25 | DeepFM：如何使用深度学习技术给数据排序?

2023-06-12 黄鸿波 来自北京

《手把手带你搭建推荐系统》

你好，我是黄鸿波。

上节课里我们讲了非常经典的 GBDT+LR 模型，虽说 GBDT+LR 的组合能够解决很大一部分问题，但是对于高阶的特征组合仍然缺乏良好的应对能力。因此就迎来了本节课要学习的内容：DeepFM。我会先从 FM 的概念入手，然后进一步讲解 DeepFM 的模型结构。

FM 算法概述

在讲解 DeepFM 之前，我们先来了解一下 FM。FM（Factorization Machines，因子分解机）是广义线性模型（GLM）的变种，是一种基于矩阵分解的机器学习算法。

当我们面对推荐系统时，数据可能是一个高维的稀疏矩阵。我们希望从这个矩阵中提取出有用的特征，并用这些特征来进行预测和推荐。常见的做法是使用矩阵分解算法（例如 SVD 和

ALS），但这种算法的计算复杂度很高，不太适用于大规模的数据集。为了解决这个问题，可以使用 FM 算法。

在一般的模型下，各个特征之间都是独立的（例如年龄和文章类别、性别和文章长度等），我们并没有考虑到特征与特征之间的相互关联。但是在实际的推荐系统中，大量的特征之间是有关联关系的，放大了来说，就是内容画像之间的各个特征其实相互关联，内容画像与用户画像之间也存在着很大的关联性。如果能够把这些特征与特征之间的关联性找到，然后挖掘其背后的关系，显然对于推荐系统来说非常有意义。

FM 算法的核心思想是通过对每个特征（或每对特征之间）建模，捕捉特征的相互影响。FM 算法将每个特征表示为一个因子向量，并通过计算这些因子向量的内积来计算特征之间的交互作用，我们可以用下面的数学公式来表示它。

$$\hat{y}(x) = w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \omega_{ij} x_i x_j$$

在这段公式中，x_i 代表特征，w_i 代表该特征的权重，n 代表样本的特征个数，特征相关的参数一共有 n(n-1)/2 个。

在实际业务中怎么用 FM 来解决问题呢？我来举个例子。

假如在信息流推荐系统中有 1000 篇文章，用户 A 只浏览过其中的 5 篇文章，剩下的 995 篇文章没有浏览过。

如果基于用户 A 的历史记录来做成一个内容矩阵，那么这里面的 995 个元素都将为 0，这里实际上就缠上了一个高维稀疏问题。

引入 FM 的优势就是处理这些高维稀疏问题。具体来说，FM 算法处理高维稀疏问题的流程如下。

1. 对于原始输入特征矩阵，进行 One-Hot 编码，将其变为一个列数等于特征数量，行数等于样本数量的矩阵。
2. 对原始输入特征矩阵进行降维，将其转化为一个低维度的隐向量矩阵，对于稀疏数据，可以用如 SVD、PCA 等降维方法得到低维度的表示，对于非稀疏数据，可以直接使用 Embedding 方法进行降维。
3. 在得到隐向量矩阵之后，使用它来代替原始的输入特征向量，再将 FM 算法应用于处理。
4. 根据隐向量矩阵求出特征交互部分的系数，计算出预测结果。

FM 算法通过隐向量的方式，将高维稀疏的特征进行了降维，使得算法可以处理更高维度的数据，并且在处理高维度数据时仍然可以保持较高的准确性。

DeepFM 模型结构

了解了 FM 后，接下来我们进入 DeepFM 的讲解。

DeepFM 是一种基于深度学习和 FM 模型的混合模型，它包含两个部分，一部分是 FM 模型，用来捕捉特征之间的交互效应；另一部分是一个深度神经网络，用来学习更高阶的特征交互和特征表达。

由于 DeepFM 算法有效地结合了 FM 和神经网络算法的优点，能够同时对低阶组合特征和高阶组合特征进行提取，因此被广泛使用。

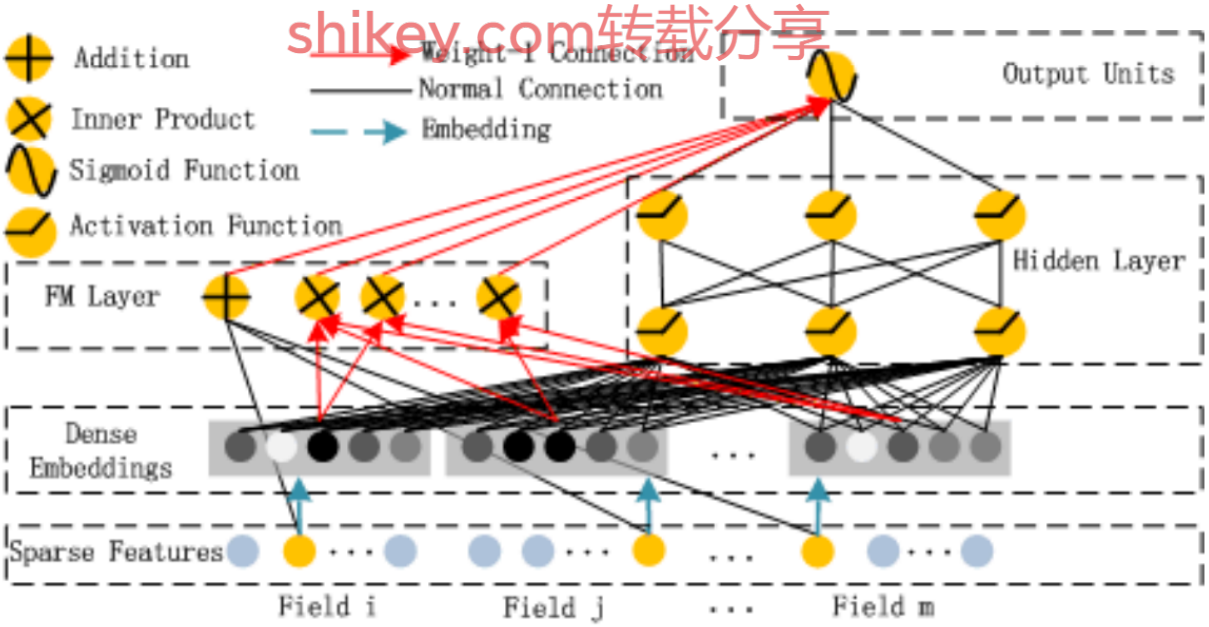
在 DeepFM 中，FM 部分用来处理低阶特征交互，包括一阶特征和二阶特征交互。FM 模型能够捕捉到特征之间的线性关系和二次关系，对于数据稀疏的情况，FM 模型以几乎线性的时间和空间复杂度快速地学习特征交互，这对于处理高维离散特征非常有优势。

而 DNN 部分用来处理高阶特征交互，包括三阶特征和更高阶的特征交互。DNN 对输入的 Embedding 向量进行深度处理，从而能够挖掘到更为复杂的高阶交互模式。同时，DNN 可以对未知的特征进行插值和泛化，因此在数据不完整的情况下也能够保持较好的预测准确率。

DeepFM 通过 FM 模型和 DNN 模型的结合，充分发挥了二者的优势，得到了更好的性能。对于大规模数据集、高维度特征和数据稀疏性较高的情况，DeepFM 能够取得更好的预测效

果。

接下来我们来看一下 DeepFM 的模型结构。



在这个模型中从下往上看，一共分成五个部分，分别是 Sparse Features、Dense Embeddings、FM Layer、Hidden Layer 以及 Output Units。

我们来分别解释下这五个部分。

Sparse Features

这一层的主要作用是对特征进行处理。在做特征处理时，需要对每一个离散型的数据来做 One-Hot 编码。经过 One-Hot 编码后，一个特征会用很多列来表示，这时整个特征就会变得非常稀疏。因此就需要去记录 One-Hot 之前的特征（即图中的 Field），我们可以暂且理解为原始输入的特征。这样做就是为了在存储矩阵时，可以把一个大的稀疏矩阵转换成两个小的矩阵和一个字典进行存储。

Sparse Features 层将每个类别特征映射为一个唯一的 ID（整数），然后将这些离散表示的类别特征转换为稀疏矩阵，其中每行表示一个样本，每列表示一个类别特征的 ID。同时，该层

还会将每个类别特征 ID 嵌入到一个低维向量空间中，使得每个类别特征 ID 对应一个稠密的低维向量。这个嵌入矩阵可以被训练到，使得每个特征向量能够更好地被表示。

通过 Sparse Features 层处理后的数据，可以输入到 DeepFM 的后续层中进行进一步的特征交叉和深度学习。在 CTR 预估问题中，Sparse Features 层通常是非常重要的一层，因为离散型类别特征通常对 CTR 的预测有着非常重要的影响。我们可以把 Sparse Features 这一层看作是经过 One-Hot 编码的类别特征与数值特征的拼接。

Dense Embeddings

DeepFM 模型中的 Dense Embeddings 层（稠密嵌入层）用于将稀疏高维的数据压缩成稠密的实数向量表示。

我们在上一步（也就是通过 Sparse Features 层的处理后）得到的是一个高维度的稀疏向量，这个高维度的稀疏向量实际上在计算时无法得到特征之间的相互关系。那么我们就可以通过 Dense Embeddings 层，将稀疏的 01 向量做一个 Embedding，将其转化成低维稠密的向量。

根据上面的结构图可以看到，实际上每一个高维稀疏向量都有自己所对应的 Embedding 向量，不同的向量之间的 Embedding 实际上是相互独立的，我们把每一个稠密向量进行横向的拼接，使其变成一个长度很长的稠密向量，然后再拼接上原始的数值特征，统一作为 Deep 与 FM 的输入。

一般来说在 DeepFM 模型中，Sparse Features 层和 Dense Embeddings 层是紧密结合的。我们所做的就是提取稀疏特征，再划分 Field 做 Dense Embeddings，将这些原始的特征转化为低维稠密的向量，方便 input 到之后的模型中。而这些 input 共同构成了一个特征嵌入的整体，旨在为数据提供更好的特征表达能力和更强的预测性能。

FM Layer

DeepFM 中的 FM 层由线性部分和交叉部分两部分组成。

线性部分指输入特征的线性组合，可以类比于 LR 模型的线性部分。它可以得到输入特征的权重，从而学习输入特征的重要程度。

交叉部分是指捕获二阶交互特征的部分，它会将所有特征向量的每个元素分别相乘，得到所有可能的二阶交互项，然后，对这些二阶交互项进行加权求和，得到交叉部分的输出。

值得注意的是，FM 层的线性和交叉部分可以共享特征向量的 Embedding 参数，有助于节省模型的参数量，提高模型训练效率。

FM 部分的模型结构如下。

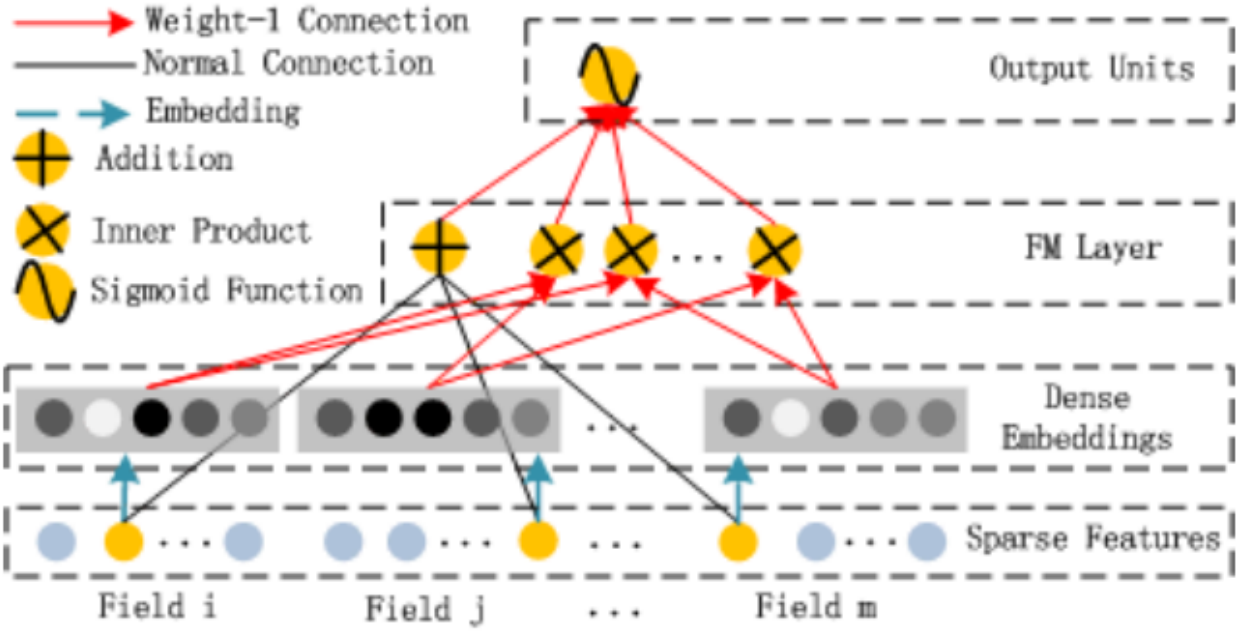


Figure 2: The architecture of FM.

Hidden Layer

在 DeepFM 中，Hidden Layer 是一个简单的前馈神经网络，由于原始特征向量中大多数都是高维度的稀疏向量、连续的特征和类别特征混合。为了能够更好地发挥 DNN 模型的特性，设计了一套子网络来将原始的系数特征转换成稠密的特征向量，也就是下面这张图中的部分。

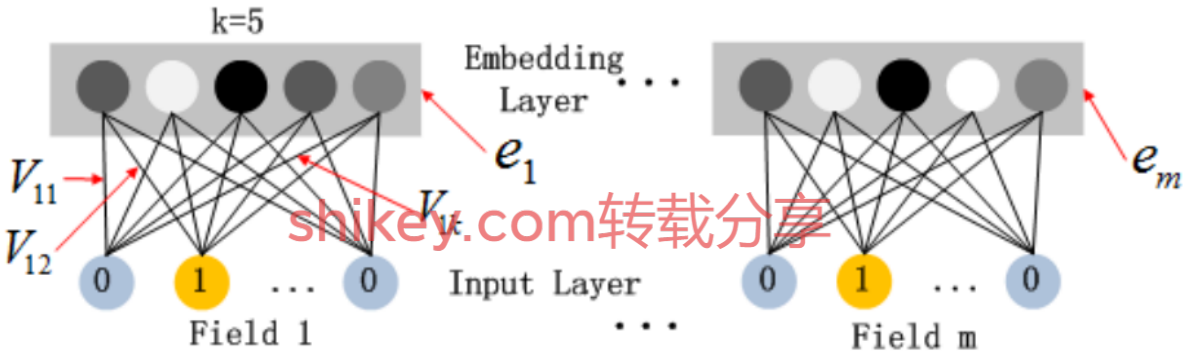


Figure 4: The structure of the embedding layer

根据论文里的介绍，这一部分的设计主要是针对离散的特征，首先经过 Embedding，然后将得到的 Embedding 进行拼接，形成一个新的向量，然后通过 DNN 学习类别之间的隐藏特征交叉，然后再输出 logits 值。这里实际上就会将稠密的特征和稀疏特征使用全连接的方式，输入到 Hidden Layer 中。这样做能够很好地解决参数爆炸的问题，也是推荐模型中的常见处理方法。

Output Units

Output Units 实际上就是将 FM 的预训练向量 V 作为网络权重，初始化替换为 FM 和 DNN 进行联合训练，从而得到一个端到端的模型。

在这一层中，会对 FM Layer 的结果与 Hidden Layer 的结果进行累加，这样做的好处是将低阶与高阶的特征交互融合，然后将得到的结果进行一个 sigmoid 操作，得到预测的概率输出。

在论文中，输出层的公式如下，实际上做的就是上面所提到的步骤。

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

DeepFM 的优缺点

到目前为止，我们已经对 DeepFM 有了一个比较深入的了解，下面一块看下 DeepFM 的优缺点。

DeepFM 的优点。

1. 考虑了高阶交叉特征：DeepFM 可以对各种交叉特征进行编码，包括高阶交叉特征，从而提高了模型的表达能力。
2. 既考虑了线性特征又考虑了非线性特征：DeepFM 同时考虑了线性模型和因子分解机模型，可以对线性特征和非线性特征进行学习和推断。
3. 适用于稀疏特征：DeepFM 可以处理具有稀疏结构的特征，例如在推荐系统中常见的用户 - 物品交互。
4. 可以通过高效的 Embedding 实现对离散特征的编码：DeepFM 基于 Embedding 层实现了对离散特征的编码，这种方法可以高效地处理海量的离散特征。

DeepFM 自身的缺点。

1. 模型训练较慢：DeepFM 中的深度模块导致了训练过程的时间和计算复杂度的增加。
2. 特征选取和处理的要求较高：DeepFM 需要对原始数据进行一定的预处理，同时对于不同的数据集，需要结合实际场景设计合适的特征。
3. 对于连续特征的处理较为有限：DeepFM 采用了 Embedding 层来处理离散特征，虽然它也支持连续特征，但是处理连续特征的方法较为简单并且直接。

DeepFM	
优点	缺点
考虑了高阶交叉特征	模型训练较慢
线性特征和非线性特征均考虑在内	特征选取和处理的要求较高
适用于稀疏特征	对连续特征的处理有限
可通过高效 Embedding 实现对离散特征的编码	调参难度大

总结

到这里，本节课的内容就结束了，今天主要给你在理论层面上介绍了 DeepFM 模型，下面我们对这节课的内容做一个简单总结。

1. DeepFM 是一种组合了因子分解机 (FM) 和神经网络 (NN) 的混合模型，旨在同时利用 FM 和 DNN 处理高维稀疏特征和连续特征。它的目标是对于二分类和回归问题，提高预测准确率和模型效率。FM 能够很好地处理特征之间的交互，而 DNN 可以更好地拟合高阶特征的非线性关系。
2. DeepFM 模型一共分为了 Sparse Features、Dense Embeddings、FM Layer、Hidden Layer 以及 Output Units 五个层，你应该熟悉每层的作用。
3. DeepFM 模型优势和劣势都很明显，在使用的过程中需要全方位考量。

课后题

今天只有一道课后题：尝试进一步了解 DeepFM，并使用一个开源数据集来实现一个 DeepFM 模型。

期待你的作业分享，如果你觉得这节课对你有帮助，也欢迎分享给有需要的朋友。

精选留言 (3)

G小调

2023-07-04 来自云南

老师，你代码，在github上哪里

曾超

2023-06-22 来自北京

希望老师每节课都能给到参考代码，比如这节课，太理论没有代码不好吸收。

曾超

2023-06-16 来自北京

老师没有参考的代码么