熊重驰. 用于因子分解机中点击通过率预测的局部和全局上下文融合网络[J]. 智能计算机与应用,2026,16(1):169-177.
DOI:10.20169/j.issn.2095-2163.24032104

用于因子分解机中点击通过率预测的局部和全局上下文融合网络

熊重驰
(上海理工大学 光电信息与计算机工程学院,上海 200093)

摘 要:点击率预测是推荐系统中最核心的算法之一,近年来已经得到了广泛的应用。尽管目前在许多推荐任务利用深度神经网络获取丰富的潜在表示以提升推荐性能已被证明是有效的,但仍然面临以下问题:如何进一步利用和挖掘多种潜在表示进行推荐。为了解决该问题,本文提出了局部和全局上下文融合网络,该模型具有并行分支,包括 2 个强大的高阶特征表示学习组件,一个分支使用交叉注意力机制来捕获全局上下文嵌入,而另一个分支使用多层感知机模块来提取局部上下文嵌入。此外,局部和全局上下文融合网络使用信息融合单元将包含局部与全局上下文信息的高阶嵌入融合,用于指导上下文特征细化。在 2 个真实世界数据集上的实验表明,所提出的模型在点击率预测任务中显著优于所有最先进的模型。

关键词:点击率预测;因子分解机;推荐系统;上下文信息
中图分类号:TP391.3;TP391.1;TP183 **文献标志码**:A **文章编号**:2095-2163(2026)01-0169-09

Local and global context fusion network for CTR prediction in Factorization Machines

XIONG Zhongchi
(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: Click-Through Rate prediction is one of the core algorithms in recommendation systems and has been widely used in the real world in recent years. Although using deep neural networks to obtain rich latent representations to improve recommendation performance has been effectively proven in many recommendation tasks in recent years, they still face the following problem: how to further utilize and mine multiple latent representations for recommendation. To solve this problem, this paper proposes a local and global context-aware fusion network, which has parallel branches and includes two powerful high-order feature representation learning components. One branch uses a cross-attention mechanism to capture the global context embedding, while the other branch uses Multi Layer Perceptron modules to extract local contextual embeddings. In addition, the local and global context-aware fusion network uses the Information Fusion Component to fuse high-order embeddings containing local and global context information to guide context feature refinement. Experiments on two real-world datasets show that the proposed model significantly outperforms all state-of-the-art models in the Click-Through Rate prediction task.

Key words: Click-Through Rate prediction; Factorization Machine; recommendation system; contextual information

0 引 言

点击率预测(Click-Through Rate prediction)是互联网公司[1]和电子商务平台[2]提供各种个性化服务的核心功能模块,每天数以百万计的用户在互联网上留下足迹,准确的点击率预测可以对在线企业的收益产生积极影响。因此,应用深度神经网络来有效学习丰富的潜在表示受到了广泛的关注。近年来,许多方法通过对特征交互进行建模来丰富特征表示[3-7],传统基于因子分解机(Factorization Machines,FM)[8-9]的方法致力于建模低阶跨特征表示来丰富特征表示,如 FFM[8]、DIFM[10]与 IFM[11]。随着深度学习在捕捉高阶特征交互方面取得卓越表现,进一步提升了 CTR 预估的准确性,如 DCN-V2[1]、xDeepFM[12]和 AutoInt[13]。尽管现有的特征交互技术在提升模型性能方面有了一定进展,但大多数都只在单一范围内学习每个特征的固定表示,却并未考虑到不同范围内同一特征可能存在不同表示的情况。例如,在卡车销售商与物流提供商的广告中可能同时存在“卡车”这一特征,在局部范围内

基金项目:国家自然科学基金(61772342)。
作者简介:熊重驰(1998—),男,硕士研究生,主要研究方向:推荐系统。Email:18879417688@163.com。
收稿日期:2024-03-21

“卡车”的特征表示是相同的,但在全局范围中,“卡车”的特征表示应该不同。本文将同一特征在局部与全局范围内具有不同的表示称为局部与全局上下文表示。因此点击率预测模型在充分挖掘多种不同的潜在表示与结合其优势以实现卓越的预测性能方面,仍有巨大的未开发潜力。

在点击率预测中,局部和全局上下文信息的挖掘与利用还未得到充分的开发,并且如何将局部和全局上下文信息结合起来同时发挥两者的优势进行推荐也是目前点击率预测面临的一个重要难题。

为了解决这些问题,本文提出了局部和全局上下文感知融合网络(LGFNet),并将其集成在 FM 上。FM_{LGFNet} 包括上下文挖掘模块与信息融合模块两个子模块。其中,上下文挖掘模块使用了 2 个并行提取器、即全局上下信息提取器和局部上下信息提取器。全局上下信息提取器使用交叉注意力机制来捕获全局上下文关系,局部上下信息提取器使用多层感知机模块来提取局部上下文关系。信息融合模块将带有局部和全局上下文信息的高阶表示融合,以指导上下文特征细化。研究中为了防止梯度消失与网络退化,使用一个可训练的超参数 β 将原始特征与上下文特征以残差连接的方式自适应集成,以生成最终的融合表示。最后,本文在 2 个公共数据集上进行了广泛的实验。结果表明,FM_{LGFNet} 优于最先进的模型。

总地来说,本文主要贡献可以被总结如下:

(1)本文提出了一个名为 FM_{LGFNet} 的新模型,该模型通过挖掘局部与全局上下文嵌入,并使用残差连接将其自适应融合来得到高阶特征表示。

(2)在 2 个真实数据集上的实验结果表明,FM_{LGFNet} 胜过目前最先进的点击率预测方法。

1 相关工作

1.1 点击率预测模型

在过去的几年里,主流的点击率预测方法专注于使用神经网络模型对特征交互进行建模来丰富特征表示,从而提高点击率预测性能。根据最近的工作[1,3],点击率预测方法可以分为传统方法[8,10-11,14-16]和基于深度学习的方法[1,3,7,12-13,17-18]两种类型。FM 是一种用于点击率预测的通用预测方法,在即使数据非常稀疏的情况下,依然能估计出可靠的参数进行预测。与传统的简单线性模型不同的是,FM 考虑了特征间的交叉,对所有嵌套变量交互进行建模(类似于 SVM 中的核函数),因此在推荐系统和计算广告领域关注的点击率(Click-Through Rate, CTR)和转化率(ConVersion Rate, CVR)两项指标上有着良好的表现。基于 FM 衍生出了很多方法, Yu 等学者[11]提出了 IFM 模型,在计算特征交叉之前预估了一个因子来改进特征表示,尝试优化特征在不同上下文场景下的表示以改进 FM 方法。Lu 等学者[10]在 IFM 的基础上提出了 DIFM 模型,通过使用 Transformer[19]的结构来学习向量之间的关系,并与比特级的特征进行组合以丰富特征表示。Pan 等学者[15]在 FM 的基础上引入了“域”的概念,提出了 FFM 模型,在不同的域中同一特征被不同的隐向量表示,使得模型建模更加准确。然而,这些方法局限于建模低阶跨特征交互来丰富特征表示,无法捕捉高阶特征的交互作用。为了解决这个问题,许多基于深度学习的方法应运而生。神经因子分解机(Neural Factorization Machines, NFM)[20]采用 FM 来捕捉二阶相互作用,并通过使用 DNN 学习非线性特征相互作用。基于因子分解机的神经网络(DeepFM)[18]利用 FM 分量和深度神经网络来分别学习低-高特征交互,并获得有效的场表示。在 DeepFM 的基础上,xDeepFM[12]结合了卷积神经网络(CNN)和递归神经网络(RNN)的功能来挖掘显性和隐性高阶特征的组合,并提出了压缩交互网络(CIN),带来了更有效的高阶线性特征交叉。然而,这些方法只学习每个特征在单一范围内的表示,没有考虑每个特征在局部和全局范围内拥有不同的表示,导致传统方法在高阶特征交互的提取中存在挑战。

1.2 局部和全局上下文建模

点击率预测中,特别是推荐和广告场景,用户的行为序列很重要。在自然语言处理领域中,局部和全局上下文关系在序列建模中都发挥着重要作用。研究人员已经探索了各种方法,将这 2 种类型的上下文结合在一个统一的模型中。

目前主流框架有 2 种。第 1 种是 2020 年由 Gulati 等学者[21]提出的 Conformer,是 Transformer 的一个变体,由 2 个前馈层、1 个多头自注意模块和 1 个卷积模块组成,形成静态的单分支架构。基于 Conformer 的方法在许多语音应用中实现了最先进的性能[22]。然而,由于单分支设计,很难分析在不同层中如何利用局部和全局交互。为了缓解这个问题, Peng 等学者[23]在 2022 年在 Conformer 的基础上改良了结构,提出了 Branchformer,采用并行 MLP-Attention 结构将全局和局部上下文特征以相

互独立的方式并行提取，并且 2 种特征最后可以通过不同方式进行结合，充分发挥了局部与全局上下文特征的作用。此外，第 2 种方法为 Wu 等学者[24]提出的 Lite Transformer，也采用了基于标准自关注和卷积的双分支架构来捕捉全局和局部依赖。并采用了基于标准自关注和卷积的双分支架构来捕捉全局使用专门的并行分支来减少移动自然语言处理应用程序的总体计算和模型大小。上述模型在点击率预测领域取得了优异的性能，然而在点击率预测领域中局部与全局上下文建模技术还没有得到充分开发，因此这项技术在点击率预测领域的应用被研究者所关注。

2　方法

2.1　问题定义

给定的数据集 D 是由 n 个实例 (X,Y) 组成，其中 x_i 通常是一对用户–项目的交互记录，由分类字段（如性别，国家）或连续字段（如身高，年龄）组成，每个分类字段使用独热向量表示，每个连续字段使用值本身或离散化后的独热向量表示。每个实例都有相应的标签 $y_i \in \{0,1\}$。这里，$y_i = 1$ 表示用户点击了该项目，否则 $y_i = 0$。点击率预测的每个实例可以表示为：$\boldsymbol{x}_i = [x_{\text{field1}}, x_{\text{field2}}, \cdots, x_{\text{field}\,j}, \cdots, x_{\text{field}m}]$，其中 $\boldsymbol{x}_i$ 表示一个 d 维向量，$x_{\text{field}\,j}$ 表示 x 的第 j 个域的向量表示。点击率预测的主要目标是在给定输入特征向量 $\boldsymbol{x}_i$ 的情况下预测用户点击项目的概率。

2.2　$\text{FM}_{\text{LGFNet}}$

本文的目标是学习局部与全局上下文融合特征交互，为此本文提出了一种基于因子分解机的局部和全局上下文融合网络。$\text{FM}_{\text{LGFNet}}$ 的架构如图 1 所示。由图 1 可知，输入层将用户的配置文件和项目的属性的所有特征字段串联起来并表示为稀疏的高维特征向量；嵌入层将稀疏的高维特征转换为稠密的低维嵌入矩阵；LGFNet 集成在 FM 模型的嵌入层与特征交互层之间，为特征交互层提供具有上下文权重的稠密嵌入；特征交互层可以学习用于推荐的线性（一阶）以及成对（二阶）的特征交互；最后，输出层使用对数损失函数来计算模型的损失。

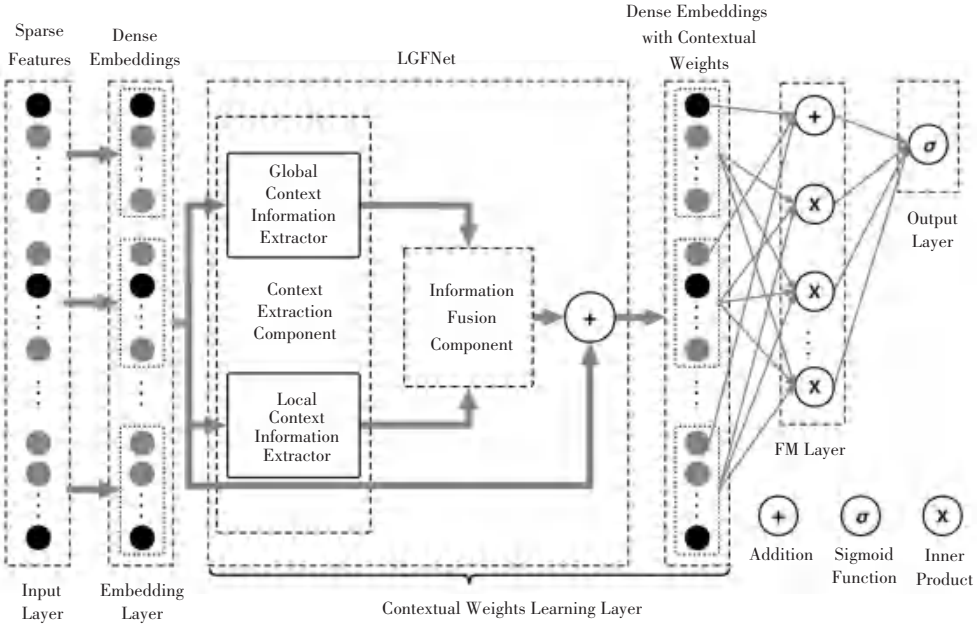

图 1　$\text{FM}_{\text{LGFNet}}$ 架构图

Fig. 1　Architecture diagram of $\text{FM}_{\text{LGFNet}}$

2.2.1　输入层

为了得到用户信息嵌入，首先在输入层将用户的配置文件和项目的属性的所有特征字段串联起来并表示为稀疏向量，可表示为：

$$\boldsymbol{x} = [x^1, x^2, \cdots, x^N] \tag{1}$$

其中，N 表示特征字段总数，x^i 表示第 i 个字段的特征表示。如果第 i 个字段是分类字段，则 x^i 使用一个独热向量表示。如果第 i 个字段是连续字段，则 x^i 使用其本身的值表示。

2.2.2　嵌入层

为了将稀疏的高维特征 x 转换为稠密的低维嵌入矩阵 $\boldsymbol{E} = [e^1, e^2, \cdots, e^N] \in \mathbb{R}^{N\times d}$；使用 look-up embedding table 对输入的实例字段 $(x_i^1, x_i^2, \cdots, x_i^N)$ 进行编码，以学习每个输入字段的潜在嵌入。嵌入的字段定义为：

$$e_i^n = \boldsymbol{x}_i^n \boldsymbol{W}^n, e_i^n \in \mathbb{R}^d \tag{2}$$

其中，$\boldsymbol{W}^n \in R^{Z_n \times d_n}$ 表示权重矩阵；Z_n 表示输入字段 X_i^n 的字段长度；d 表示字段的固定嵌入尺寸；$\boldsymbol{x}_i^n$ 表示实例 i 的第 n 个字段的二进制输入向量。

2.2.3 局部和全局上下文感知融合网络

局部和全局上下文感知融合网络（Local and Global context-aware Fusion Network，LGFNet）架构如图 2 所示。LGFNet 包含上下文提取组件与信息融合组件两个相互独立的并行组件，上下文提取组件可以并行独立挖掘包含局部和全局上下文信息的高阶关系表示。信息融合组件可以将局部和全局的上下文特征表示融合，以实现上下文感知的特征表示学习。

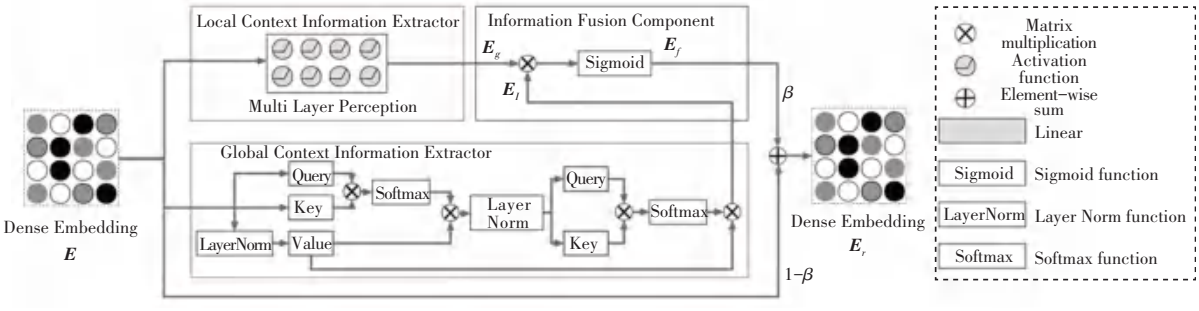

图 2 局部和全局上下文感知融合网络架构图

Fig. 2 Diagram of Local and Global context-aware Fusion Network architecture

2.2.4 上下文提取组件

由图 1 可知，上下文提取组件中的上分支全局上下文提取器旨在对输入序列中的全局上下文进行建模。设计中采用了改良了的交叉注意机制。第 1 个注意力是 1 个自注意力模块，自注意力模块首先计算所有特征对之间的重要性，并通过计算相关特征的加权和来生成新的表示。首先将输入矩阵 $\boldsymbol{E}$ 映射为 3 个不同的矩阵，并对最后一个矩阵做层归一化操作：

$$\boldsymbol{Q}_1, \boldsymbol{K}_1, \boldsymbol{V}_1 = \boldsymbol{E}\boldsymbol{W}^Q, \boldsymbol{E}\boldsymbol{W}^K, \text{LayerNorm}(\boldsymbol{W}^V) \quad (3)$$

其中，$\boldsymbol{Q}_1, \boldsymbol{K}_1, \boldsymbol{V}_1$ 分别表示注意力函数的输入；$\boldsymbol{W}^Q, \boldsymbol{W}^K, \boldsymbol{W}^V \in \mathbb{R}^{d \times d_k}$ 表示变换矩阵；d_k 表示注意力大小。然后通过应用 Query（$\boldsymbol{Q}$）和 Key（$\boldsymbol{K}$）的点积和 Softmax 函数，获得 Value（$\boldsymbol{V}$）上的注意力矩阵，如下所示：

$$\text{Attn}(\boldsymbol{Q}_1, \boldsymbol{K}_1, \boldsymbol{V}_1) = \text{Softmax}(\boldsymbol{Q}_1 \boldsymbol{K}_1^{\mathrm{T}}) \boldsymbol{V}_1 \in \mathbb{R}^{N \times d_k} \quad (4)$$

随后通过投影矩阵 $\boldsymbol{W}^P \in \mathbb{R}^{d_k \times d}$ 将输出矩阵的维数变换为与输入矩阵的维数相同。自注意模块的输出（$\boldsymbol{O}$）如下：

$$\boldsymbol{O} = \text{Attn}(\boldsymbol{Q}_1, \boldsymbol{K}_1, \boldsymbol{V}_1) \boldsymbol{W}^P \in \mathbb{R}^{N \times d} \quad (5)$$

交叉注意力的第 2 个注意力旨在从第 1 个注意力得到的高阶特征中挖掘全局上下文特征。具体是将输入矩阵 $\boldsymbol{O}$ 映射为 2 个不同的矩阵，并对输入矩阵 $\boldsymbol{O}$ 做层归一化操作：

$$\boldsymbol{Q}_2, \boldsymbol{K}_2, \boldsymbol{V}_2 = \text{LayerNorm}(\boldsymbol{O}), \text{LayerNorm}(\boldsymbol{O}), \boldsymbol{V}_1 \quad (6)$$

最后，与第 1 个注意力一样，通过应用 Query（$\boldsymbol{Q}$）和 Key（$\boldsymbol{K}$）的点积和 Softmax 函数，获得 Value（$\boldsymbol{V}$）上的注意力矩阵，再通过投影矩阵 $\boldsymbol{W}^P \in \mathbb{R}^{d_k \times d}$ 将输出矩阵的维数变换为与输入矩阵的维数相同。全局上下文嵌入特征 $\boldsymbol{E}_g$ 如下所示：

$$\boldsymbol{E}_g = \text{Attention}(\boldsymbol{Q}_2, \boldsymbol{K}_2, \boldsymbol{V}_2) \boldsymbol{W}^P \in \mathbb{R}^{N \times d} \quad (7)$$

交叉注意机制可以通过捕捉所有特征对之间的跨特征关系并对全局上下文关系进行挖掘，从而实现局部上下文感知的特征表示学习。然而，交叉注意只利用由成对特征交互表示的全局上下文信息，因此无法利用完整的上下文信息来指导特征细化。其数学表达如下：

$$\boldsymbol{h}_{i+1} = \text{ReLU}(\boldsymbol{W}_i \boldsymbol{h}_i + \boldsymbol{b}_i) \quad (8)$$

其中，$\boldsymbol{h}_i \in \mathbb{R}^{n_i}$，$\boldsymbol{h}_{i+1} \in \mathbb{R}^{n_{i+1}}$ 分别表示第 i 和第 $i+1$ 层隐藏层，并且 $\boldsymbol{h}_0 = \boldsymbol{E} \in \mathbb{R}^{1 \times (N \times d)}$，$\boldsymbol{W}_i \in \mathbb{R}^{n_{i+1} \times n_i}$；$\boldsymbol{b}_i$ 表示第 $i+1$ 层的可学习参数；ReLU（·）是 ReLU 函数。在最后一个隐藏层将局部上下文向量投影到嵌入大小 d，则局部上下文嵌入为：

$$\boldsymbol{E}_l = \text{ReLU}(\boldsymbol{W}_L \boldsymbol{h}_L + \boldsymbol{b}_L) \in \mathbb{R}^{1 \times d} \quad (9)$$

2.2.5 信息融合组件

在获得上下文信息 $\boldsymbol{E}_l$ 与 $\boldsymbol{E}_g$ 之后，使用 $\boldsymbol{E}_l$ 来加权特征表示 $\boldsymbol{E}_g$。其计算如下：

$$\boldsymbol{E}_f = \text{Sigmoid}(\boldsymbol{E}_l \odot \boldsymbol{E}_g) \in \mathbb{R}^{N \times d} \quad (10)$$

其中，“⊙”表示元素的乘积；$\boldsymbol{E}_l$ 是 MLP 模块的局部特征表示，可捕捉跨特征上下文关系；$\boldsymbol{E}_g$ 是使每个特征表示都能感知上下文信息的全局上下文信息。式（10）确保全局与局部上下文信息能够集成得到细化的融合上下文表示 $\boldsymbol{E}_f$。

此外，为了避免梯度消失与网络退化的问题，使用一个可训练的超参数 β，取值范围为 0 到 1 之间。将原始特征 $\boldsymbol{E}$ 与融合上下文特征 $\boldsymbol{E}_f$ 以残差连接的方式自适应集成，以生成最终的融合表示 $\boldsymbol{E}_r$。该计算流程可以被表示为：

$$\boldsymbol{E}_r = \beta \boldsymbol{E} + (1-\beta)\boldsymbol{E}_f \in \mathbb{R}^{N\times d} \tag{11}$$

总之，LGFNet 通过 3 个步骤生成上下文感知特征表示：

(1)使用上下文提取组件生成局部与全局上下文特征表示。

(2) 通过信息融合组件计算融合上下文信息。

(3)利用残差连接通过融合上下文信息表示和原始特征表示来生成上下文感知特征交互。

2.2.6　特征交叉层

为了学习用于推荐的线性(一阶)以及成对(二阶)的特征交互，该层使用了 FM 的特征交叉层。与传统的简单线性模型不同的是，FM 考虑了特征间的交叉，对所有嵌套变量交互进行建模。因此在即使数据非常稀疏的情况下，依然能估计出可靠的参数进行预测。由图 1 可知，特征交叉层的输出是加法单元和许多内积单元的总和：

$$Y = \sum_{i=1}^{n} \omega_i x_i + \sum_{i=1}^{n}\sum_{j=i+1}^{n} \langle V_i, V_j \rangle x_i x_j \tag{12}$$

其中，$\omega_i \in \mathbb{R}^N$，$V_i \in \mathbb{R}^k$，$k \in \mathbb{N}_0^+$ 表示定义因子分解维数的超参数。

2.2.7　输出层

为了将点击率预测映射为一个二分类问题，设计了以对数损失函数为核心的预测层用来计算模型的 loss。数学定义公式为：

$$\hat{Y} = \mathrm{Sigmoid}(Y) \tag{13}$$

$$\mathrm{loss} = -\frac{1}{N}\sum_{i=1}^{N} Y_i \log(\mathrm{Sigmoid}(\hat{Y}_i)) + (1-Y_i)\log(1-\mathrm{Sigmoid}(\hat{Y}_i)) \tag{14}$$

其中，N 表示训练实例的总数。

3　实验

在本节中，进行了实验，以回答以下研究问题：

问题 1：与最先进的模型相比，该模型的性能与效率如何？

问题 2：挖掘局部和全局上下文信息是否对模型的性能有积极影响？

问题 3：信息融合模块是否是必要的？学习融合上下文信息对模型的性能有什么影响？

问题 4：学习组合因子 β 的效果如何？与固定组合因子 β 的值相比哪种方法更加合理？

3.1　实验设置

3.1.1　数据集

本文使用 2 个公开可用的数据集：Criteo 和 Frappe。这 2 个数据集的统计数据汇总在表 1 中。

表 1　数据集统计

Table 1　Statistics of the dataset

数据集	训练集	验证集	测试集	特征
Criteo	35 840 617	5 000 000	5 000 000	1 086 810
Frappe	202 027	57 722	28 860	5 382

Criteo 是点击率预测最著名的行业基准数据集，包含 Criteo7 天内的一部分流量，每条记录对应一个由 Criteo 提供的展示广告，包括 26 个类别型特征和 13 个数字型特征。在数据预处理阶段，过滤了出现不到 10 次的特征，测试集使用最后 500 万条记录进行测试。

Frappe 数据集包含用户在不同上下文时的应用程序使用日志，一共包括 96 203 个应用程序。每条日志包含 8 个上下文特征(如天气、城市、时间等)，采用独热向量编码后特征有 5 382 维，目标值表示用户是否在上下文下使用了应用程序。

3.1.2　基线

本文将 LGFNet 应用于 FM，称为 $\mathrm{FM}_{\mathrm{LGFNet}}$。

本文使用了 3 种类型的方法比较 $\mathrm{FM}_{\mathrm{LGFNet}}$：

(1)基于 FM 的方法。捕捉二阶或高阶特征交互，包括 FM[9]、IFM[11]、DIFM[10]。

(2)基于深度学习的方法。对高阶特征交互进行建模，包括 FINT[7]、NFM[20]、IPNN[25]、OPNN[25]。

(3)集成方法。采用多塔特征交互结构来集成不同类型的方法，包括 DCN-V2[1]、AFN[3]、TFNET[5]、FED[6]、xDeepFM[12]、AutoInt[13]、NON[14]、WDL[17]、DeepFM[18]、DCN[26]和 FiBiNET[27]。

3.1.3　评估指标

为了评估点击率预测方法的性能，采用 ROC 曲线下面积(AUC)和二进制交叉熵损失(Logloss)作为评估指标。注意，稍高的 AUC 或较低的 Logloss，例如在 0.001 水平，可以被视为点击率预测任务的显著改善。

3.1.4　实验细节

为了确保比较的公平性，本文通过改变随机种子来运行所有实验 5 次，并记录平均值。通过实验观察到，$\mathrm{FM}_{\mathrm{LGFNet}}$ 的所有标准偏差都在 1e-4 的数量级，这表明结果非常稳定。

3.2 实验结果比较

3.2.1 推荐性能比较

表 2 总结了 LGFNet 的有效性以及 2 个数据集上所有比较方法。尽管 FM[9] 具有最差的性能,但 FM_{LGFNet} 在推荐性能上显著优于所有比较方法。在 2 个数据集上,AUC 分别比 FM[9] 高 1.15%、1.86%(Logloss 分别为 1.99%、2.36%),这表明学习上下文感知融合特征表示在 CTR 预测中是有效的。同时,LGFNet 在 2 个数据集上都具有最好的平均性能提升(AUC 和 Logloss)。表 2 表明,通过 LGFNet 学习上下文感知融合特征表示比其他特征交互技术更有效,例如 xDeepFM、NON 和 DCN-V2 中的特征交互技术。

表 2 2 个数据集的总体准确性比较

Table 2 Comparison of overall accuracy of two datasets

算法	Criteo		Frappe	
	AUC	Logloss	AUC	Logloss
FM[9]	0.802 8	0.451 4	0.970 8	0.193 4
IFM[11]	0.806 6	0.447 0	0.976 5	0.189 6
DIFM[10]	0.808 5	0.445 7	0.978 8	0.186 0
NFM[20]	0.805 7	0.448 3	0.974 6	0.191 5
IPNN[25]	0.808 8	0.445 4	0.979 1	0.175 9
OPNN[25]	0.809 6	0.444 6	0.979 5	0.180 5
CIN	0.808 2	0.445 9	0.977 6	0.201 0
FINT[7]	0.809 0	0.445 2	0.979 1	0.192 1
WDL[17]	0.806 8	0.447 4	0.977 6	0.189 5
DCN[26]	0.809 1	0.445 2	0.978 9	0.181 4
FiBiNET[27]	0.809 3	0.445 0	0.978 7	0.186 7
DeepFM[18]	0.808 4	0.445 8	0.978 9	0.177 0
xDeepFM[12]	0.808 6	0.445 6	0.979 2	0.188 9
AutoInt[13]	0.808 8	0.445 6	0.978 6	0.189 0
AFN[3]	0.809 5	0.444 7	0.979 1	0.182 4
NON[14]	0.809 6	0.444 6	0.979 2	0.181 3
TFNET[5]	0.809 2	0.444 9	0.978 7	0.194 2
FED[6]	0.808 7	0.445 8	0.979 7	0.180 2
DCN-V2[1]	0.809 8	0.444 3	0.980 2	0.178 3
FM_{LGFNet}	**0.812 0**	**0.442 4**	**0.983 0**	**0.160 7**

3.2.2 推荐效率比较

图 3 中比较了不同方法的模型大小和运行时间。通常,基于 FM 的方法比基于深度学习方法或集成的方法具有更少的参数。与 FM_{LGFNet} 相比,DIFM[10] 和 xDeepFM[12] 仅比 FM[9] 增加了 104 K 的学习参数,并且 DIFM[10] 和 xDeepFM[12] 分别比 FM 增加 266 K 和 483 K 的学习系数。同时,因为 DIFM 和 CIN 由复杂的结构组成,两者相对 FM_{LGFNet} 来说更加耗时。并且从图 3 中观察到 FM_{LGFNet} 与 IFM[11] 和 DCN[26] 的耗时量相当,还具有更少的模型参数,并且比所有其他基线方法更有效。值得注意的是与表现最好的基线 DCN-V2[1] 相比,FM_{LGFNet} 具有更少的模型参数、更快的训练速度和更好的性能。

3.3 超参数研究

为了分析 LGFNet 中隐藏层的数量与注意力大小对模型性能的影响,使用网格搜索法进行实验,实验结果如图 4、图 5 所示。

(1)隐藏层的数量。图 4 和图 5 显示了 MLP 模块中隐藏层数量对推荐性能的影响。对于 Criteo 和 Frappe,最合适的隐藏层数是 1。这证实了上下文信息不是很复杂,并且使用浅层 MLP 已经足够对来自每个实例的上下文信息进行编码。

(2)注意力大小。由图 4、图 5 可知,Criteo 和 Frappe 的最佳注意力大小分别为 10 和 20。

3.4 消融实验

本文在 Criteo 和 Frappe 上进行了实验,以证明 LGFNet 中的每个组件或设计在提高 CTR 预估的性能方面起着至关重要的作用。消融研究结果见表 3。表 3 中,$\boldsymbol{E}$、$\boldsymbol{E}_l$、$\boldsymbol{E}_g$、$\boldsymbol{E}_f$ 和 FM_{LGFNet} 分别表示原始的、局部上下文的、全局上下文的、融合上下文的和最终的特征表示。β 为权重结合超参数。由表 3 可知,使用方程来描述如何在 $\boldsymbol{E}$ 的基础上通过移除或更换 LGFNet 中的一个组件计算 $\boldsymbol{E}_r$。

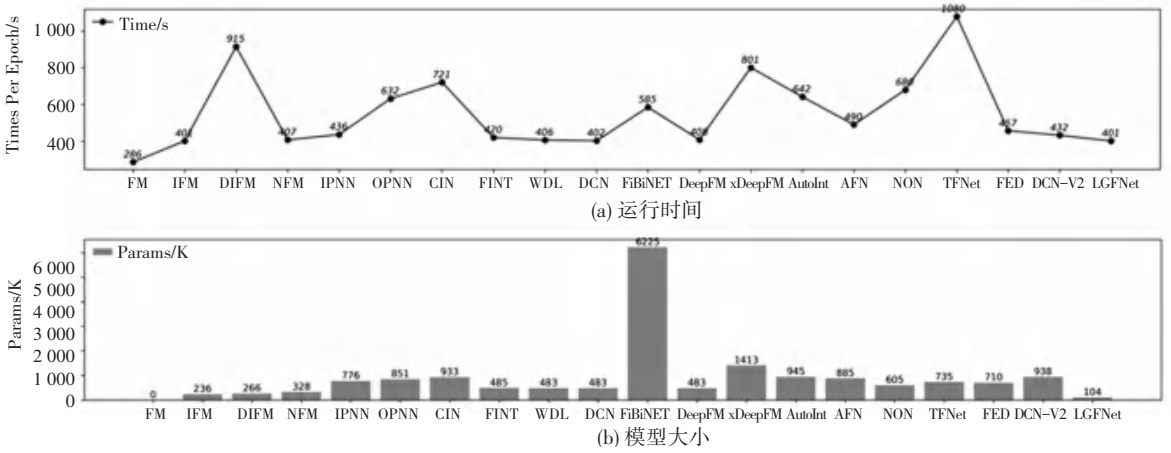

图 3 在 Criteo 数据集上不同算法在模型大小和运行时间方面的效率比较

Fig. 3 Comparison of the efficiency of different algorithms in terms of model size and running time on the Criteo dataset

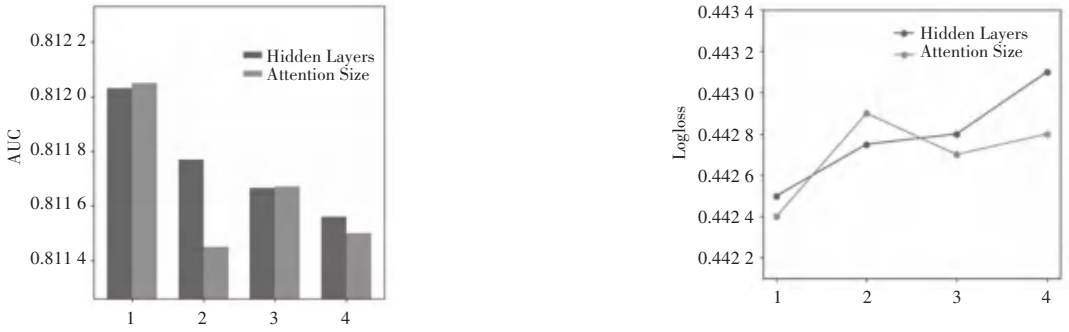

图 4　在 Criteo 数据集上隐藏层和注意力大小对模型性能的影响

Fig. 4　The impact of hidden layer size and attention size on model performance in the Criteo dataset

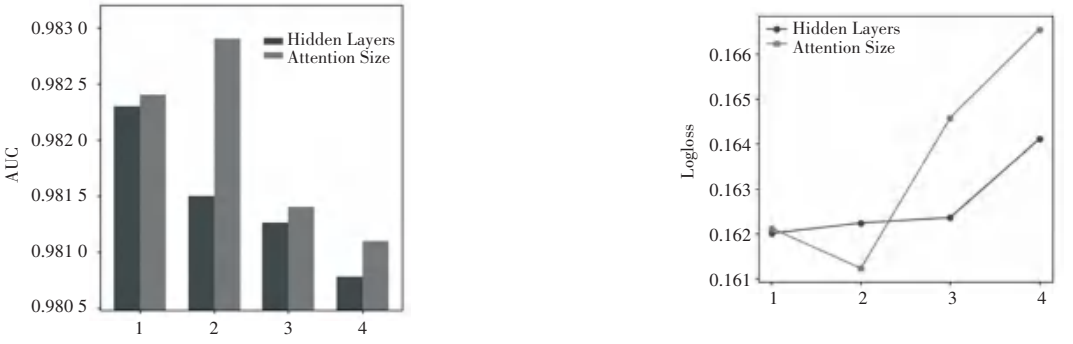

图 5　在 Frappe 数据集上隐藏层和注意力大小对模型性能的影响

Fig. 5　The impact of hidden layer size and attention size on model performance in the Frappe dataset

表 3　消融研究

Table 3　Ablation research

$\boldsymbol{E}_r$	变量	Criteo		Frappe	
		AUC	Logloss	AUC	Logloss
FM($\boldsymbol{E}$)	#1	0.802 8	0.451 4	0.970 8	0.193 4
$\beta\boldsymbol{E}+(1-\beta)\boldsymbol{E}_l$	#2	0.805 6	0.448 3	0.971 7	0.191 2
$\beta\boldsymbol{E}+(1-\beta)\boldsymbol{E}_g$	#3	0.807 1	0.447 0	0.974 4	0.189 7
$\beta\boldsymbol{E}+(1-\beta)(\boldsymbol{E}_g+\boldsymbol{E}_l)$	#4	0.807 3	0.446 8	0.975 4	0.187 8
$\boldsymbol{E}_f$	#5	0.809 0	0.445 2	0.977 8	0.182 1
$\boldsymbol{E}_g$	#6	0.811 0	0.444 3	0.979 3	0.171 3
$\boldsymbol{E}_l$	#7	0.811 3	0.443 7	0.979 7	0.169 7
$\text{FM}_{\text{LGFNet}}$	#8	**0.812 0**	**0.442 4**	**0.983 0**	**0.160 7**

3.4.1　学习上下文表示的效果

本节研究了局部与全局上下文提取模块与上下文信息融合模块在 LGFNet 中的效果，对于局部与全局上下文提取模块，变量#2 与变量#3 分别表达了上下文提取组件中只保留局部或全局上下文信息提取模块的嵌入。LGFNet 的上下文提取模块只保留了局部或全局上下文提取器，并屏蔽了上下文信息融合模块。学习超参数 β 对模型性能的影响见表 4。表 4 中的结果表明，学习局部和全局上下文特征是对提升模型性能有帮助的。对于上下文信息融合模块，变量#4 表达了在 LGFNet 中屏蔽了信息融合组件，只是简单地将局部与全局上下文嵌入相加的效果。根据变量#4 与变量#8 的结果表明，使用 IFC 学习上下文融合特征是合理的。

表 4　学习超参数 β 对模型性能的影响

Table 4　Impact of learning hyperparameter β on model performance

模型	Frappe		Criteo	
	AUC	Logloss	AUC	Logloss
$\text{FM}_{\text{LGFNet}}$	0.985 3	0.140 8	0.956 5	0.284 0
$\text{FM}_{\text{LGFNet}}$ With β_L	**0.985 6**	**0.139 5**	**0.959 0**	**0.283 3**

3.4.2　组合超参数 β 的影响

为了避免梯度消失与网络退化，LGFNet 被设计成了 2 个并行分支，在得到上下文融合特征嵌入后，将其与原始特征 $\boldsymbol{E}$ 加权相结合，因此，每个分支都有助于最终输出概率。图 6 表明了 β 的变化带来的影响，β 在 0 到 1 之间变化，其中 β = 0 为只保留融合上下文嵌入式(变量#5)，β = 1 为只保留原始特征嵌入(变量#1)。研究可知，β 对于不同数据集的值不同，多头自注意分支对最终输出的贡献更高，特别是对于 Frappe 和 MovieLens 数据集。

此外，图 6 显示了改变 2 组嵌入组合的权重对 Logloss 的影响。对于 Frappe 数据集，对数损失的差异在 0.2～0.6 之间是最小的。相反，最佳 Logloss

是在 0.6 时获得的,对于 MovieLens 数据集,该模型在 0.8 时获得的 Logloss 最好。

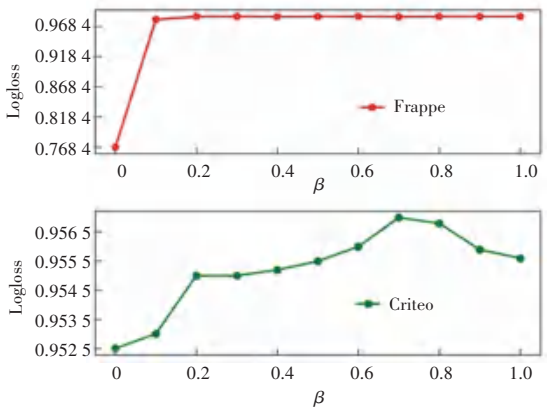

图 6 更改 β 的值对模型性能的影响

Fig. 6 The impact of changing the value of β on model performance

3.4.3 学习组合超参数 β 的影响

上一节展示了上下文融合特征嵌入 $\boldsymbol{E}_f$ 与原始特征 $\boldsymbol{E}$ 权重的改变对最终性能的影响的消融研究。这里通过调整每个组件的贡献权重来了解每个部件是如何影响模型性能,提出一种方法是通过学习组合超参数以及模型的参数来优化目标函数。首先初始化权重 β 为 0.5,并将其设置为可与模型参数同时训练。表 4 说明了使用固定组合因子 β 与学习组合因子 β_L 所获得的结果。由表 4 可知,可学习权重对 Frappe 数据集有积极影响;图 7 展示了在 2 个数据集中 β 系数随着 epoch 的变化;对于 Frappe 数据集,β 超参数稳步上升至 0.66,而对于 LastFM 数据集,β 超参数从 0.5 略微下降至 0.44。

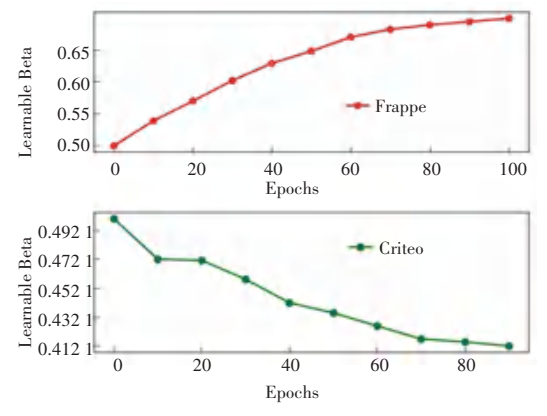

图 7 可学习的 β 值随着训练轮数的变化

Fig. 7 The learnable β value changes with the number of training epochs

4 结束语

本文提出了一个局部和全局上下文感知融合网络(LGFNet)的模型,可以挖掘局部与全局上下文特征表示,并将其融合获得完整的上下文表示,以提高其性能。$\text{FM}_{\text{LGFNet}}$ 设计了上下文提取组件来挖掘局部和全局上下文信息,使模型能充分使用局部与全局上下文信息来指导特征细化。此外还设计了信息融合组件,让局部和全局上下文特征表示融合,再将原始特征和融合上下文的特征表示相集成。详细的消融研究表明,LGFNet 的每一种设计都有助于提高整体性能。此外,综合实验验证了 LGFNet 的有效性和高效性。未来的工作将会聚焦于进一步挖掘用户的潜在表示,提升模型的性能。

参考文献

[1] WANG Ruoxi, SHIVANNA R, Cheng D, et al. DCN v2: Improved deep & cross network and practical lessons for web-scale learning to rank systems[C]//Proceedings of the Web Conference. New York:ACM,2021: 1785-1797.

[2] ZHOU Guorui, SONG Chengru, ZHU Xiaoqiang, et al. Deep interest network for click-through rate prediction[C]// Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM, 2018: 1059-1068.

[3] CHENG Weiyu, SHEN Yanyan, HUANG Linpeng. Adaptive factorization network: Learning adaptive-order feature interactions[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(4): 3609-3616.

[4] LIU Bin, TANG Ruiming, CHEN Yingzhi, et al. Feature generation by convolutional neural network for click-through rate prediction[J]. arXiv preprint arXiv,1904.04447,2019.

[5] WU Shu, YU Feng, YU Xueli, et al. Tfnet: Multi-semantic feature interaction for ctr prediction[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York:ACM, 2020: 1885-1888.

[6] ZHAO Zihao, FANG Zhiwei, LI Yong, et al. Dimension relation modeling for click-through rate prediction[C]//Proceedings of the 29th ACM International Conference on Information & Knowledge Management. New York:ACM, 2020: 2333-2336.

[7] ZHAO Zhishan, YANG Sen, LIU Guohui, et al. FINT: Field-aware interaction neural network for CTR prediction[J]. arXiv preprint arXiv,2107.01999, 2021.

[8] JUAN Y, ZHUANG Yong, CHIN W S, et al. Field-aware factorization machines for CTR prediction[C]//Proceedings of the 10th ACM Conference on Recommender Systems. New York: ACM, 2016: 43-50.

[9] RENDLE S. Factorization machines[C]// Proceedings of 2010 IEEE International Conference on Data Mining. Piscataway, NJ: IEEE, 2010: 995-1000.

[10] LU Wantong, YU Yantao, CHANG Yongzhe, et al. A dual input-aware factorization machine for CTR prediction[C]// Proceedings of the Twenty-ninth International Joint Conferences on Artificial Intelligence. TAL, 2021: 3139-3145.

[11] YU Yantao, WANG Zhen, YUAN Bo. An input-aware factorization machine for sparse prediction[C]// Proceedings of

the Twenty－ninth International Joint Conferences on Artificial Intelligence. New York：ACM，2019：1466－1472.

[12] LIAN Jianxun，ZHOU Xiaohuan，ZHANG Fuzheng，et al. xdeepfm：Combining explicit and implicit feature interactions for recommender systems [C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York：ACM，2018：1754－1763.

[13] SONG Weiping，SHI Chence，XIAO Zhiping，et al. Autoint：Automatic feature interaction learning via self－attentive neural networks [C]//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. New York：ACM，2019：1161－1170.

[14] LUO Yuanfei，ZHOU Hao，TU Weiwei，et al. Network on network for tabular data classification in real－world applications [C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York：ACM，2020：2317－2326.

[15] PAN Junwei，XU Jian，RUIZ A L，et al. Field－weighted factorization machines for click－through rate prediction in display advertising [C]//Proceedings of the 2018 World Wide Web Conference. New York：ACM，2018：1349－1357.

[16] RENDLE S. Factorization machines with libFM [J]. ACM Transactions on Intelligent Systems and Technology (TIST)，2012，3(3)：57.

[17] CHENG H T，KOC L，HARMSEN J，et al. Wide & deep learning for recommender systems [C]//Proceedings of the 1st Workshop on Deep Learning for Recommender Systems. New York：ACM，2016：7－10.

[18] GUO Huifeng，TANG Ruiming，YE Yunming，et al. DeepFM：A factorization－machine based neural network for CTR prediction [J]. arXiv preprint arXiv，1703.04247，2017.

[19] VASWANI A，SHAZEER N，PARMAR N，et al. Attention is all you need [C]//Advances in Neural Information Processing Systems. Long Beach，UAS：NIPS Foundation，2017：5998－6008.

[20] HE Xiangnan，CHUA T S. Neural factorization machines for sparse predictive analytics [C]//Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York：ACM，2017：355－364.

[21] GULATI A，QIN J，CHIU C C，et al. Conformer：Convolution－augmented transformer for speech recognition [J]. arXiv preprint arXiv，2005.08100v1，2020.

[22] GUO Pengcheng，BOYER F，CHANG Xuankai，et al. Recent developments on espnet toolkit boosted by conformer [C]//Proceedings of 2021 IEEE International Conference on Acoustics，Speech and Signal Processing (ICASSP). Piscataway，NJ：IEEE，2021：5874－5878.

[23] PENG Yifan，DALMIA S，LANE I，et al. Branchformer：Parallel mlp－attention architectures to capture local and global context for speech recognition and understanding [J]. arXiv preprint arXiv，2207.02971，2022.

[24] WU Zhanghao，LIU Zhijian，LIN Ji，et al. Lite transformer with long－short range attention [J]. arXiv preprint arXiv，2004.11886，2020.

[25] QU Yanru，FANG Bohui，ZHANG Weinan，et al. Product－based neural networks for user response prediction over multi－field categorical data [J]. ACM Transactions on Information Systems，2018，37(1)：5.

[26] WANG Ruoxi，FU Bin，FU Gang，et al. Deep & cross network for ad click predictions [J]. arXiv preprint arXiv，1708.05123，2017.

[27] HUANG Tongwen，ZHANG Zhiqi，ZHANG Junlin. FiBiNET：Combining feature importance and bilinear feature interaction for click－through rate prediction [C]//Proceedings of the 13th ACM Conference on Recommender Systems. New York：ACM，2019：169－177.