

基于积注意力交互网络模型的点击率预测

张安勤 王迎香 田秀霞

(上海电力大学计算机科学与技术学院 上海 200135)

摘要 如何提高广告点击率是对大数据网络营销的一个具有挑战的问题。考虑到用户点击行为的不确定性,提出一种基于积注意力交互网络模型的点击率预测模型。将用户的行为向量进行内积或外积,并根据广告自身的特征赋予交互后向量相应权重,然后进行点击率预测。在两个数据集上进行实验验证,结果表明该模型相对于传统的点击率预测模型在归一化基尼系数上提高了2%以上,预测效果更好。

关键词 点击率 注意力机制 因子分解机 内积 外积

中图分类号 TP301.6

文献标志码 A

DOI:10.3969/j.issn.1000-386x.2024.03.010

CLICK THROUGH RATE PREDICTION BASED ON PRODUCT ATTENTION INTERACTION NETWORK MODEL

Zhang Anqin Wang Yingxiang Tian Xiuxia

(College of Computer Science and Technology, Shanghai University of Electric Power, Shanghai 200135, China)

Abstract How to improve the click through rate of advertisement is a challenge to the network marketing in the era of big data. Considering the uncertainty of user's click behavior, a click-through rate prediction model based on product attention interactive network model is proposed. The model made the inner or outer product of the user's behavior vector, and gave the corresponding weight to the interactive vector according to the characteristics of advertising itself. Experiments were carried out on two data sets. The results show that the proposed model can improve the normalized Gini coefficient by more than 2% compared with the traditional hit rate prediction model, and can predict more accurately.

Keywords Click through rate Attention mechanism Factorization machine Inner product Outer product

0 引言

在大数据信息时代,随着互联网技术的不断发展,点击率预测在广告推荐的服务中占有很大比例,学术界和业界针对点击率预测方面开展了大量的相关研究^[1]。如何提高点击率预测的精度,将直接影响用户体验的满意度,预测的越准确,对于广告推荐效果越好,从而提高商家以较小的成本来获取最大的利润。

点击率预测的模型在近几年已经吸引了众多研究者的眼球。传统的广告点击率预测模型有因子分解机 FM (Factorization Machine),它是由 Rendle 等^[2]在 2010 年提出的。该模型核心在于特征组合,以此来减少人工参与特征组合工作,通过引入二阶组合特征来

增强线性回归模型^[3],可以在非常稀疏的数据中进行参数估计。因子分解机作为适用于任何实值特征向量的预测器,模拟所有嵌套变量交互,同时使用分解参数化方法,在线性的时间内完成计算。随着数据量的不断增大,仅仅特征两两之间的交互已经不能很准确地表现数据整体之间的联系,从而引导有更多的研究者考虑更高维的特征组合方式^[4]。

近年来,随着深度学习^[5-6]的发展,众多的研究者将深度学习应用于点击率预测。现阶段关于广告点击率的预测研究大致可分为针对特征交互的方式和针对时序的兴趣提取两个方面,研究者们提出了很多创新的模型。关于特征交互方式,2016年由微软团队 Shan 等^[7]提出的 Deep Crossing 模型,可以说是深度学习点击率预测的最典型和最基础性的模型,为了探究交叉

特征,该模型利用残差网络结构挖掘特征间的关系。同年,上海交大团队提出了由 Product-Based 神经网络构成的 PNN(Product-based Neural Network)模型^[8],通过构建内积层或外积层或内外积结合层,对嵌入后的特征向量进行组合,将得到的向量输入到神经网络中进行训练,从而使用不同的低阶交互方式增加了特征的交叉性。2016年由 Google 团队提出的 Wide & Deep 模型^[9],将线性模型与特征交叉结合,利用线性模型的记忆能力以及深度神经网络的泛化能力来对特征的相关性进行预测。2017年同样由 Google 团队提出的 DCN(Deep & Cross Network)模型^[10],在 Wide & Deep 模型的基础上,通过引入 Cross 网络代替原来的 Wide 部分,利用 Cross 来增加特征交互的力度,使用多层 cross 来对输入向量进行特征交叉,从而改进 Wide & Deep 模型部分表达能力不足的问题。2017年 He 等^[11]将深度学习结合在因子分解机上,提出了一种新的模型 NFM(Neural Factorization Machine),在嵌入层后加入 Bi-Interaction 操作来对二阶组合特征进行建模,从而学习更高维的特征组合方式。同年,由 Guo 等^[12]提出的 DeepFM(Deep Factorization Machine)通过并联的网络结构方式,在 FM 的基础上加入了神经网络部分,同时考虑了低阶特征和高阶特征。为了探究特征交互的不同方式,也引发了研究者们思考特征在数据中所占有的重要程度,由此注意力机制被研究者们所关注。注意力机制最早在 2014 年^[13]被应用于深度学习中,用来区别特征的重要程度。2017年 Xiao 等^[14]提出基于注意力机制的因子分解机模型 AFM(Attentional Factorization Machine),在因子分解机的基础上引入注意力机制,对于因子分解之后的结果赋予不同的权重,进行求和预测。

除了对于特征交互方式的探究,关于时序兴趣的提取,研究者们也提出很多模型,2018年, Zhou 等^[15]提出的 DIN(Deep Interest Network)模型,对用户的行为序列进行处理,用来提取用户的兴趣向量,通过引入注意力机制来对嵌入的向量赋予权重,根据不同广告的特性以及用户过去的点击行为来提取出用户的兴趣向量。同年,阿里团队提出的 DIEN 模型^[16],将用户的行为序列按照时间进行排序,利用 GRU(Gate Recurrent Unit)以及变种 GRU 模块来发掘用户行为序列中隐含的兴趣向量。2019年 Feng 等^[17]提出的 DSIN 模型,将用户的行为序列进行划分,每一个小的时间段作为一个会话,利用多头注意力机制和双向 LSTM 技术在会话中发掘用户的短期兴趣,从而对广告点击率进行预测。2020年 Xiao 等^[18]提出的 DMIN(Deep Multi-Interest Network)模型,利用两层多头注意力机制

分别对用户的行为序列进行行为的细化以及多兴趣的提取,从而提取用户的兴趣。对于广告点击率预测的两种方向,特征交叉的方式是根据用户的行为特征以及广告的自有属性来探究特征与特征之间的隐藏关系,从而得到用户可能会关注的信息。对于用户的行为序列的分析,可以得到用户长期或短期的兴趣方向,从而给出预测的结果。

对于特征交叉的方式这一方面,传统的点击率预测模型虽然考虑了特征交叉的方式以及特征交叉后向量的权重问题,但都没有充分将二者结合,没有形成优势互补。根据这一观察结论,针对解决用户的广告点击率问题,从特征交叉方式的角度,提出了基于内积或外积的注意力交互网络模型 PAIN(Product Attention Interaction Network),根据内积和外积的交互方式不同,可分为 IPAIN(Inner Product Attention Interaction Network)和 OPAIN(Outer Product Attention Interaction Network)两种模型。模型总体可以分为因子分解部分和神经网络部分,因子分解部分用来对低阶特征进行表示,神经网络部分通过结合了注意力机制的内积或外积的交叉方式来训练出高阶特征。在神经网络训练部分,通过对比不同的激活函数,选取线性整流激活函数的模型用来验证,最终使用模型中提出的归一化基尼系数^[19-20]来评判模型。实验结果可以证明本文提出的模型在预测归一化基尼系数的指标上相对于传统的点击率预测模型有较好的体现。

从特征交叉的角度,本文提出的模型相对于传统的点击率预测模型有以下几个优势:

(1)考虑特征交叉对于用户点击行为的影响,首次将注意力机制与内外积相结合,来筛选对用户点击行为影响较大的特征交叉组合。

(2)考虑高阶特征交互的同时,使用因子分解来对低阶特征向量进行组合,使整个模型考虑到属性自身对于用户点击行为的影响。

(3)实验使用归一化基尼系数作为评价指标,在 Kaggle 数据集和科大讯飞数据集上进行了多组对比实验,同时对比了不同激活函数对模型性能的影响。实验结果表明,与传统点击率预测模型相比,本文提出的 PAIN 模型在预测广告点击率方面具有优势。

1 相关工作

本节主要介绍了有关于不同特征交互方式的传统点击率预测模型。

1.1 AFM 模型

FM 模型是一种非线性模型,通过加入隐向量来获

取二阶交互特征。目前,被广泛应用于广告预测模型中,相比于传统的线性模型而言,因子分解机主要目标是解决在数据稀疏的情况下,学习特征交叉存在的联系。AFM模型在FM模型的基础上通过引入注意力机制,赋予特征交叉后向量相应权重,来区别不同特征交叉的重要程度。但是AFM模型没有考虑到低阶特征向量对结果的影响,并且赋予权重之后的向量也没有经过神经网络的训练,只是进行了简单的累加。相比之下,PAIN模型不仅考虑了低阶特征也同时应用神经网络来训练高阶特征,增强了模型的普遍性。

1.2 PNN模型

PNN模型在2017年提出,被用来探索特征交互的方式,通过利用内积和外积对特征进行交叉组合,再送往神经网络中进行训练,从而得出高阶组合特征来对点击率进行预测。PNN模型中对特征进行了处理,认为特征之间的关系更多的是一种“且”的关系,而不仅仅是“或”的关系。

PNN模型中提出的Product-base层,主要基于乘法的运算来体现特征交叉的网络结构。对于模型中的Product层,由两部分构成,一部分是线性部分,一部分是非线性部分,虽然模型在对于高阶特征训练时考虑了特征交叉的因素,但是忽略了有些不必要的特征交叉,在网络训练中同样会成为噪声而影响预测的效果。同时PNN模型只考虑了高阶特征的训练,但是忽略了低阶特征对预测的影响。PAIN模型不仅考虑了特征交叉后的权重问题,同时也考虑到了低阶特征的影响。

1.3 DeepFM模型

DeepFM模型建立在FM模型的基础之上,通过并联的方式加入了神经网络,用来训练高阶特征。虽然DeepFM训练出了高阶特征,并且使用因子分解机作为低阶特征的获取,但是没有考虑特征交叉在高阶特征训练中的必要性。同时DeepFM没有考虑到每个特征所存在的权重问题,对于那些不重要的特征,在网络训练中俨然成为噪声的一部分。PAIN模型不仅考虑了特征交叉在高阶特征训练的必要性,也考虑到特征所存在的权重问题。

2 模型描述

2.1 问题定义

假设数据集,先确定数据集中整体数据的特征类型、预测目标。若其包含多分类变量参数,需要先进行独热编码(One-HotEncoder)^[21]或者标签编码(LabelEncoder)进行编码。对于缺失值的处理,先确定缺失

值的位置,使用-1、平均值或者众数进行填充,从而尽可能地减小缺失值对预测结果的影响。对处理过后的数据集进行特征分析,删除关联度极小的特征,选取相关系数较大的特征列进行保留。

2.2 基于积注意力特征交互的深度神经网络

随着点击率预测在推荐服务中比重的增加,预测模型也在不断地更新发展,相对于线性求和的特征交互方式,本文在内外积基础上加入注意力机制,利用非线性的变化,可以表现出特征之间可能存在的耦合或不连续关系。积注意力特征交互通过改变内外积交互后的向量权重,并将加权过后的向量输入到神经网络中进行训练。根据不同特征之间的交互重要程度,由神经网络经过多层训练得到的结果来预测点击率。本文模型的框架如图1所示。

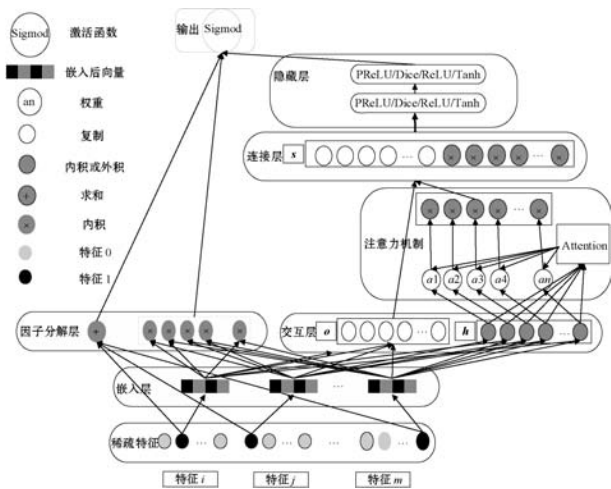


图1 PAIN模型

从模型图可以看出,PAIN模型主要包括两个部分,因子分解部分和深度神经网络部分,这两部分享有共同的输入。PAIN模型的预测结果可以写成式(1):

$$\hat{y} = \text{sigmoid}(y_{FM} + y_{DNN}) \quad (1)$$

因子分解部分如图2所示。

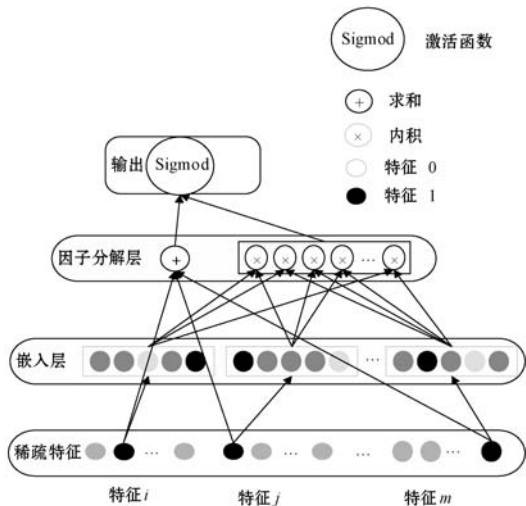


图2 因子分解

在因子分解层得到的结果为:

$$y_{FM} = w_0 + \sum_{i=1}^n w_i x_i + \sum_{i=1}^n \sum_{j=i+1}^n w_{i,j} x_i x_j \quad (2)$$

式中: n 代表样本的特征数量; x_i 是第 i 个特征的值; $w_0, w_i, w_{i,j}$ 是模型参数。

其中 $w_{i,j}$ 参数均可以组成一个对称矩阵 \mathbf{W} , 并分解为 $\mathbf{W} = \mathbf{V}^T \mathbf{V}$, \mathbf{V} 的第 j 列便是第 j 维特征的隐向量, 即每个参数 $w_{i,j} = \langle \mathbf{v}_i, \mathbf{v}_j \rangle$, 所以因子分解机模型公式为:

$$y_{FM} = w_0 + \sum_{i=1}^n w_i x_i + \sum_{i=1}^n \sum_{j=i+1}^n \langle \mathbf{v}_i, \mathbf{v}_j \rangle x_i x_j \quad (3)$$

式中: $\langle \rangle$ 符号代表的含义是内积, 如式(4)所示。

$$\langle \mathbf{v}_i, \mathbf{v}_j \rangle = \sum_{f=1}^k v_{i,f} \cdot v_{j,f} \quad (4)$$

神经网络部分如图 3 所示。

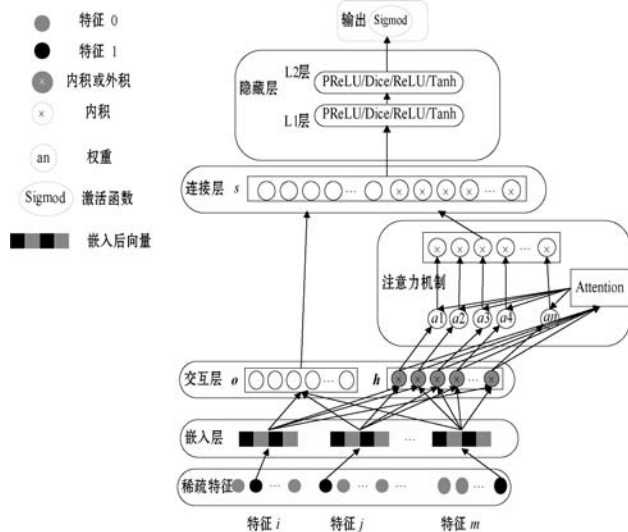


图 3 深度神经网络

自上而下的角度来看, 连接层的输出作为隐藏层的输入, 对于输出层来说, 其前馈过程如式(5)所示。

$$y_{DNN} = \sigma(\mathbf{W}_3 \mathbf{l}_2 + \mathbf{b}_3) \quad (5)$$

式中: $\mathbf{W}_3 \in \mathbf{R}^{1 \times D^2}$ 和 \mathbf{b}_3 是输出层的参数; $\mathbf{l}_2 \in \mathbf{R}^{D^2}$ 是第二个隐藏层的输出, 同时 D_i 代表第 i 个隐藏层的维度; $\sigma(x)$ 是 ReLU 激活函数。 \mathbf{l}_2 层的输出如式(6)所示。

$$\mathbf{l}_2 = \text{ReLU}(\mathbf{W}_2 \mathbf{l}_1 + \mathbf{b}_2) \quad (6)$$

式中: $\mathbf{l}_1 \in \mathbf{R}^{D^1}$ 是第一层的输出。实验对比了四种激活函数, 最终选取模型效果更好并且稳定的 ReLU 函数作为隐藏层的激活函数。

\mathbf{l}_1 层是与连接层完全连接, \mathbf{l}_1 层的输入是 \mathbf{l}_h 加权的向量和 \mathbf{l}_o 连接得到的 \mathbf{l}_s 向量, 则 \mathbf{l}_1 层的公式如式(7)所示:

$$\mathbf{l}_1 = \text{ReLU}(\mathbf{W}_1 \mathbf{l}_s + \mathbf{b}_1) \quad (7)$$

式中: $\mathbf{l}_1, \mathbf{W}_1, \mathbf{l}_s, \mathbf{b}_1 \in \mathbf{R}^{D^1}$ 。

首先定义内积的概念, 如式(8)所示。

$$A \odot B \triangleq \sum_{i,j} A_{i,j} B_{i,j} \quad (8)$$

即对 A, B 进行逐元素乘法, 然后将结果求和为一个标量。之后分别通过 \mathbf{o} 和 \mathbf{h} 计算 \mathbf{l}_o 和 \mathbf{l}_h :

$$\mathbf{l}_o = (l_o^1, l_o^2, \dots, l_o^n, \dots, l_o^{D^1}), \mathbf{l}_o^n = \mathbf{W}_o^n \odot \mathbf{O} \quad (9)$$

$$\mathbf{l}_h = (l_h^1, l_h^2, \dots, l_h^n, \dots, l_h^{D^1}), \mathbf{l}_h^n = \mathbf{a}_{i,j}^n \odot \mathbf{h}$$

式中: \mathbf{W} 为交互层的权重, $n \in \mathbf{R}$, $a_{i,j}$ 的计算式如式(10)所示。

$$\hat{a}_{i,j} = \mathbf{h}^T \text{ReLU}(\mathbf{W}(\mathbf{v}_i \odot \mathbf{v}_j) \mathbf{f}_i \mathbf{f}_j + \mathbf{b})$$

$$a_{i,j} = \frac{\exp(\hat{a}_{i,j})}{\sum_{ij} \exp(\hat{a}_{i,j})} \quad (10)$$

式中: $\mathbf{W} \in \mathbf{R}^{1 \times k}$; $\mathbf{b} \in \mathbf{R}^1, \mathbf{h} \in \mathbf{R}^1$; 而 $\mathbf{p} \in \mathbf{R}^k$; k 为嵌入后的向量维度; t 为注意力网络的隐藏维度。

对于式(9)中的 \mathbf{o} 和 \mathbf{h} , 有如式(11) - 式(12)公式的映射。

$$\mathbf{O} = (\mathbf{O}_1, \mathbf{O}_2, \dots, \mathbf{O}_N) \triangleq (\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_N) \quad (11)$$

$$\mathbf{h} = \{h_{i,j}\} \quad i = 1, 2, \dots, N; j = 1, 2, \dots, N \quad (12)$$

$$h_{i,j} = g(\mathbf{f}_i, \mathbf{f}_j)$$

式中: $\mathbf{f}_i \in \mathbf{R}^M$ 是特征 i 嵌入后的向量, $N \in \mathbf{N}^+$, g 代表函数, 对于不同的 g 有不同的计算方法。

(1) IPAIN。

当 g 选择内积时, 即为基于内积的注意力神经网络, 在 IPAIN 模型中, 由式(4)可知内积的定义, 则对于 \mathbf{o} 和 \mathbf{h} 的结果如式(13) - 式(14)所示。

$$\mathbf{l}_o^n = \mathbf{W}_o^n \odot \mathbf{O} = \sum_{i=1}^N \sum_{j=1}^N (\mathbf{W}_o^n)_{i,j} \mathbf{O}_{i,j} \quad (13)$$

$$\mathbf{l}_h^n = \mathbf{a}_{i,j}^n \odot \mathbf{h} = \sum_{i=1}^N \sum_{j=1}^N a_{i,j} \langle \mathbf{f}_i, \mathbf{f}_j \rangle \quad (14)$$

$\mathbf{a}_{i,j}$ 可以分解为:

$$a_{i,j} = \theta^n \theta^{n^T} \quad (15)$$

式中: $\theta \in \mathbf{R}^+$ 。因此式(14)可以表示为:

$$\mathbf{l}_h^n = \sum_{i=1}^N \sum_{j=1}^N \theta_i^n \theta_j^{n^T} \langle \mathbf{f}_i, \mathbf{f}_j \rangle = \langle \sum_{i=1}^N \alpha_i^n, \sum_{j=1}^N \alpha_j^n \rangle \quad (16)$$

式中: α_i 定义为 $\alpha_i = \theta_i \mathbf{f}_i$ 。通过 n 个节点的分解, \mathbf{l}_h 的计算式为:

$$\mathbf{l}_h = (\| \sum_i \alpha_i^1 \|, \dots, \| \sum_i \alpha_i^2 \|, \dots, \| \sum_i \alpha_i^{D^1} \|) \quad (17)$$

在时间复杂度的角度, 通过对 $\mathbf{a}_{i,j}$ 的分解, 将 \mathbf{l}_h 结果的次方降低为线性。这个方程使得到的一些中间结果可以重复使用, 从而降低了时间复杂度。

(2) OPAIN。

当 g 选择外积时, 即为基于外积的注意力神经网络, 在 OPAIN 模型中, 和内积不同的是向量外积取一对向量, 产生一个矩阵。在 OPAIN 模型中, 将特征交互定义为:

$$g(\mathbf{f}_i, \mathbf{f}_j) = \mathbf{f}_i \mathbf{f}_j^T \quad (18)$$

对于 h 中的每个元素, $\mathbf{h}_{i,j} \in \mathbf{R}^{M \times M}$ 是一个方阵。则对 h 的定义可表示为:

$$\mathbf{h} = \sum_{i=1}^N \sum_{j=1}^N \mathbf{f}_i \mathbf{f}_j^T = \mathbf{f}_\Sigma (\mathbf{f}_\Sigma)^T \mathbf{f}_\Sigma = \sum_{i=1}^N \mathbf{f}_i \quad (19)$$

式中: $\mathbf{h} \in \mathbf{R}^{M \times M}$ 是对称的, 因此 a^n 也是对称的。

(3) PAIN*。

PAIN* 模型, 是在 IPAIN 和 OPAIN 模型基础上, 同时计算嵌入后向量的内积以及外积, 将内积和外积应用注意力机制后的向量进行连接, 再将连接后的向量输入到神经网络中训练。

2.3 防止过拟合

推荐系统中的数据过于稀疏的特性通常容易导致过拟合的问题, 因此在模型中加入随机失活函数^[22]增加泛化能力、正则化项以约束参数大小, 防止最终解过于复杂, 本文使用最常见的 l_2 -范数正则化^[23]。并且通过优化函数 Adam^[24] 来学习模型参数, Adam 优化算法是随机梯度下降(SGD)^[25]的延伸算法。

3 实验与结果分析

3.1 数据集描述

本实验中使用了 2017 年 Kaggle 上 Porto Seguro 比赛和 2018 年科大讯飞 AI 营销算法大赛的公共数据集, 详细的数据统计如表 1 所示。

表 1 数据集

数据集	Test 属性列	Test 数据数	Train 属性列	Train 数据数
Kaggle	55	892 816	56	595 212
科大讯飞	35	1 001 650	34	40 024

3.2 评价指标

模型中抛弃了传统的模型评估指标 AUC, 使用归一化基尼系数作为模型的评价指标。之所以不用 AUC 来对模型进行评判, 是因为 AUC 主要是针对预测的正例排在负例之前的概率, 在 AUC 的基础上使用基尼系数来对模型进行评判, 不仅考虑了 AUC 的结果, 并且基尼系数也是对流行度分布的一种体现, 可以更好地体现出点击率的准确度。同时将得到的基尼系数进行归一化是为了消除奇异样本数据导致的不良影响。有关基尼系数的定义:

$$G = \frac{1}{n} \left(n + 1 - 2 \frac{\sum_{i=1}^n (n + 1 - i) y_i}{\sum_{i=1}^n y_i} \right) \quad (20)$$

式中: n 代表 y 的个数; 式(23)括号中 y_i 求和部分在本实验中代表总的真实值; 括号中 $(n + 1 - i) y_i$ 求和代表实验值累计求和。将得到的基尼系数进行归一化, 使用归一化后的基尼系数来评判预测结果的准确度。本实验中根据预测值对真实值进行从小到大的排序, 所以最后求得的归一化基尼系数越大越好。同时本实验使用了交叉切分进行多次训练, 求得的结果取平均值用来最终评判模型。

3.3 实验细节

实验在两个不同的数据集上进行验证, 数据分为训练集和测试集, 所使用的主要参数如表 2 所示。

表 2 主要参数

数据集	epoch	batch_size	learning_rate	l2_reg
Kaggle	30	1 024	0.000 5	0.01
科大讯飞	20	1 024	0.000 1	0.01

3.4 对比模型

将本文模型与以下模型进行对比:

(1) 神经网络(DNN)。直接将数据传入神经网络中训练。

(2) 因子分解机模型(FM)。在线性模型基础上, 考虑二阶特征交互。

(3) 神经因子分解机模型(NFM)。该模型在 FM 模型的基础上加入神经网络, 用来训练特征两两交互后的隐藏联系。

(4) 注意力因子分解机模型(AFM)。该模型在 FM 模型的基础上加入注意力机制, 给特征交互后的结果赋予不同的权重, 来体现不同交互结果对预测的影响。

(5) 深度因子分解机模型(DeepFM)。在 FM 模型的基础上, 通过神经网络来训练出高阶特征, 通过并联的模型结构, 将 FM 模型得到的低阶特征与神经网络训练的高阶特征相关联。

(6) PNN 模型。通过引入 Product 层, 分别使用内积、外积以及内积和外积相结合来训练出高阶特征。

(7) 深度兴趣网络模型(DIN)。利用用户的历史点击行为, 自适应地提取用户兴趣表示向量, 并利用注意力机制来赋予特征不同的权重。

3.5 实验对比分析

实验分别在两个数据集上对模型的性能进行验证, 利用本模型与主流的点击率预测模型进行对比, 实验进行了三次随机采样, 分别求出每次采样的归一化基尼系数, 最终取三次采样的平均值作为最终的归一化基尼系数。结果如表 3 所示。

表 3 模型结果对比

数据集	Kaggle	Kaggle	科大讯飞	科大讯飞
评价指标	方差	归一化基尼系数	方差	归一化基尼系数
DNN	0.008 12	0.224 26	0.000 40	0.523 58
FM	0.004 65	0.240 79	0.000 74	0.501 07
NFM	0.003 77	0.266 19	0.000 17	0.526 30
AFM	0.007 07	0.259 03	0.000 80	0.522 95
DeepFM	0.005 14	0.267 16	0.000 22	0.526 43
IPNN	0.004 67	0.265 53	0.000 26	0.524 88
OPNN	0.004 60	0.266 49	0.000 36	0.525 06
PNN *	0.004 68	0.264 21	0.000 88	0.521 49
DIN	0.004 86	0.264 88	0.000 09	0.526 34
IPAIN	0.002 81	0.267 73	0.000 38	0.526 80
OPAIN	0.002 47	0.269 30	0.000 34	0.526 78
PAIN *	0.005 71	0.268 35	0.000 26	0.526 75

在归一化基尼系数的指标中,由于实验是用预测值的大小来对真实值进行排序,所以归一化基尼系数越大越好,从表 3 中可以看出,实验提出的 PAIN 模型相比于主流的点击率预测模型,有更高的基尼系数。结合方差的大小,PAIN 模型的归一化基尼系数波动范围也相比于主流的点击率预测模型有更好的实验效果。综合来看,本文模型相对于其他主流点击率预测模型有较好的效果,一定程度上提升了预测的准确性。

3.6 模块对比分析

为了验证实验所提模型在引入注意力机制以及因子分解结构后模型效率的提升,在 Kaggle 数据集上对所提模型不加注意力机制和不加因子分解结构分别进行对比,实验在 IPAIN 模型的基础上进行,对比的归一化基尼系数结果表 4 所示。

表 4 模块对比

数据集	Kaggle	Kaggle
评价指标	方差	归一化基尼系数
IPAIN_noatt_fm	0.000 5	0.267 7
IPAIN_att_nofm	0.001 4	0.263 4
IPAIN_att_fm	0.000 1	0.269 4

可以看出,加入注意力机制以及因子分解后的模型的归一化基尼系数更大,模型的效果更好。同时参考对应归一化基尼系数的方差,同时加入注意力机制和因子分解的模型方差更小,模型的稳定性更高,由此得出加入相应模块对于模型性能的提升是可行的。

3.7 激活函数对比分析

实验在 IPAIN 模型的基础上进行,采用了四种不同的激活函数来对神经网络进行训练,其余参数均保持一致,来区分不同的激活函数对模型性能的影响。在实验数据集上,分别用 ReLU 激活函数、Tanh 激活函数、PReLU 激活函数^[26]以及自适应(Dice)激活函数对神经网络进行激活,实验的结果如表 5 激活函数对比所示。

表 5 激活函数对比

数据集	Kaggle	Kaggle	科大讯飞	科大讯飞
评价指标	方差	归一化基尼系数	方差	归一化基尼系数
Dice	0.005 10	0.268 32	0.000 73	0.526 83
PReLU	0.005 31	0.267 77	0.000 22	0.526 73
ReLU	0.005 30	0.268 78	0.000 29	0.526 79
Tanh	0.004 28	0.269 10	0.000 19	0.526 88

可以看出,在实验数据集上,使用 Tanh 激活函数的效果均比其余激活函数的效果要更好,可以得到更高的基尼系数以及更低的方差。但是实验最终选择了 ReLU 函数作为模型的激活函数,因为 ReLU 函数不仅解决了网络训练时梯度消失的问题,也加快了网络的训练速度,同时在训练节点为负值时不会被激活,避免了错误值对结果的影响。所以实验最终采取 ReLU 激活函数来训练神经网络部分。

4 结 语

本文探讨了基于因子分解机、注意力机制的点击率预测模型的改进实验,从而引入神经网络,进一步对结果进行训练,针对解决用户在日常生活中的广告点击率问题,最终提出了将因子分解机、内外积、注意力机制以及神经网络结合于一体的模型,利用内外积的特征交互方式来发掘用户过去行为习惯的隐藏兴趣,再利用注意力机制来对交互后的兴趣向量进行筛选。本文在 Kaggle 比赛以及科大讯飞比赛数据集上进行了对比实验,对比的模型为主流的点击率预测模型,采用归一化的基尼系数作为评判标准。实验结果表明,相比于神经因子分解机模型以及注意力因子分解机模型,本文的模型在预测中有更高的基尼系数,从而证明了本文模型在给定两个数据集中所实验的效果对比模型更好,在一定程度上提高了模型的性能。

尽管本文提出的模型性能有一定提高,但仍然有

很长的路要走。根据点击率预测和相关推荐方面的最新研究,提出了几个未来方向:

(1) 时序兴趣网络。尽管神经网络应用于注意力交互模型有一定的提升,但在现实生活中,用户的点击率受短期兴趣和长期兴趣的影响也有所不同。因此,有必要考虑到用户的时序兴趣行为,并设计更好的模型。

(2) 数据不平衡。对于点击率数据高维稀疏的缺点,可以通过引入迁移学习来生成对抗网络,完善数据集残缺的部分,从而更加准确地表现出用户的行为习惯。

随着深度学习对点击率预测的研究迅速发展,将点击率预测的模型应用于实际应用将会得到更多关注。在这种情况下,如何同时确保预测的准确性和模型的计算效率可能是最具有挑战性的问题。

参 考 文 献

- [1] 纪文迪,王晓玲,周傲英. 广告点击率估算技术综述[J]. 华东师范大学学报(自然科学版),2013,2013(3):1-14.
- [2] Rendle S. Factorization machines[C]//IEEE International Conference on Data Mining,2010:995-1000.
- [3] 高集体,赵林城. 部分线性回归模型中的自适应估计[J]. 中国科学A辑,1992(8):791-803.
- [4] 梅宏,黄罡,邢岩,等. 特征交互问题导论[J]. 电子学报,2002,30(S1):1923-1927.
- [5] Wang Q, Liu F, Xing S N, et al. Research on CTR prediction based on deep learning[J]. IEEE Access,2019,7:12779-12789.
- [6] Bengio Y S, Courville A, Vincent P. Representation learning: A review and new perspectives[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence,2013,35(8):1798-1828.
- [7] Shan Y, Hoens T, Jiao J, et al. Deep crossing: Web-scale modeling without manually crafted combinatorial features[C]//22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,2016:255-262.
- [8] Qu Y R, Cai H, Ren K, et al. Product-based neural networks for user response prediction[C]//16th International Conference on Data Mining,2016:1149-1154.
- [9] Cheng H, Koc L, Harmsen J, et al. Wide & deep learning for recommender systems[C]//1st Workshop on Deep Learning for Recommender Systems,2016:7-10.
- [10] Wang R X, Fu B, Fu G, et al. Deep & cross network for ad click predictions[C]//23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,2017:1-7.
- [11] He X N, Chua T. Neural factorization machines for sparse predictive analytics[C]//40th International ACM SIGIR Conference on Research and Development in Information Retrieval,2017:355-364.
- [12] Guo H F, Tang R M, Ye Y M, et al. DeepFM: A factorization-machine based neural network for ctr prediction[C]//26th International Joint Conference on Artificial Intelligence,2017:1725-1731.
- [13] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate[EB]. arXiv:1409.0473v1,2014.
- [14] Xiao J, Ye H, He X N, et al. Attentional factorization machines: Learning the weight of feature interactions via attention networks[C]//26th International Joint Conference on Artificial Intelligence,2017:3119-3125.
- [15] Zhou G R, Gai K, Zhu X Q, et al. Deep interest network for click-through rate prediction[C]//24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining,2018:1059-1068.
- [16] Zhou G R, Mou N, Fan Y, et al. Deep interest evolution network for click-through rate prediction[C]//33rd AAAI Conference on Artificial Intelligence,2019:5941-5948.
- [17] Feng Y F, Lv F Y, Shen W C, et al. Deep session interest network for click-through rate prediction[C]//28th International Joint Conference on Artificial Intelligence,2019:2301-2307.
- [18] Xiao Z B, Yang L W, Jiang W, et al. Deep multi-interest network for click-through rate prediction[C]//29th ACM International Conference on Information and Knowledge Management,2020:2265-2268.
- [19] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[C]//32nd International Conference on International Conference on Machine Learning,2015:448-456.
- [20] 张彦琦,唐贵立,王文昌,等. 基尼系数和泰尔指数在卫生资源配置公平性研究中的应用[J]. 中国卫生统计,2008,25(3):243-246.
- [21] Prusa J D, Khoshgoftaar T M. Improving deep neural network design with new text data representations[J]. Journal of Big Data,2017,4(1):7.
- [22] Hinton G E, Srivastava N, Krizhevsky A, et al. Improving neural networks by preventing co-adaptation of feature detectors[EB]. arXiv:1207.0580,2012.
- [23] Murugan P, Durairaj S. Regularization and optimization strategies in deep convolutional neural network[EB]. arXiv:1712.04711v1,2017.
- [24] Kingma D, Ba J. Adam: A method for stochastic optimization[EB]. arXiv:1412.6980v1,2014.

从上述系统响应曲线的仿真结果可以看出,改进 FOSMC-DFC 方法具有较强的抗扰动能力、精确程度高,对推力和磁链抖动的抑制效果强,大幅度削弱超调量,转速、磁链和推力波动得到明显改善,系统响应速度更快,利用组合趋近律能够有效削弱系统振荡,显著提高整体性能。

5 结 语

本文针对 PMLSM-DFC 中 FOSMC 存在的精确程度低、磁链和推力抖动大等问题,提出一种改进的 FOSMC,无超调量,减小系统振荡,增强精确程度。仿真结果证明采用改进 FOSMC 方法的 PMLSM-DFC 具有更好的跟随能力和强鲁棒性,能够大幅度削弱磁链和推力抖动,提高响应速度,增强抗扰动能力。采取分数阶双幂次指数趋近律与分数阶变速趋近律相互作用的组合趋近律可以进一步增强整体运行能力,显著抑制系统振荡,适合应用于对系统性能和精确程度有很高需求的控制系统。

参 考 文 献

[1] 王健,蓝益鹏. 磁悬浮电励磁三相直线同步电机离线参数辨识[J]. 电机与控制应用,2019,46(12):47-54.

[2] 宋雪玮,赵吉文,董菲,等. 基于 PSO-LSSVM 的永磁同步直线电机局部退磁故障识别[J]. 中国电机工程学报,2019,39(8):2426-2435.

[3] 崔霆锐,宗立明,李熙. 直线电机地铁车辆牵引系统国产化应用研究[J]. 电机与控制应用,2018,45(4):61-66,89.

[4] 殷红梅,姜海林,汪木兰,等. 基于直接转矩控制策略的 PMLSM 电磁推力控制与仿真[J]. 机械设计与研究,2018,34(2):113-116,121.

[5] 赵希梅,吴勇慷. 基于自适应修正拉盖尔递归神经网络的永磁直线同步电机反推控制[J]. 电工技术学报,2018,33(10):2392-2399.

[6] 季明丽,钱厚亮. 基于扰动观测器和推力观测器的永磁直线同步电机推力控制研究[J]. 机电工程,2019,36(3):333-337.

[7] 向凡,许鸣珠. 基于改进滑模速度控制器的永磁直线同步电机直接推力控制[J]. 电机与控制应用,2019,46(6):38-43.

[8] 原浩,赵希梅. 基于积分滑模的永磁直线同步电动机直接推力控制[J]. 电工技术学报,2019,34(3):483-488.

[9] Yang Z B, Zhang D, Sun X D, et al. Adaptive exponential sliding mode control for a bearingless induction motor based on a disturbance observer[J]. IEEE Access,2018,6:35425

-35434.

[10] Pukdeboon C. Extended state observer based third order sliding mode finite time attitude tracking controller for rigid spacecraft[J]. Science China Information Sciences,2019,62(1):1-16.

[11] Chen W, Zeng S, Zhang G Z, et al. A modified double vectors model predictive torque control of permanent magnet synchronous motor[J]. IEEE Transactions on Power Electronics,2019,34(11):11419-11428.

[12] He Z Y, Dong F, Zhao J W, et al. Thrust ripple reduction in permanent magnet synchronous linear motor based on tuned viscoelastic damper[J]. IEEE Transactions on Industrial Electronics,2019,66(2):977-987.

[13] Xu W, Elmorshedy M, Liu Y, et al. Finite-set model predictive control based thrust maximization of linear induction motors used in linear metros[J]. IEEE Transactions on Vehicular Technology,2019,68(6):5443-5458.

[14] 张言. 开关磁通永磁直线电机直接推力控制策略的研究[D]. 西安:西安电子科技大学,2018.

[15] 菅玲,董婷. 一种削弱永磁直线同步电机振动的方法研究[J]. 机电信息,2020(12):28-30.

[16] 李争,王蕾永,史雁鹏,等. 改进型积分滑模算法在永磁同步直线电机中的应用[J]. 微电机,2019,52(6):40-44.

[17] 卢琴芬,沈熸明,叶云岳. 永磁直线电动机结构及研究发展综述[J]. 中国电机工程学报,2019,39(9):2575-2588.

[18] 文叶叶. 基于分数阶滑模转速调节器的 PMSM 控制研究[D]. 西安:西安理工大学,2020.

(上接第 15 页)

[22] Elmhadi L, Karray M, Archimède B. A modular ontology for semantically enhanced interoperability in operational disaster response[C]//16th International Conference on Information Systems for Crisis Response and Management-ISCRAM, 2019:1021-1029.

[23] Fathy N, Gad W, Badr N. A unified access to heterogeneous big data through ontology-based semantic integration [C]//2019 9th International Conference on Intelligent Computing and Information Systems,2019:387-392.

(上接第 69 页)

[25] Ge R, Huang F R, Jin C, et al. Escaping from saddle points-online stochastic gradient for tensor decomposition [EB]. arXiv:1503.02101,2015.

[26] He K M, Zhang X Y, Ren S Q, et al. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification[C]//IEEE International Conference on Computer Vision,2015:1026-1034.