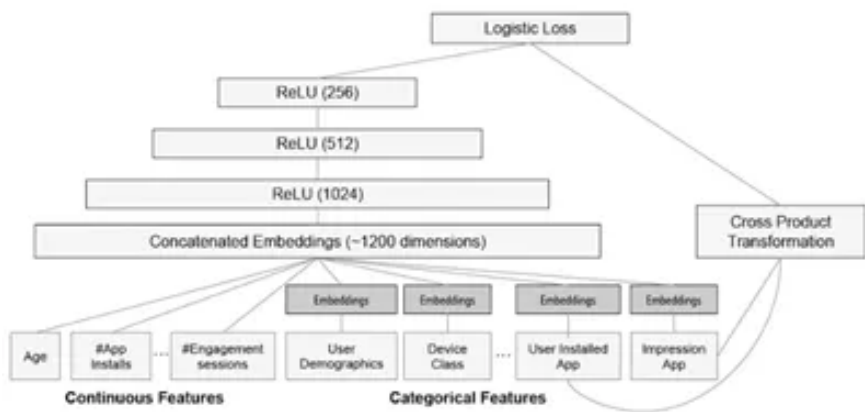


深度特征挖掘（一）： wide&deep

原创 梅子行 大数据风控与机器学习 2019-08-04

Wide&deep算法最早由谷歌于2016年提出。核心思想是希望将浅层网络的高鲁棒性与深层网络的学习能力融合，从而得到一种更加强大的学习器。它是一种思路非常新颖的算法，后续大量在风控领域大放异彩的算法都是基于该算法的框架衍生得到的。

当然基础的wide&deep网络在业界应用并不多， 主要作为基础方法进行介绍，以便读者更好的理解后面的内容。顾名思义， wide&deep算法可以拆分成两部分： wide网络和deep网络。接下来分开介绍一下两部分网络的具体形式。



1. wide网络

Wide网络部分是一种较为特殊的网络。它的输入与输出直连，故属于广义线性模型的范畴。可以用：

来表示。但是在的特征部分包括基础特征和交叉特征。交叉特征在wide网络中非常重要，可以帮助线性模型捕捉特征直接的交互信息，为模型增加一定的非线性。而交叉特征可以表示为：

也就是onehot编码后的特征的笛卡尔积。

举个例子，当前有两个特征，分别是：

{性别：男，女}，
{学历：专科，本科，硕士，博士}，

那新的组合特征之一就有**性别_男_学历_博士**。这一特征等于1，当且仅当一个人的性别为男，并且学历为博士。除此之外，这个特征的值都为0。

在从前对于高维稀疏特征的处理，主要是使用线性模型和特征交叉进行组合，从而为模型带来很好的记忆能力，以及高度的可解释性。但是想要提高模型的识别能力需要引入更多的人工特征。

2. deep网络

Deep网络部分也非常简单。它的本质就是最简单的前馈神经网络（Deep Neural Network, DNN）。主要的作用是用于对基础特征的自动挖掘，从而学习到更深层次的交叉特征。DNN是LSTM的鼻祖，和LSTM一样是一种表示学习方法。DNN几乎不需要特征工程，通过对低纬度的稠密特征进行组合可以学习到更深层次的隐藏特征。

特征在DNN中，首先转换为低维稠密向量，维度通常不会超过这一范围。之后经过随机初始化，通过逐层的线性拟合加Relu激活函数引入非线性。可以表示为：

在网络中加入deep网络可以大大的增强模型的学习能力，弥补广义线性模型的缺点。

但是DNN有过度学习的倾向，单独的DNN模型通常方差较大。在wide&deep中，deep网络与wide网络相互结合，模型可以各取所长，从而得到泛化能力和预测能力都很优秀的模型。

3. 联合训练

Wide&deep模型中，两部分模型需要联合在一起进行训练，而不是进行模型集成。在前面的文章中有介绍过模型集成的相关概念。对于模型集成，在训练时，两个模型是分别训练的。可以串行，也可以并行。只有在最终做预测的时候，两个模型才会进行加权组合。而模型的联合训练，需要在模型学习的过程之前，首先将两个模型的损失函数进行组合，得到一个整体的损失，然后进行学习。

首先在前向传播中，wide和deep两部分的输出，通过加权进行组合的，然后通过sigmoid函数进行映射得到最终输出。

联合模型求解，原论文中使用的是FTRL算法，辅以L1正则。DNN网络的部分则使用AdaGrad优化算法。感兴趣的读者可以自行深入了解。

4. 应用

首先wide&deep具备着广义线性模型和DNN两者的优点。所以会比单一使用其中一种效果更好。在风控领域有过一定尝试，效果在逻辑回归之上，但是在跨时间验证集上的表现一般，整体考察下来没有超出DNN的预测能力范畴。

较为突出的一个使用场景是对一些长尾数据进行建模的时候，DNN由于可以将高维稀疏特征进行嵌入，对于手机装载app名单（applist）、浏览器搜索记录等经过onehot编码后会变成高维稀疏特征的变量，提取效果会比其他常用的风控模型效果更好。同样对这种变量支持度较好的一种建模方式是通过XGBoost进行特征的离散交叉，后带入逻辑回归进行建模，这种方法在下一节中介绍。

此外，在建模的实际操作中，对于wide&deep通常需要对连续特征进行离散化。在wide的部分，由于是线性模型，显然是需要进行离散化的。而在DNN中，对联系变量做离散化，可以使得一个特征的不同区间获得不同的表示（Embedding），从而在网络中获得更加细致的刻画，从而使得模型的效果得到提高。

[阅读原文](#)