

# 基于 DeepFM 的深度兴趣因子分解机网络

王瑞平 贾真 刘畅 陈泽威 李天瑞

西南交通大学信息科学与技术学院 成都 611756

(abcwrp@163.com)



**摘要** 推荐系统能够根据用户的喜好从海量信息中筛选出其可能感兴趣的信息并进行排序展示。随着深度学习在多个研究领域取得了良好的效果,其也开始应用于推荐系统。目前基于深度学习的推荐排序算法常采用 Embedding&MLP 模式,只能获得高阶的特征交互。为了解决该问题,DeepFM 在上述模式中加入了因子分解机(Factorization Machine, FM),能够实现端到端的低阶与高阶特征交互学习,但其缺乏用户兴趣多样性的表示。鉴于此,通过将多头注意力机制引入 DeepFM,提出了深度兴趣因子分解机网络(Deep Interest Factorization Machine Network, DIFMN)。DIFMN 能够根据待推荐的不同物品自适应地学习用户表示,展示用户兴趣的多样性。此外,该模型根据用户历史行为的种类添加了喜好表征,从而不仅能够应用于只记录用户爱好的历史行为的任务,还可以处理同时记录用户喜欢与不喜欢的历史行为的任务。采用 tensorflow-gpu 进行算法的实现,在 Amazon(Electronics)和 movieLen-20m 两个公开数据集上进行对比测试,实验表明所提算法相比 DeepFM 分别有 17.70%和 35.24%的 Relaimpr 提升,验证了其可行性与有效性。

**关键词:** 推荐算法; DeepFM; 多头注意力机制; 深度学习; CTR 预测; 用户兴趣建模

中图法分类号 TP391

## Deep Interest Factorization Machine Network Based on DeepFM

WANG Rui-ping, JIA Zhen, LIU Chang, CHEN Ze-wei and LI Tian-rui

School of Information Science and Technology, Southwest Jiaotong University, Chengdu 611756, China

**Abstract** The recommendation system can sort out and display the information that may be of interest from the mass of information according to users' preferences. As deep learning has achieved good results in multiple research fields, it has also begun to be applied to recommendation systems. However, the current recommendation ranking algorithms based on deep learning often use Embedding & MLP mode and can only obtain high-level feature interactions. In order to solve the problem that only high-order feature interaction can be obtained, DeepFM adds FM to the above mode, which can learn the low-order and high-order feature interaction end-to-end. But the DeepFM cannot express the diversity of user interests. In view of this, this paper proposes a Deep Interest Factorization Machine Network(DIFMN) by introducing the multi-head attention mechanism into DeepFM. DIFMN can adaptively learn the user representation according to the different items to be recommended, showing the diversity of user interests. In addition, the model adds preference representations according to the type of user's historical behaviors, so that it can be applied not only to tasks that record only historical behaviors that the user likes, but also to tasks that record both historical behaviors that the user likes and dislikes. This paper uses tensorflow-gpu to implement the algorithm, and performs comparative tests on two public datasets of Amazon(Electronics) and movieLen-20m. Experiment results show that Relaimpr improves by 17.70% and 35.24% respectively compared to DeepFM, which validates the feasibility and effectiveness of the proposed method.

**Keywords** Recommendation algorithm, DeepFM, Multi-head attention mechanism, Deep learning, CTR prediction, User interest modeling

## 1 引言

随着计算机技术的迅速发展,互联网逐渐满足了人们对信息的需求,但同时也将人们带进了信息过载时代。目前,网络信息呈现出爆炸式增长的态势<sup>[1]</sup>,人们越来越难以找到适

合的信息,且获取感兴趣的信息需要付出的代价也越来越高昂。如何解决信息过载问题已成为一个热点话题。近年来,推荐系统被证明是一种应对信息过载的有效方法<sup>[2]</sup>。

推荐系统常常分为召回阶段和排序阶段,两个阶段的任务要求不同。在召回阶段,系统需要从大量的物品中根据用

到稿日期:2019-12-16 返修日期:2020-05-17 本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。

基金项目:国家重点研发计划(2017YFB1401400)

This work was supported by the National Key R&D Program of China(2017YFB1401400).

通信作者:李天瑞(trli@swjtu.edu.cn)

户的喜好筛选出一个用户可能感兴趣的物品子集。在排序阶段,系统使用更细粒度的特征对物品进行排序,最终将用户兴趣度高的物品靠前列。其中,对于排序阶段中物品的排序问题,常需要转换为物品点击率(Click-Through-Rate, CTR)预测任务。

CTR 预测任务用于估计用户点击推荐物品的概率,在推荐系统中极为重要。随着系统的数据量越来越大,人们将能够处理海量数据的深度学习方法应用到 CTR 预测任务<sup>[3-5]</sup>中,并取得了良好的效果。这些模型大多采用 Embedding & MLP 模式,即将高维稀疏的特征数据经过 embedding 层转化为低维稠密向量,之后将映射结果传入 MLP 模型进行高阶非线性特征交互的学习。为了使模型中的用户表示具有多样性,文献[5-6]在 Embedding&MLP 模式的基础上引入了注意力机制,能够根据候选物品动态生成相应的用户特征。但是,上述模型中只采用了通用注意力机制,不能从多个方面获取特征的表示;文献[7]在 embedding 层进行了修改,通过引入辅助损失及注意力机制,获得了用户的自适应表示,但其之后将表示传入 MLP 模型进行特征交互学习,也属于 Embedding&MLP 模式。Embedding&MLP 模式有其固有的局限,即在该模式下,模型只能学习高阶特征交互,而缺乏对低阶特征交互的学习。

CTR 预测任务主要使用的方法是进行用户与物品之间的特征交互建模。CTR 预测任务中用户与物品之间的交互特征主要有两种模式:浅层模式与深层模式。浅层模式是指能够明显找出来的特征关系,如高跟鞋与女性、雨天与雨伞等;深层模式则是指需要进行深层分析才能找出特征的相关性,如经典的啤酒与尿布的关系,即从表面属性看,两者并没有任何关系,需要进行更深层次的关系挖掘才能够显现两者的关系。在推荐系统中,用户与物品的属性种类繁多,属性之间的关系难以明确,为了取得较好的推荐效果,在进行 CTR 预测的同时需要考虑浅层与深层关系。

在 CTR 预测任务中同时考虑浅层与深层特征交互的有 Wide&Deep<sup>[8]</sup>模型,但是该模型需要特征工程。为了去除特征工程的使用,Guo 等<sup>[9]</sup>提出了 DeepFM 模型,但是该模型不能表示用户兴趣的多样性。

综上所述,推荐系统是计算机应用领域的一个研究热点。在物品点击率预测任务中,目前的推荐算法往往顾此失彼,或者关注用户的兴趣多样性表示而缺失浅层特征交互,或者关注浅层特征交互而缺失用户的兴趣多样性表示。本文基于 DeepFM 提出的 DIFMN 模型能够兼顾两者,此外,引入了用户特征的多样性表示,训练时间也不会有明显的增加。本文的主要贡献如下:

(1)在 DeepFM 基础上提出了一个神经网络模型——DIFMN,其能够同时对浅层特征和深层特征交互表示进行建模,并且能够关注用户兴趣多样性的特征表示;

(2)提出的模型能够处理只记录用户爱好行为特征的任务,也可以应用于同时记录用户喜欢与不喜欢的行为特征的情况;

(3)模型在两个公开数据集上都取得了很好的效果,从而验证了所提方法的可行性与有效性。

## 2 DIFMN 模型分析

### 2.1 特征表示

推荐系统的主要作用是将用户可能感兴趣的物品推荐给用户,因此需要对用户的喜好和物品的属性进行建模。用户和物品的表示能够决定最终推荐结果的质量,因此将用户喜好和物品进行高效地表示很有必要。

由于推荐系统中物品属性繁多,通过 one-hot 编码后每个物品的表征维数很高,具有稀疏性,需要通过 embedding 层将高维稀疏特征转化为低维稠密特征,但是直接将高维数据传入 embedding 层会导致参数过多,从而影响模型效果。本文采用基于领域知识的因子分解机(Field-aware Factorization Machines, FFM)<sup>[3]</sup>的格式,引入了 field 的概念,将相同性质的特征归入一个 field,对不同的 field 各自建模,形成各自的 embedding。对于不同的 field,输入的维度可能不同,但是要求输出的维度相同,以便确定 embedding 层的输出维度。Embedding 层的输出不仅与确定的 embedding 维数  $k$  有关,还与 field 的个数有关,其结构如图 1 所示。

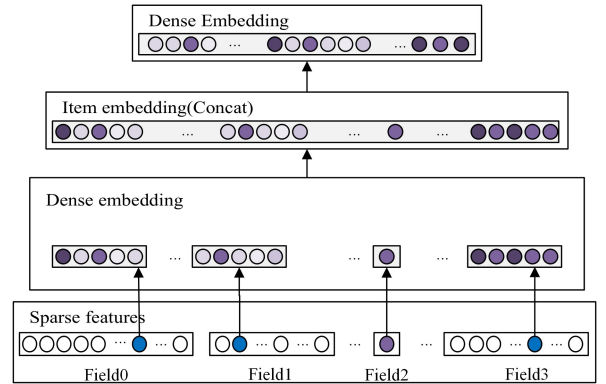


图 1 embedding 结构图<sup>[3]</sup>

Fig. 1 Structure diagram of embedding<sup>[3]</sup>

### 2.2 DeepFM

DeepFM 的结构如图 2 所示,包括 FM 与 DNN (Deep Neural Network)两部分,这两个部分共享输入,其本质是将推荐系统的 FM 模型与特征学习工程中的深度学习结合起来,从而能够同时学习浅层特征与深层特征交互信息。

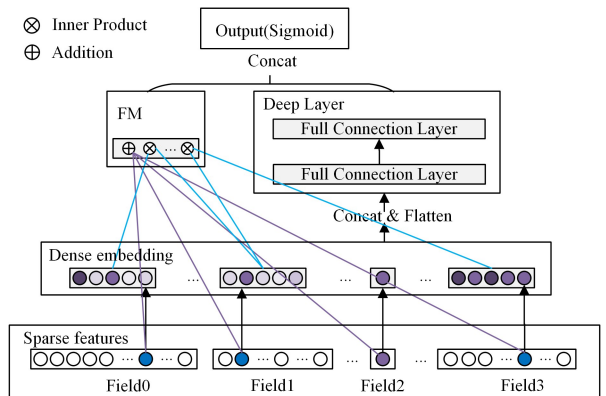


图 2 DeepFM 结构图<sup>[9]</sup>

Fig. 2 Structure diagram of DeepFM<sup>[9]</sup>

在 DeepFM 中,对于每一个特征  $i$ ,都有一个标量参数

$bias_i$  和一个向量参数  $e_i$  与其对应。其中, 标量参数表征特征  $i$  的一阶重要性; 向量特征同时应用在 FM 与 DNN 中。在 FM 部分, 向量参数通过与其他特征的向量参数进行内积来建模两个特征之间的二阶相关性; 在 DNN 部分, 将向量特征作为 DNN 的输入, 获得特征交互的高阶信息。DeepFM 的输出由 Sigmoid 函数产生, 如式(1)所示。

$$y' = \text{Sigmoid}(W[y_{FM}, y_{DNN}] + b) \tag{1}$$

其中,  $y' \in (0, 1)$  是整个模型的输出, 表示一个物品可能被点击的概率;  $[y_{FM}, y_{DNN}]$  表示将 FM 部分与 DNN 的输出进行拼接;  $W$  与  $b$  为全连接层参数。

DeepFM 中的 FM 部分是 Steffen 等<sup>[10]</sup> 在 2010 年提出的因子分解机, 其能够有效地解决高维稀疏特征组合问题。在 FM 提出之前, 衡量特征  $i$  与特征  $j$  之间的交互信息时, 需要两个特征同时存在于一条记录中, 但是推荐系统中的数据具有稀疏性, 同时记录两个特征信息的数据非常少, 因此不能进行有效的学习。FM 的提出为每一个特征引入了一个具有低维与稠密特性的向量特征, 并使用特征间向量特征的内积来衡量特征间的相关性, 此时即使推荐系统中两个特征共同存在的数据很少甚至没有, 也可以衡量两者之间的相关性, 从而有效解决了推荐系统中数据稀疏所导致的难以计算特征交互的问题。FM 能够衡量特征的一阶重要性与特征之间的二阶相关性, 其输出为一维特征与向量内积各元素的和:

$$y_{FM} = bias + \langle bias_i, x_i \rangle + \sum_{j_1=1}^d \sum_{j_2=j_1+1}^d \langle e_{i, j_2} \cdot e_{j, j_1} \rangle x_i \cdot x_j \tag{2}$$

其中,  $bias_i \in R^d, e \in R^k$  ( $d$  为特征数,  $k$  为向量维数);  $\langle bias_i, x_i \rangle$  表示特征之间的一阶重要性, 内积部分为特征之间的二阶交互, 其中每个特征的向量特征  $e$  通过 embedding 层来实现。

DNN 部分用于学习特征的高阶交互, 为全连接前馈神经网络。由于推荐系统中物品与用户的属性通常为分类与数值混合, 经过 one-hot 编码后, 数据具有稀疏性, 使得神经网络的学习极其困难, 且效果不佳, 因此需要通过 embedding 层将高维稀疏特征转化为低维稠密特征, 再输入全连接网络层。设 DNN 的输入为  $a^{(0)} = [e_0, e_1, \dots, e_d]$  ( $d$  为 field 个数), 则 DNN 每层的输出为:

$$a^{(l+1)} = \sigma(W^{(l+1)} a^{(l)} + b^{(l)}) \tag{3}$$

其中,  $l$  为层数,  $W$  和  $b$  为神经网络的参数。

### 2.3 DIFMN 模型

用户行为列表是根据用户兴趣直接交互产生的, 因此对用户行为列表的表示就是对用户兴趣进行了表征。DIFMN 是对 DeepFM 模型的改进。DeepFM 模型在进行用户兴趣表示时采用了平均池化, 使得获得的用户兴趣表示具有一致性, 缺乏多样性, 即对于任何候选物品, 用户的兴趣表征都相同。实际上, 用户的兴趣爱好是多样化的, 只是简单地使用行为列表中每个物品的平均表示不能展现出其多样化。例如, 一名用户常看玄幻与悬疑小说, 同时偶尔也看散文, 则用户行为列表中有大量的玄幻与悬疑小说而只有极少量的散文信息, 在为该用户推荐可能感兴趣的书籍时, 如果是玄幻或悬疑小说, 则系统能够进行正确的推荐, 而与散文相关的信息在最后的用户融合表示中只有极少甚至会被其他表示削弱而不存在,

从而不能进行散文的准确推荐。

为了解决 DeepFM 中存在的问题, DIFMN 通过在为用户行为列表中的每个物品进行 embedding 后, 引入多头注意力机制<sup>[11]</sup>。多头注意力机制是注意力机制的一种。近几年, 注意力机制的应用极为广泛。模型通过使用注意力机制, 能够在全局中明确需要重点关注的内容, 然后为该内容提供更多的注意力资源, 从而获取更多的需要关注目标的细节信息。多头注意力机制每个“头”的注意力机制中, 计算方式完全一致, 只是参数不同, 从而能够从多个子空间中进行特征表示, 相比通用的注意力机制能够获得更多的特征; 此外, 在输出维度相同时, 其计算时间更短。多头注意力机制在实际操作时只执行一次计算, 然后将结果进行合并。多头注意力机制的基础为缩放点积注意力机制, 缩放点积注意力机制的结构如图 3 所示。

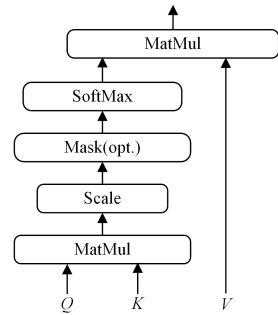


图 3 缩放点积注意力的结构图<sup>[11]</sup>

Fig. 3 Structure diagram of scaled dot-product attention<sup>[11]</sup>

缩放点积注意力机制的输入由  $Q, K$  和  $V$  组成, 其中,  $Q, K \in R^{b \times k}, V \in R^{b \times d}, b$  表示  $Q, K$  和  $V$  中分别有  $b$  个组成向量。缩放点积注意力机制首先计算  $Q$  与  $K$  的点积, 之后填加缩放层来抑制结果的增长速度, 因为当  $Q$  或者  $K$  中的数值较大时, 经过点积后的结果在数量级上会快速增加, 进而影响最终模型的结果, 因此需要增加缩放层来抑制结果的增长速度。缩放点积注意力机制最后的输出为:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{4}$$

多头注意力机制通过将  $Q, K$  和  $V$  进行  $h$  次不同的线性映射, 并对  $h$  次不同的映射结果并行地进行缩放点积注意力操作, 然后将缩放点积注意力的结果进行拼接后输入线性映射层, 最后得到多头注意力机制的结果。该机制通过  $h$  次线性映射, 使得模型能够在不同的表示子空间里学习到相关的信息。多头注意力机制的结构如图 4 所示。

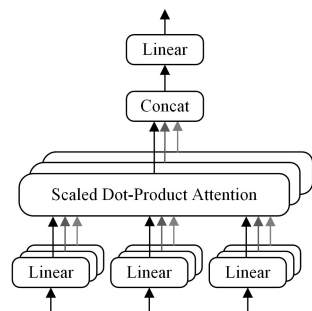


图 4 多头注意力机制的结构图<sup>[11]</sup>

Fig. 4 Structure diagram of multi-head attention mechanism<sup>[11]</sup>

多头注意力的输出如下:

$$\begin{cases} MultiHead(Q,K,V) = Concat(head_1, \dots, head_h)W^O \\ head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \end{cases} \quad (5)$$

其中,  $W_i^Q, W_i^K, W_i^V \in R^{d_{model} \times d_h}$ ,  $W^O \in R^{d_h \times d_{model}}$ 。

DIFMN 的结构如图 5 所示。模型在 embedding 之后加入了多头注意力机制,其在形成用户表示时能够更加关注在行为列表中与被推荐物品相关的物品,从而在进行推荐时能够根据候选物品动态地生成不同的用户表示,表现出用户爱好的多样性。

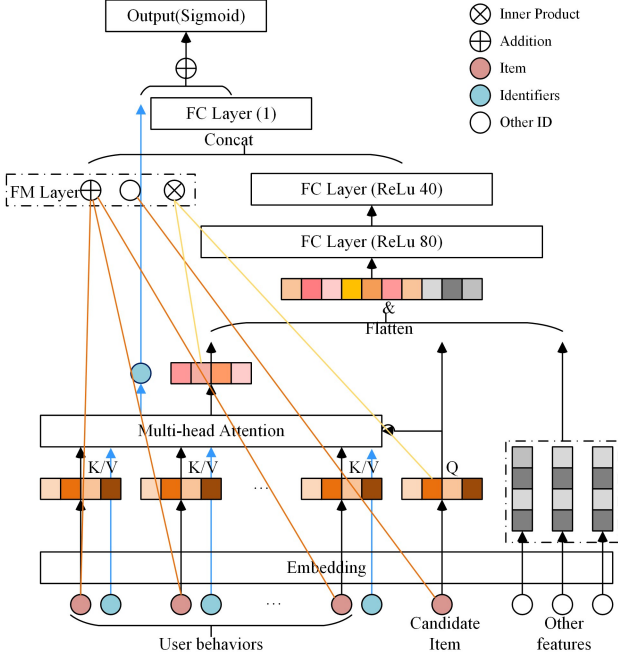


图 5 DIFMN 的结构图

Fig. 5 Structure diagram of DIFMN

多头注意力的加入,本质相当于在用户表示生成时增加了一个候选物品表示的相关性特征,其能够在不同的子空间中进行特征学习,增加了信息量。用户表示生成的主要机制如下:

$$v_u = MultiAttend(v_{candi}, (e_1, e_2, \dots, e_N)) \quad (6)$$

其中,  $v_u, v_{candi}, e_1, e_2, \dots, e_N \in R^{d_{model}}$ ,  $v_u$  为最后的用户表示,  $v_{candi}$  为候选物品,  $e_1, e_2, \dots, e_N$  为用户行为列表中的物品表示。

## 2.4 将喜好特征加入 DIFMN

DIFMN 增加了多头注意力机制,能够很好地基于记录用户喜欢的行为数据集进行推荐。但是,有些系统以相同的格式同时记录用户喜欢和不喜欢的物品,如果直接使用上述模型进行推荐,模型会根据被推荐的物品自动关注用户行为列表中相关的物品而形成最终的用户表示。如果该推荐物品与行为列表中用户喜欢的物品相似度高,则该物品会被推荐给用户;如果与行为列表中用户不喜欢的物品相似度高,也会由于与被推荐物品的相似度高而有很大概率被推荐给用户。这样,推荐系统不仅会推荐用户喜欢的物品,也会推荐用户不喜欢的物品,与推荐系统的作用相悖。

为了能够有效解决上述问题,模型根据用户的评分增加了一种记录用户对行为列表中每个物品的喜好特征(1 表示喜欢, -1 表示不喜欢)。由于多头注意力机制产生的权重衡量了在生成用户表征时行为列表中每个物品的相关性,因此通过将喜好特征与多头注意力机制中的权重相乘,能够得到与用户表征对应的喜好表征。最后将该表征加入之前模型的输出进行学习,结构如图 5 左上角所示。设用户喜好的记录特征为:

$$iden = (1, -1, 1, 1, \dots, 1, -1) \quad (7)$$

其中,  $iden \in R^d$ 。

多头注意力机制的权重为:

$$w = MultiAttend_w(v_{candi}, (e_1, e_2, \dots, e_N)) \quad (8)$$

其中,  $w \in R^d$ 。

喜好表示为:

$$label = w^T iden \quad (9)$$

最后模型的输出为:

$$output = Sigmoid(label + W[y_{FM}, y_{DNN}] + b) \quad (10)$$

其中,  $y_{FM}, y_{DNN}$  分别为 FM 部分与 DNN 部分的输出。将两部分拼接并经过线性映射后,与最终喜好表示相加,再经过 Sigmoid 激活函数,作为模型的最终输出。

## 3 实验

### 3.1 实验环境

代码在开源的深度神经网络框架 Tensorflow-1.9.0 上进行开发,底层为 Python3,所用操作系统为 Ubuntu 16.04.6。实验所用服务器为拥有 12GB 显存的 NVIDIA TITAN V GPU 和 11GB 显存的 GeForce RTX 2080 Ti GPU, Intel(R) Core (TM) i9-7920X CPU。

### 3.2 数据集

本文使用两个公开的具有用户行为的数据集作为实验数据。数据集的统计信息如表 1 所列。

表 1 数据集统计信息

Table 1 Dataset statistics

数据集	用户数	物品数	类别数	评价数
Amazon	192403	63001	801	1689188
MovieLens	138493	26744	20	20000263

(1) Amazon 数据集<sup>1)</sup>。Amazon 数据集收集了从 1996 年 5 月到 2018 年 10 月 Amazon 网站上的评论数据与元数据,为文献[5,12-13]的基准数据集。本文使用其 Electronics 数据子集作为实验数据集,每个用户至少有 5 个评论数据,具体包含 19203 名用户、63001 个物品、801 条类别信息和 1689188 条评论。本文在实验时采用与文献[5]一致的数据预处理方式,其中,用户对每个物品的评价都被认为是积极的。模型的任务为根据用户评价过的前  $n$  个物品预测用户对给定的第  $n+1$  个物品的喜欢程度。

(2) MovieLens 数据集<sup>2)</sup>。MovieLens 数据集<sup>[14]</sup>是一种稳定的基准数据集,包括匿名用户的评分及电影的描述,每个

<sup>1)</sup> <https://nijianmo.github.io/amazon/index.html>

<sup>2)</sup> <https://grouplens.org/datasets/movielens/>

用户至少有 20 个评分数据,其中用户的评分为 0~5 之间的连续数据,为避免在进行模型训练时发生数据倾斜,将低于 4 的评分标记为用户不喜欢(10004853 条),将高于或等于 4 的评分标记为用户喜欢(9995410 条)。MovieLens 数据集共包括 26744 部影片、138493 名匿名用户、20 种电影类型。实验时,将 100000 名用户作为训练集,测试集则由剩余的用户组成。在构成数据时,需要记录用户对每个物品的喜好,如果用用户喜欢则标记为 1,否则标记为-1。

### 3.3 对比方法

为了体现 DIFMN 能够实现对高低阶特征的有效提取,且能够获得用户兴趣的多样性表示,本文只与一些以 Embedding&MLP 为基本模式的 CTR 预测方法进行对比,其中不包括对 embedding 层或 MLP 层进行了修改的方法。

(1) MLP。多层感知机模型(Multilayer Perceptron, MLP)是将深度学习应用于推荐系统任务的基础模型。在 CTR 预测任务中有大量的模型都基于此模型,本文使用此模型作为弱基准模型。

(2) PNN<sup>[15]</sup>。Product-based Neural Networks(PNN)是对 MLP 的一种改进,在 embedding 层之后加入了一个乘积层来捕获属于分类类型的 filed 之间的交互信息。

(3) DIN<sup>[5]</sup>。深度兴趣模型(Deep Interest Network, DIN)在 MLP 的基础上引入了注意力机制,使得用户表示具有多样性,但是没有考虑浅层特征交互的应用。

(4) DIN\_MultiAT。将 DIN 中的注意力机制替换为多头注意力机制。

(5) DeepFM<sup>[9]</sup>。同时考虑浅层特征与深层特征交互,但是没有用户多样化的表示,是本文的强基准模型。

(6) DIFMN\_AT。利用通用的注意力机制来代替 DIFMN 中的多头注意力。

### 3.4 评价标准

在 CTR 预测任务中,AUC(Area Under the ROC Curve)是使用最广泛的评价标准,但是在测试集上 AUC 值高不代表在使用中一定有好的效果<sup>[8]</sup>。这主要是因为 AUC 将所有用户在所有场景下的结果一起进行排序计算,并不会根据用户或场景的不同而进行自适应性计算。对此,GAUC(GroupAUC)<sup>[16]</sup>应运而生。GAUC 首先计算每个用户的 AUC 值,然后对其进行加权平均,得到最终的 GAUC 值。本文采用 GAUC 作为评价标准。GAUC 的计算方式如下<sup>[5]</sup>:

$$GAUC = \frac{\sum_{i=1}^n \# impression_i \times AUC_i}{\sum_{i=1}^n \# impression_i} \quad (11)$$

其中, $n$  表示用户数, $\# impression_i$  表示给用户  $i$  展示物品的次数, $AUC_i$  表示与用户  $i$  相关的 AUC 值。由于对 Amazon 数据集和 MovieLens 数据集的处理方式不同,因此对于不同数据集, $AUC_i$  的计算方式不同。

对于 Amazon 数据集,将用户评价过的物品标记为该用户的正样例,每个用户未评价过的物品标记对应用户的负样例。但是,每个用户评价过的物品在所有物品中的占比较小,从而使得正负类别不平衡。为解决该问题,本文采用欠采样方法,即针对每个用户评价过的物品序列对应生成相同长度

的未评价过的物品序列共同作为训练数据。本文采用文献[13]的 AUC 计算方法,如下所示:

$$AUC_i = \frac{1}{|E(u)|} \sum_{(i,j) \in E(u)} \delta(f_{u,i}(t_{ui}) > f_{u,j}(t_{ui})) \quad (12)$$

其中, $\delta(b) = \begin{cases} 1, & b = \text{true} \\ 0, & b = \text{false} \end{cases}$ , $f_{u,i}(t_{ui})$  和  $f_{u,j}(t_{ui})$  分别表示用户  $u$  在以正样例与负样例为历史信息的情况下对物品  $i$  的预测得分。设  $P_u$  表示用户评价物品列表, $N_u$  对应生成的负样例列表,则有: $E(u) = \{(i,j), i \in P_u, j \in N_u\}$ 。

对于 MovieLens 数据集,由于正负样本数接近 1:1,在训练时不需要进行负样本采样,因此  $AUC_i$  直接使用 sklearn 工具包中的 roc\_auc\_score 函数进行计算。

此外,本文采用与文献[5,17]一致的评价指标 RelAImpr 来衡量各种方法的相对重要性。RelAImpr 的定义如下:

$$RelAImpr = \left( \frac{AUC(\text{measured model}) - 0.5}{AUC(\text{base model}) - 0.5} - 1 \right) \times 100\% \quad (13)$$

### 3.5 实验分析

为了验证网络模型,首先在 Amazon 数据集上进行了实验对比,实验结果如表 2 所列。

表 2 不同方法在 Amazon 数据集上的结果

Table 2 Results of different methods on amazon dataset

Model	GAUC	RelAImpr/%
MLP	0.8402	-4.11
PNN <sup>[15]</sup>	0.8469	-2.23
DIN <sup>[5]</sup>	0.8737	5.33
DIN_MultiAT	0.8888	9.58
DeepFM <sup>[9]</sup>	0.8548	0.00
DIFMN(AT)	—	—
DIFMN	0.9176	17.70

在进行参数设置时,本文采用与文献[5]一致的方式:在 DNN 中采用 3 层全连接层,每层节点数为 80,40 和 1;每个 field 的 embedding 大小为 64;每个批次的大小为 256;激活函数与 DeepFM 的参数一致,为 ReLu;优化器选用 Adam;学习率为 0.001。我们发现,在模型验证时,从 6 批次开始模型效果减弱,因此本文设定批次为 10。

实验中的所有结果均是进行 5 次实验后的结果的平均值。从表 2 可以看出,在进行 CTR 预测中,MLP 模型的效果最差,远差于 DeepFM 的效果。PNN 相对于 MLP,在 embedding 后增加了一个乘积层,效果有一定的提升,但是差于 DeepFM。DIN 相对于 MLP,在 embedding 后添加了一层注意力机制,效果有了明显的提升,该方法使用候选物品对用户历史行为进行注意力对齐,使得用户有多样性的表示。DIFMN(AT)在 DeepFM 中的 embedding 层后添加了注意力层,但是最后的效果不稳定,波动幅度较大。原因可能在于注意力机制只能在一个表示子空间中进行特征学习,对应获得的用户多样性表示在 DeepFM 中的 FM 部分进行低阶特征交互时效果欠佳。考虑到上述问题与多头注意力机制的原理,将多头注意力机制引入 CTR 预测任务的模型中。DIN\_MultiAT 为在 DIN 的基础上,将通用的注意力机制替换为多头注意力机制,相比 DIN,GAUC 提高了 1.51%,RelAImpr 提高了 4.25%;而将多头注意力机制引入 DeepFM 后整体效果大

幅提升,相比 DeepFM,RelaImpr 有了 17.70% 的提升,且效果稳定。DIFMN 在具有丰富用户行为的 Amazon 数据集上有突出的效果,主要原因是 DIFMN 在 DeepFM 上加入了多头注意力机制,既有 DeepFM 的结构,能够同时利用特征的低价与高阶交互,又能通过多头注意力机制在生成用户表示时更加关注与被推荐物品相近的物品,另外从多个表示子空间中进行特征学习,使得用户特征更加稳定且具有多样性。在 Amazon 数据集上的实验证明了多头注意力机制加入 DeepFM 模型的有效性。模型的训练时间如图 6 所示。

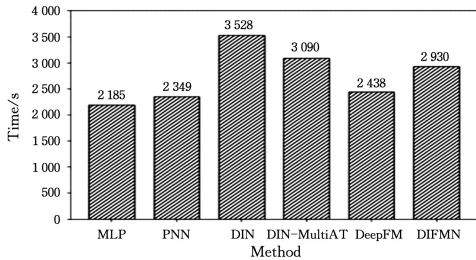


图 6 训练时间的对比

Fig. 6 Comparison of training time

由图 6 可以看出,MLP 训练时间最短,PNN 次之。DIN 与 PNN 都是在 embedding 后进行结构的改变,但是 DIN 训练时间有显著的增加,改为多头注意力机制后,DIN\_MultiAT 的时间更短,且效果有显著的改善,从而验证了多头注意力机制在 CTR 预测模型中的有效性。DIFMN 相对于 DeepFM 有了显著的效果提高;与此同时,增加的模型训练时间完全可以接受,这进一步验证了 DIFMN 模型的有效性。

MovieLens 数据集上的实验结果如表 3 所列。

表 3 不同方法在 MovieLens 数据集上的结果

Table 3 Results of different methods on MovieLens dataset

Model	GAUC	RelaImpr/%
MLP	0.6906	-8.50
PNN <sup>[15]</sup>	0.7047	-1.73
DIN <sup>[5]</sup>	0.6954	-6.19
DeepFM <sup>[9]</sup>	0.7083	0.00
DIN_MultiAT_noLabel	0.7017	-3.17
DIN_MultiAT	0.7797	34.28
DIFMN_noLabel	0.7061	-1.06
DIFMN	0.7817	35.24

在进行参数设置时,由于实验设备的内存限制,批次大小改为 128,其余设置与在 Amazon 数据集上的实验设置一致。此外,为了验证加入喜好特征的有效性,本实验增加了两个对比模型:DIN\_MultiAT\_noLabel 和 DIFMN\_noLabel。其中,DIN\_MultiAT\_noLabel 相对于 DIN\_MultiAT 缺少了喜好特征的应用,DIFMN\_noLabel 相对于 DIFMN 缺少了喜好特征的使用。

由表 3 可以看出,基于 MovieLens 数据集进行实验时,与 Amazon 数据集的实验一致,MLP 和 PNN 的性能结果低于 DeepFM,同时 PNN 的效果略高于 MLP。但是,在 MovieLens 数据集上,DIN,DIN\_MultiAT\_noLabel 和 DIFMN\_noLabel 的实验效果却低于 DeepFM。其原因可能在于,上述模型都采用了注意力机制,针对被推荐物品,在生成用户表示时更加关注用户行为列表中与被推荐物品相关的物品信息,

且将所有物品都看作一致,而忽略了用户交互过物品的喜好信息,容易将用户不喜欢但与行为列表中相似度高的物品推荐给用户。而 DIN\_MultiAT 加入了喜好特征,相比 DeepFM,GAUC 提高 7.14%,RelaImpr 提升了 34.28%。而 DIFMN 也取得了最好的效果,相比 DeepFM,GAUC 提高了 7.34%,RelaImpr 提升了 35.24%,从而验证了加入喜好特征是有有效的。

## 4 相关工作

CTR 预测任务用来估计用户点击推荐物品的概率,在推荐系统中极为重要。在 CTR 预测任务中,最简单和直观的是逻辑回归技术<sup>[17-18]</sup>。矩阵因子分解将交互矩阵进行分解,再进行重构,从而获得评分预测,但是随着用户与物品数据量的增加,分解难度与时间均持续增加。FM<sup>[9]</sup>主要用于稀疏特征的处理,并能够获得 1 阶与 2 阶的浅层特征交互,但不能获得高阶特征交互。为了解决上述问题,Yu 等<sup>[3]</sup>在 FM 的基础上引入 field 的概念,提出了 FFM,将每个 field 的 embedding 值传入 MLP,从而获取了高阶特征交互。PNN<sup>[15]</sup>在 FFM 的基础上增加了一个内/外乘积层,能够建模特征间的关系。Covington 等<sup>[4]</sup>从工业角度将深度学习应用于 CTR 预测问题,能够处理更多的特征表示,但是该模型对不同推荐物品采用的用户表示相同,不能表示出用户喜好的多样性。为了解决用户同时可能具有多种兴趣爱好的问题,研究者常在模型中引入注意力机制<sup>[19-20]</sup>。Zhou 等<sup>[5]</sup>提出了一个深度兴趣模型 DIN,在 embedding 层后添加了注意力机制,根据推荐的物品动态生成相应的用户特征。类似地,Chen 等<sup>[6]</sup>提出了 BST 模型,其本质是将 DIN 模型中的注意力机制替换为 Transformer,并根据用户浏览物品的时间加入 Position 特征。Zhou 等<sup>[7]</sup>在文献[5]的模型的基础上进行了改进,并提出了深度兴趣进化网络 DIEN,添加了兴趣抽取层与兴趣进化层,能够获得用户兴趣的准确表达与动态变化性,但其本质只是对用户 embedding 层的修改。上述模型都采用了 Embedding&MLP 模式,只获得了特征交互的深层次表示,并没有获取浅层特征交互的表示。Cheng 等<sup>[8]</sup>提出的 Wide&Deep 模型能够同时利用浅层特征和深层特征交互,但是在进行浅层特征交互表示时需要特征工程。Guo 等<sup>[9]</sup>为了减少 Wide&Deep 模型中的特征工程,提出了 DeepFM,将 embedding 后的特征表示同时传入浅层网络和深层网络,通过端到端的方式同时获得了浅层特征交互表示与深层特征交互表示,但是用户对所有物品的表示全部相同,不能体现出用户兴趣的多样性。

**结束语** 在推荐系统中,用户通常只关注排列靠前的几个推荐结果,因此需要将用户可能感兴趣的物品尽可能地靠前排,以提高推荐质量。本文基于 DeepFM 提出了改进的 DIFMN 模型。DIFMN 模型同时考虑用户与物品的浅层特征与深层特征交互,并能够根据推荐物品的不同动态生成相应的表征,表示其兴趣的多样性。此外,DIFMN 不仅能够处理只有用户爱好记录的推荐任务,还可以处理同时记录用户喜欢与不喜欢的推荐任务。最后,为了进行性能评价,在两个公开数据集上进行了实验,并在相同的评估指标下比较了不同

的方案,实验结果证明 DIFMN 在两个数据集上都能取得很好的效果。后续,我们会将 DIFMN 模型应用于政策项目推荐的实际应用中;此外,有研究表明 embedding 的表示能够影响推荐的效果,因此后续将在 DIFMN 模型的基础上探究 embedding 表示,以提高推荐效果。

### 参 考 文 献

- [1] MARZ N, WARREN J. Big Data: Principles and best practices of scalable realtime data systems[M]. Manning Publications, 2015.
- [2] RICCI F, ROKACH L, SHAPIRA B. Introduction to Recommender Systems Handbook[M]// Recommender Systems Handbook. Boston: Springer, 2011: 1-35.
- [3] YU C J, ZHUANG Y, WEI S C, et al. Field-aware factorization machines for CTR prediction[C]// Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016: 43-50.
- [4] COVINGTON P, ADAMS J, SARGIN E. Deep neural networks for youtube recommendations[C]// Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016: 191-198.
- [5] ZHOU G R, ZHU X Q, SONG C R, et al. Deep interest network for click-through rate prediction[C]// Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2018: 1059-1068.
- [6] CHEN Q, ZHAO H, LI W, et al. Behavior Sequence Transformer for E-commerce Recommendation in Alibaba[J]. arXiv:1905.06874, 2019.
- [7] ZHOU G R, MOU N, FAN Y, et al. Deep interest evolution network for click-through rate prediction[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2019: 5941-5948.
- [8] CHENG H T, KOC L, HARMSSEN J, et al. Wide & deep learning for recommender systems[C]// Proceedings of the 1st Workshop on Deep Learning for Recommender Systems. ACM, 2016: 7-10.
- [9] GUO H F, TANG R M, YE Y M, et al. DeepFM: a factorization-machine based neural network for CTR prediction[C]// Proceedings of the 26th International Joint Conference on Artificial Intelligence. AAAI Press, 2017: 1725-1731.
- [10] RENDLE S. Factorization machines[C]// Proceedings of 2010 IEEE International Conference on Data Mining. IEEE, 2010: 995-1000.
- [11] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]// Advances in Neural Information Processing Systems. 2017: 5998-6008.
- [12] MCAULEY J, TARGETT C, SHI Q, et al. Image-based recommendations on styles and substitutes[C]// Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2015: 43-52.
- [13] HE R, MCAULEY J. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering[C]// Proceedings of the 25th International Conference on World Wide Web. 2016: 507-517.
- [14] HARPER F M, KONSTAN J A. The movielens datasets: History and context[J]. ACM Transactions on Interactive Intelligent Systems, 2015, 5(4): 19.
- [15] QU Y, CAI H, REN K, et al. Product-based neural networks for user response prediction[C]// Proceedings of 2016 IEEE 16th International Conference on Data Mining. 2016: 1149-1154.
- [16] ZHU H, JIN J, TAN C, et al. Optimized cost per click in taobao display advertising[C]// Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2019: 2191-2200.
- [17] YAN L, LI W J, XUE G R, et al. Coupled group lasso for web-scale ctr prediction in display advertising[C]// Proceedings of International Conference on Machine Learning. 2014: 802-810.
- [18] RICHARDSON M, DOMINOWSKA E, RAGNO R. Predicting clicks: estimating the click-through rate for new ads[C]// Proceedings of the 16th International Conference on World Wide Web. ACM, 2007: 521-530.
- [19] FENG Y F, LV F Y, SHEN W C, et al. Deep session interest network for click-through rate prediction[C]// Proceedings of 28th International Joint Conference on Artificial Intelligence. 2019.
- [20] ZHU H, LI X, ZHANG P, et al. Learning tree-based deep model for recommender systems[C]// Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018: 1079-1088.



**WANG Rui-ping**, born in 1995, post-graduate. Her main research interests include recommendation algorithm and natural language processing.



**LI Tian-rui**, born in 1969, Ph.D, professor, Ph.D supervisor, is a distinguished member of China Computer Federation. His main research interests include big data intelligence, rough sets and granular computing.