Deep & Cross Network理论与实践

# 背景

特征工程是很多预测任务成功与否的关键，然而其过程耗时耗力。DNN依靠神经网络强大的学习能力，可以在一定程度上实现自动学习特征组合,但是很难学到所有的组合特征。另外，DNN隐式学习的组合特征具有不可解释性，且效率极低。Deep & Cross Network (以下简称DCN) 模型，在保留了DNN的优势的基础上加入了cross network，使其能够更高效地学习特征交互。DCN在每一层中都加入了特征交叉，无需人工特征工程。另外，相比DNN，DCN的复杂度提升微不足道。

# 相关工作

下面这个相关工作总结的很好，做下记录。

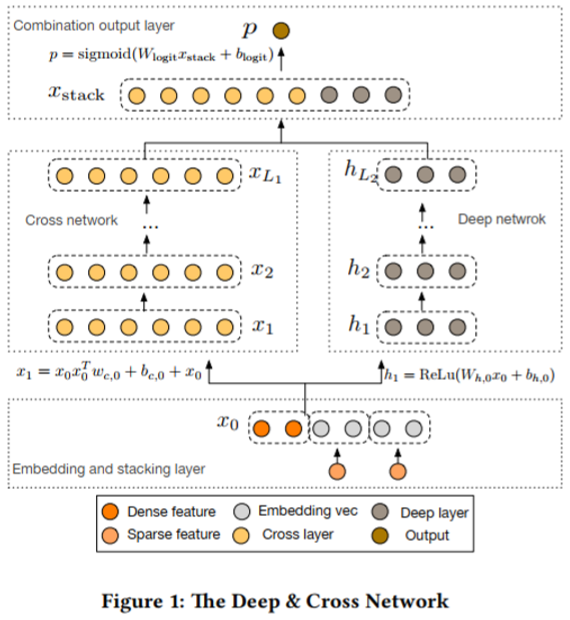
最开始的FM模型使用隐向量的内积来建模组合特征；FFM在此基础上引入field的概念，针对不同的field上使用不同隐向量。但是，这两者都是针对低阶的特征组合进行建模的。随着DNN在计算机视觉、自然语言处理、语音识别等领域取得重要进展，DNN几乎无限 的表达能力被广泛的研究。同样也尝试被用来解决web产品中输入高维高稀疏的问题。 DNN可以对高维组合特征进行建模，但DNN是否就是针对此类问题最高效的建模方式那？直到现在，业界也没有一个准确的答案。

在Kaggle上的很多比赛中，大部分的获胜方案都是使用的人工特征工程，构造低阶的组合特征，这些特征意义明确且高效。而DNN学习到的特征都是高度非线性的高阶组合特征，含义非常难以解释。那么是否能设计一种DNN的特定网络结构来改善DNN，使得其学习 起来更加高效那？

业内进行了很多探索，DCN就是其中一个。

# Deep & Cross Network

## DCN模型结构



可以看到DCN分成4部分。最底层是“Embedding and stacking layer”（嵌入层和堆积层），中间部分是并行的“Cross network”和“Deep network”，最上层是“Combination output layer”（组合层），它结合了cross network和deep network两个网络的输出。下面分别对每个部分进行讲解。

由上图可以看到DCN与Wide & Deep有相似的地方，比如两种模型中两个网络都是同时训练，再比如两个网络的输出，concat之后再输入到sigmoid函数中。

## Embedding and Stacking Layer

这一层是模型的输入层。输入的特征分为稠密特征和稀疏特征（类别特征）。

对于稀疏特征，通常会被处理成one-hot向量，这会导致特征的维度很高，且数据非常稀疏。为了减小维度，通常用embedding来把它们转化为稠密向量。

为embedding向量，为第i个类别型特征one-hot后的输入向量

,, W为embedding矩阵，会在网络中和其他参数一起优化。和分别为embedding尺寸和vocabulary尺寸。

Embedding操作其实就是用一个矩阵和one-hot之后的输入相乘，也可以看成是一次查 询（lookup）。这个Embedding矩阵跟网络中的其他参数是一样的，是需要随着网络一起学习的。

最后把稠密特征和经过转换的稀疏特征对应的稠密向量concat起来组成模型的最终输 入，也就是x0：



再将X0输入神经网络。

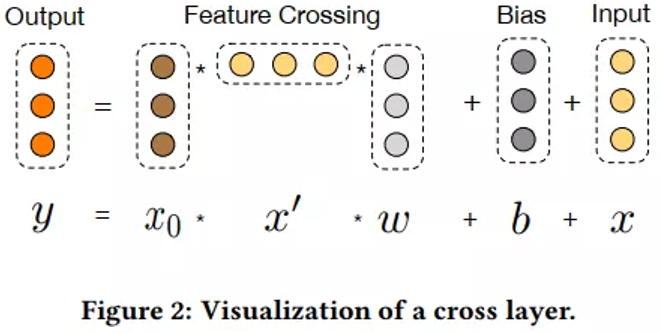
## Cross Network

cross network的核心是高效的学习组合特征。cross network由cross layer组成，每一层的公式如下：



其中，xl和xl+1分别为第l层和第l+1层的输出，wl和bl分别为第l层的weight和 bias,是需要学习的参数。每层的输入和原始输入x0进行交互，最后还要加上本层的输入。特殊的cross network结构导致特征的交互随着层数增加而更深入。注意上式中所有的变量均是列向量，W也是列向量，并不是矩阵。

上面的式子可以这样理解。对于，可以这样理解。每一层的输出，都是上一层的输出加上feature crossing f。而f就是在拟合该层输出和上一层输出的残差。one cross layer可视化如下图。



**复杂度分析：**假设Lc表示交互层层数，d表示输入维度。则整个网络的参数数量为：所以时间复杂度和空间复杂度依然是线性的。因为每一层的W和b都是d维度的。从这个式子可以发现，复杂度是输入维度d的线性函数。所以相比于deep network，cross network引入的复杂度微不足道。这样就保证了DCN 的复杂度和DNN是一个级别的。论文中表示，Cross Network之所以能够高效的学习组合特征，就是因为 的秩为1，使得我们不用计算并存储整个的矩阵就可以得到所有的cross terms。

为什么这么设计？

Cross Network特殊的网络结构使得cross feature的阶数随着layer depth的增加而 增加。当cross network为1层的时候，得到最高是2维的特征交叉；cross network 为2层的时候，得到最高是3维的特征交叉；cross network为3层的时候，得到最高是4维的特征交叉；以此类推。。。

因此cross network以一种参数共享的方式，通过对叠加层数的控制，高效地学习低维特征交叉组合，避免了人工特征工程。

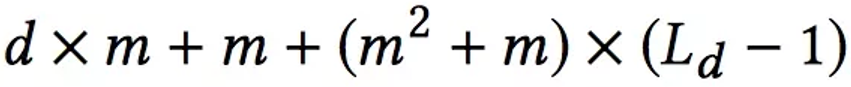
由于cross network的参数数量很少，这限制了模型的性能，为了能够学习高度 非线性的组合特征，因此引入了一个 deep network（DNN）来捕获非线性特征交互。

## Deep Network

deep network是一个传统的全连接前向传播神经网络，用来学习高维非线性特征交叉组合。公式如下：

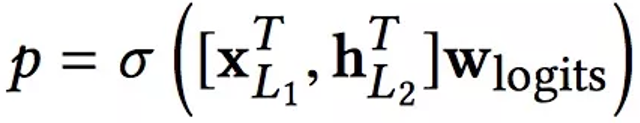


假设输入x0维度为d，一共有Ld层神经网络，每一层的神经元个数 都是m个。那么总的参数或者复杂度为：



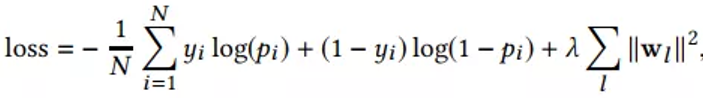
## Combination Layer

Combination Layer把Cross Network和Deep Network的输出拼接起来，然后经过加权求和后得到logits，然后经过sigmoid函数得到最终的预测概率。形式化如下：



其中，p是最终的预测概率；是d维的，表示Cross Network的最终输出；是m维的， 表示Deep Network的最终输出；是Combination Layer的权重；最后经过sigmoid函数，得到最终预测概率。sigmoid函数的公式如下：

损失函数是一个带有正则项的交叉熵损失函数：



其中，pi是Combination Layer的输出，yi是真实的label，N为输入的样本数量, 公式最右边一项为l2正则化。

注：在DCN中，Cross Network和Deep Network是一起训练的，这样网络可以知道另外一个网络的存在。

# DCN的特点

1.使用cross network，在每一层都应用feature crossing。高效的学习了bounded degree组合特征。不需要人工特征工程。

2.网络结构简单且高效。复杂度由layer depth决定。

3.相比于DNN，DCN的logloss更低，而且参数的数量将近少了一个数量级。

# 泛化FM

论文中提到DCN本质是FM的拓展，可以理解为利用既有的特征，通过交叉产生独立于现有特征的高阶特征。要使用DCN获得高阶特征，只需要增加cross network结构的层数即可，而且需要的参数也是输入层维数d的线性倍数。这也是DCN的优势所在。

DCN和FM模型一样都使用了参数共享机制，只是DCN把这种参数共享的思想从一层扩展到多层，并且可以学习高阶的特征组合。参数共享不仅仅使得模型更加高效而且使得模型可以泛化到之前没有出现过的特征组合，并且对噪声的抵抗性更加强。

FM是一个非常浅的结构，并且限制在表达二阶组合特征上，Deep & Cross Network(DCN) 把这种参数共享的思想从一层扩展到多层，并且可以学习高阶的特征组合。但是和FM的高 阶版本的变体不同，DCN的参数随着输入维度的增长是线性增长的。

# 实践

参考：https://github.com/jpegbert/code\_study

# 总结

DCN是一个可以同时高效学习低维特征交叉和高维非线性特征的深度模型，不需要人工 特征工程或遍历搜索，所需的计算资源非常低。对比Wide&Deep，DCN不需要特征工程来获得高阶的交叉特征（Wide&Deep模型中，Wide部分需要人工特征工程）。对比FM系列的模型，DCN拥有更高的计算效率并且能够提取到更高阶的交叉特征。

参考

<https://mp.weixin.qq.com/s/S0gFnjU8UTHTM9bic59dhA>

<https://mp.weixin.qq.com/s/Xp_xTmcx56tJqfjMhFsArA>

<https://mp.weixin.qq.com/s/lF_WLAn6JyQqf10076hsjA>

<https://mp.weixin.qq.com/s/DEXqbevceav03ee88k1W5Q>

<https://mp.weixin.qq.com/s/ulXRVoj1m7MMK3X-JRfONw>

<https://mp.weixin.qq.com/s/XK0doAhDfyzxzJVgMlnc1g>

<https://mp.weixin.qq.com/s/kz3dxF-UdzdM5kAvC7KsPw>