得了最高的 AUC 值和最低的 LogLoss，说明模型通过软阈值剪枝，能够对原始的稀疏特征信息进行自动降噪，为隐式交叉网络提供了有效的输入特征，使得模型能够建模有效的高阶特征交叉信息.

3.3 消融实验

为了验证本文提出的特征子集生成网络和特征交叉滤波网络的有效性，本文基于 MovieLens 上的最优模型进行了消融实验，设置了仅保留特征子集生成网络和仅保留特征交叉滤波网络 2 组实验来观察 2 个模块对模型整体结果的影响，其中在使用特征子集生成网络的实验中，保留了特征交叉滤波网络中的特征交叉部分，实验结果如表 3 所示.

Table 3 Ablation Experiment Results

表 3 消融实验结果

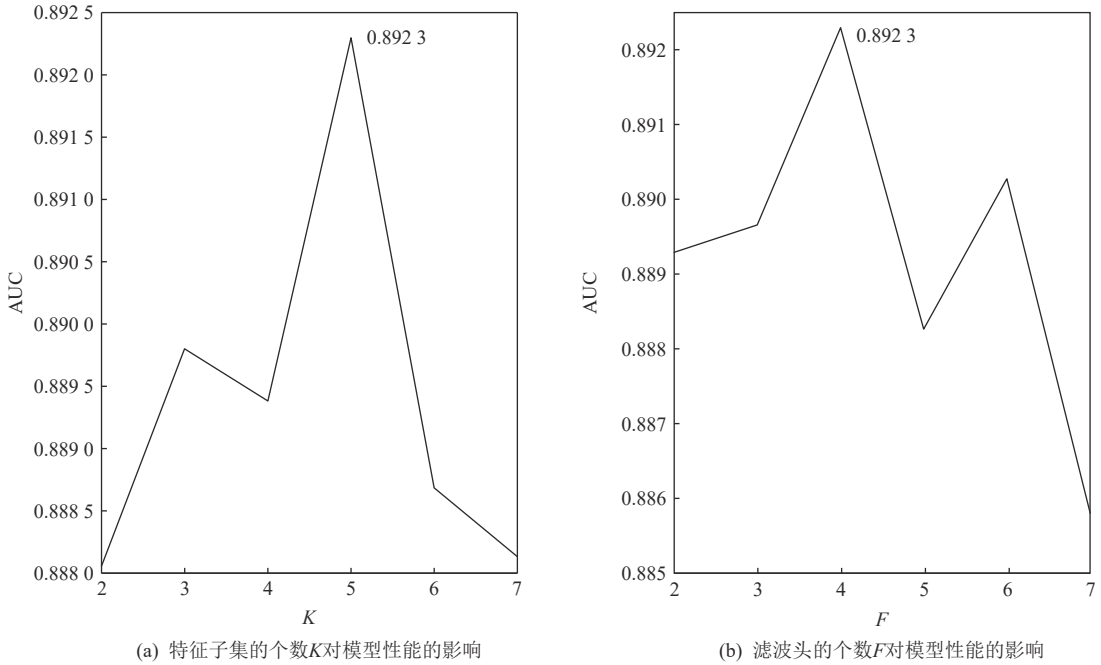
特征子集生成网络	特征交叉滤波网络	AUC
	√	0.889 5
√		0.891 1
√	√	0.892 3

注：“√”表示保留该网络.

由表 3 可以看出，仅有特征交叉滤波网络的对照组 AUC 值大幅度下降，说明如果在特征交叉前不进行剪枝处理，那么在特征交叉时会因为搜索空间太大且单一导致模型不能学习到带有不同高阶信息的特征交叉组合，限制了模型的表达能力. 使用特征子集生成网络的模型 AUC 值也有一定程度的下降，说明没有特征交叉滤波网络，特征中存在的噪声不能有效地被去除，进而影响模型的预测效果. 与都保留这 2 种网络相比，2 组实验的 AUC 值都有所下降，验证了特征子集生成网络和特征交叉滤波网络的有效性.

3.4 模型超参数分析

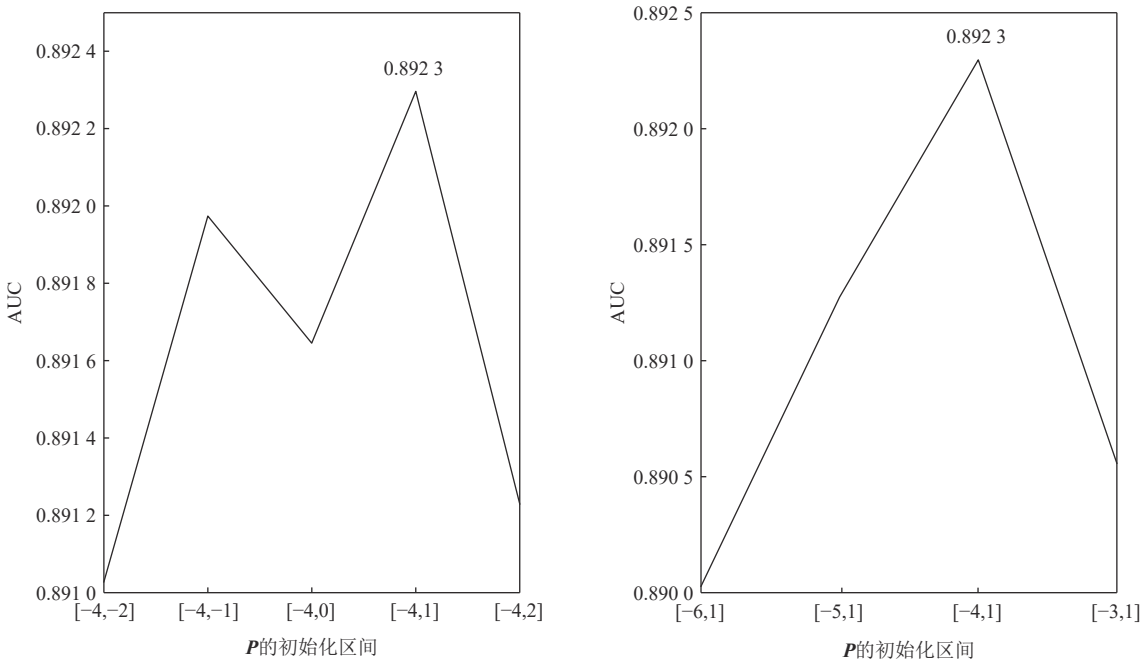
为了探究超参数对模型效果的影响，本文选取 MovieLens 上最优模型配置，进行了模型超参数分析实验，选取特征子集生成网络中特征子集生成个数 K、特征交叉滤波网络中滤波头的个数 F 和剪枝参数 $\boldsymbol{P}$ 进行分析. 图 3 和图 4 为 MovieLens 数据集上，超参



(a) 特征子集的个数K对模型性能的影响　　(b) 滤波头的个数F对模型性能的影响

Fig. 3 Effect of K and F on model performance

图 3 K 和 F 对模型性能的影响



(a) 固定区间左端点时$\boldsymbol{P}$对模型性能的影响 (b) 固定区间右端点时$\boldsymbol{P}$对模型性能的影响

Fig. 4 Effect of $\boldsymbol{P}$ on model performance

图 4 $\boldsymbol{P}$ 对模型性能的影响

数 K, F, $\boldsymbol{P}$ 对于模型 AUC 值的影响. 可以看出:

1)如图 3(a), 随着子集个数 K 的增加, 模型 AUC 值呈现先增大后减小的趋势. 当子集个数 K=5 时, 模型取得最优效果. 产生这种现象的原因在于: 子集中带有不同的特征组合信息, 随着 K 的增大, 特征交叉滤波网络可以学习到更加多样的特征交叉信息, 因此模型的 AUC 值提高. 但是随着 K 的继续增大, 学习的参数空间也会增加, 加大了模型拟合的难度, 导致模型的 AUC 值降低.

2)如图 3(b), 随着滤波头个数 F 的增长, 模型 AUC 值也呈现先增大后减小的趋势. 当滤波头个数 F=4 时, 模型取得最优效果. 产生这种现象是因为随着滤波头个数的增加, 滤波后拼接得到的向量长度会成倍增长. 向量长度过长, 就会导致模型训练参数增加, 拟合速度变慢, 同时加大了拟合难度; 向量长度过短, 就限制了模型的表达能力, 不能产生有效的高阶交叉特征, 模型 AUC 值降低.

3)图 4(a)是固定剪枝参数 $\boldsymbol{P}$ 的区间左端点时, 模型 AUC 值随右端点变化的情况; 图 4(b)是固定右端点时, 模型 AUC 值随左端点变化的情况. 可以看出, 模型 AUC 值随另一个端点的变化呈先增大后减小的趋势.剪枝参数 $\boldsymbol{P}$ 的初始化范围是 [−4, 1] 时, 模型取得最优效果. 当端点取值较小时, 剪枝后每个子集都保留了大部分特征信息, 在特征交叉时能够得到的信息增多, 模型的建模能力增强, 模型 AUC 值提高. 但是随着端点取值的增大, 剪枝后的权重 $\boldsymbol{W}$ 会不断趋近于 $\boldsymbol{0}$, 导致每个子集中只能保留少量特征, 模型中的特征信息量减少, 模型的建模能力受到了限制, 模型 AUC 值降低.

4 总 结

本文从多粒度剪枝的角度出发, 设计了一个点击率预测模型 FeatNet. FeatNet 能够在特征级别进行特征选择, 对不相关的特征进行剪枝, 产生多个特征子集, 并保证各个子集之间的差异性; 该模型还能够在元素级别对特征进行隐式交叉, 构造有效的高阶交叉特征. 目前工作中剪枝策略的阈值是通过超参数进行搜索, 未来工作中将探索阈值自动学习方法, 进一步提升点击率预测的效果.

作者贡献声明: 白婷提出了算法思路、实验方案并修改论文; 刘轩宁完成实验并撰写论文; 吴斌提出指导意见并修改论文; 张梓滨、徐志远和林康熠提供了微信数据集并对模型构建和写作提出了指导意见.

参 考 文 献

[1] Beutel A, Covington P, Jain S, et al. Latent cross: Making use of context in recurrent recommender systems[C]//Proc of the 11th ACM

Int Conf on Web Search and Data Mining. New York: ACM, 2018: 46−54

[2] Broder A Z. Computational advertising and recommender systems[C] //Proc of the 2nd ACM Conf on Recommender Systems. New York: ACM, 2008: 1−2

[3] Cao Zhe, Qin Tao, Liu Tieyan, et al. Learning to rank: From pairwise approach to listwise approach[C] //Proc of the 24th Int Conf on Machine Learning. New York: ACM, 2007: 129−136

[4] Wang Yiting, Lan Yanyan, Pang Liang, et al. Unbiased learning to rank based on relevance correction[J]. Journal of Computer Research and Development, 2022, 59(12): 2867−2877 (in Chinese)
(王奕婷, 兰艳艳, 庞亮, 等. 基于相关修正的无偏排序学习方法[J]. 机研究与发展, 2022, 59(12): 2867−2877)

[5] He Xiangnan, Chua T S. Neural factorization machines for sparse predictive analytics[C] //Proc of the 40th Int ACM SIGIR Conf on Research and Development in Information Retrieval. New York: ACM, 2017: 355−364

[6] Li Zeyu, Cheng Wei, Chen Yang, et al. Interpretable click-through rate prediction through hierarchical attention[C] //Proc of the 13th Int Conf on Web Search and Data Mining. New York: ACM, 2020: 313−321

[7] Chen Ting, Lin Ji, Lin Tian, et al. Adaptive mixture of low-rank factorizations for compact neural modeling[C/OL] // Proc of the 7th Int Conf on Learning Representations. 2019[2022-11-10].https://openreview.net/forum?id=r1xFE3Rqt7

[8] Shan Ying, Hoens T R, Jiao Jian, et al. Deep crossing: Web-scale modeling without manually crafted combinatorial features[C] //Proc of the 22nd ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2016: 255−262

[9] Rendle S. Factorization machines[C] //Proc of the 10th IEEE Int Conf on Data Mining. Piscataway, NJ: IEEE, 2010: 995−1000

[10] Guo Huifeng, Tang Ruiming, Ye Yunming, et al. DeepFM: A factorization-machine based neural network for CTR prediction[C] //Proc of the 26th Int Joint Conf on Artificial Intelligence. New York: Curran Associates, 2017: 1725−1731

[11] Lian Jianxun, Zhou Xiaohuan, Zhang Fuzheng, et al. xDeepFM: Combining explicit and implicit feature interactions for recommender systems[C] //Proc of the 24th ACM SIGKDD Int Conf on Knowledge Discovery & Data Mining. New York: ACM, 2018: 1754−1763

[12] Huang Tongwen, Zhang Zhiqi, Zhang Junlin. FiBiNET: Combining feature importance and bilinear feature interaction for click-through rate prediction[C] //Proc of the 13th ACM Conf on Recommender Systems. New York: ACM, 2019: 169−177

[13] Song Weiping, Shi Chence, Xiao Zhiping, et al. AutoInt: Automatic feature interaction learning via self-attentive neural networks[C] //Proc of the 28th ACM Int Conf on Information and Knowledge Management. New York: ACM, 2019: 1161−1170

[14] Cheng Weiyu, Shen Yanyan, Huang Linpeng. Adaptive factorization network: Learning adaptive-order feature interactions[C] //Proc of the 34th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2020: 3609−3616

[15] Tao Zhulin, Wang Xiang, He Xiangnan, et al. HoAFM: A high-order attentive factorization machine for CTR prediction[J]. Information Processing & Management, 2020, 57(6): 102076

[16] Wang Ruoxi, Fu Bin, Fu Gang, et al. Deep & cross network for ad click predictions[C] // Proc of the 7th Workshop on Data Mining for Online Advertising. New York: ACM, 2017: 12: 1−12: 7

[17] Wang Rouxi, Shivanna R, Cheng Zhiyuan, et al. DCN V2: Improved deep & cross network and practical lessons for web-scale learning to rank systems[C] //Proc of the 30th Web Conf. New York: ACM, 2021: 1785−1797

[18] Cheng H T, Koc L, Harmsen J, et al. Wide & Deep learning for recommender systems[C] //Proc of the 1st Workshop on Deep Learning for Recommender Systems. New York: ACM, 2016: 7−10

[19] Chapelle O, Manavoglu E, Rosales R. Simple and scalable response prediction for display advertising[J]. ACM Transactions on Intelligent Systems and Technology, 2014, 5(4): 1−34

[20] Deng Xiaotie, Goldberg P, Sun Yang, et al. Pricing ad slots with consecutive multi-unit demand[J]. Autonomous Agents and Multi-Agent Systems, 2017, 31(3): 584−605

[21] McMahan H B, Holt G, Sculley D, et al. Ad click prediction: A view from the trenches[C] //Proc of the 19th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2013: 1222−1230

[22] Richardson M, Dominowska E, Ragno R. Predicting clicks: Estimating the click-through rate for new ads[C] //Proc of the 16th Int Conf on World Wide Web. New York: ACM, 2007: 521−530

[23] Sun Yang, Zhou Yunhong, Yin Ming, et al. On the convergence and robustness of reserve pricing in keyword auctions[C] //Proc of the 14th Annual Int Conf on Electronic Commerce. New York: ACM, 2012: 113−120

[24] Blondel M, Fujino A, Ueda N, et al. Higher-order factorization machines[C] //Proc of the 30th Int Conf on Neural Information Processing Systems. New York: Curran Associates, 2016: 3359−3367

[25] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C] //Proc of the 31st Int Conf on Neural Information Processing Systems. New York: Curran Associates, 2017: 5998−6008

[26] Pan Junwei, Xu Jian, Ruiz A L, et al. Field-weighted factorization machines for click-through rate prediction in display advertising[C] //Proc of the 27th World Wide Web Conf. New York: ACM, 2018: 1349−1357

[27] Hong Fuxing, Huang Dongbo, Chen Ge. Interaction-aware factorization machines for recommender systems[C] //Proc of the 33rd AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2019: 3804−3811

[28] Hu Jie, Shen Li, Sun Gang. Squeeze-and-excitation networks[C] //Proc of the 31st Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2018: 7132−7141

[29] Liu Bin, Zhu Chenxu, Li Guilin, et al. AutoFIS: Automatic feature interaction selection in factorization models for click-through rate prediction[C] //Proc of the 26th ACM SIGKDD Int Conf on Knowledge Discovery & Data Mining. New York: ACM, 2020: 2636−2645

[30] Kusupati A, Ramanujan V, Somani R, et al. Soft threshold weight

reparameterization for learnable sparsity[C] //Proc of the 37th Int Conf on Machine Learning. New York: ACM, 2020: 5544−5555

[31] Yu Jiahui, Lin Zhe, Yang Jimei, et al. Generative image inpainting with contextual attention[C] //Proc of the 31st Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2018: 5505−5514

[32] Yu Jiahui, Lin Zhe, Yang Jimei, et al. Free-form image inpainting with gated convolution[C] //Proc of the 17th Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2019: 4470−4479

[33] Louizos C, Welling M, Kingma D P. Learning sparse neural networks through L_0 regularization[C/OL] //Proc of the 6th Int Conf on Learning Representations. 2018[2022-11-10].https://openreview.net/forum?id=H1Y8hhg0b

[34] Su Yixin, Zhang Rui, Erfani S, et al. Detecting beneficial feature interactions for recommender systems[C] //Proc of the 34th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2021: 4357−4365

[35] Liu Siyi, Gao Chen, Chen Yihong, et al. Learnable embedding sizes for recommender systems[C/OL] //Proc of the 9th Int Conf on Learning Representations. 2021[2022-11-10].https://openreview.net/forum?id=vQzcqQWIS0q

[36] Zhou Kun, Yu Hui, Zhao Xin, et al. Filter-enhanced MLP is all you need for sequential recommendation[C] //Proc of the 31st ACM Web Conf. New York: ACM, 2022: 2388−2399

[37] Shi Cunhui, Hu Yaokang, Feng Bin, et al. A hierarchical knowledge based topic recommendation method in public opinion scenario[J]. Journal of Computer Research and Development, 2021, 58(8): 1811−1819 (in Chinese)
(史存会, 胡耀康, 冯彬, 等. 舆情场景下基于层次知识的话题推荐方法[J]. 机研究与发展, 2021, 58(8): 1811−1819)

Bai Ting, born in 1992. PhD, lecturer. Member of CCF. Her main research interests include information retrieval and data mining.

白 婷, 1992 年生. 博士, 讲师. CCF 会员. 主要研究方向为信息检索、数据挖掘.

Liu Xuanning, born in 2000. PhD. His main research interests include information retrieval and data mining.

刘轩宁, 2000 年生. 博士. 主要研究方向为信息检索、数据挖掘.

Wu Bin, born in 1969. PhD, PhD supervisor. His main research interests include data mining and graph data analysis.

吴 斌, 1969 年生. 博士, 博士生导师. 主要研究方向为数据挖掘、图数据分析.

Zhang Zibin, born in 1992. Master. Senior algorithm engineer at Tencent. His main research interests include recommendation system and data science.

张梓滨, 1992 年生. 硕士. 腾讯高级算法工程师. 主要研究方向为推荐系统、数据科学.

Xu Zhiyuan, born in 1990. Master. Senior algorithm engineer at Tencent. His main research interests include recommendation system and data science.

徐志远, 1990 年生. 硕士. 腾讯高级算法工程师. 主要研究方向为推荐系统、数据科学.

Lin Kangyi, born in 1985. Bachelor. Senior algorithm engineer at Tencent. His main research interests include recommendation system and data science.

林康熠, 1985 年生. 学士. 腾讯高级算法工程师. 主要研究方向为推荐系统、数据科学.