[DIN]阿里点击率预估任务中的深度兴趣网络

原创 飞翔的死胖子 高山仰止z 3月15日

1.简介：

DIN是阿里盖坤大神团队提出的模型，论文发表于18年9月，全名为：《Deep Interest Network for Click-Through Rate Prediction》。DIN模型的特点是引入了attention机制，充分挖掘用户历史行为序列中的信息。DIN模型已经被成功的应用在阿里的线上广告系统中了。下面一起来看看吧。

近年来，一些基于深度学习的模型被应用在点击率预估任务当中。这些模型采用了类似的embedding & MLP的形式。通常的做法是将高维的稀疏特征映射到低维空间，然后再转化为一个定长的向量，最后将这些定长的向量输入到多层感知机中，去学习特征之间的非线性关系。这种做法将用户特征转化为了一个定长的向量，这个定长的向量成为了瓶颈，会制约模型捕捉多变的用户兴趣。本文提出的DIN模型就是为了解决这个问题。DIN用一个局部激活单元从用户历史行为中适应性地学习用户兴趣的表示方法。

低维的定长向量难以表示大量的不同的的用户兴趣，但是如果用定长向量表示所有的的用户兴趣，那么将这个向量的维度会变得特别大，这会导致要学习的参数特别多，加大过拟合风险并且会带来储存方面的困难。另一方面在预测一个候选集时，对于一个用户，并不需要将他所有的兴趣放在同一个向量之中。因为一个在购买游泳用品的用户，给他推荐护目镜就行了，并不需要给他推荐他上周想买的鞋子。DIN模型更关注和候选集相关的用户历史行为。

除此之外，本文还提出了novel mini-batch aware regularization和新的激活函数PReLU ，新的正则化方法能够减小计算量，PReLU 考虑了输入数据的分布情况。

2.特征表示：

文中用到的特征分为4部分：用户档案特征、用户行为特征、ad特征、上下文特征。所有的特征没有进行交叉，我们将用DNN捕捉交互特征。

3.BASE MODEL：

在介绍DIN之前我们来看看base model。

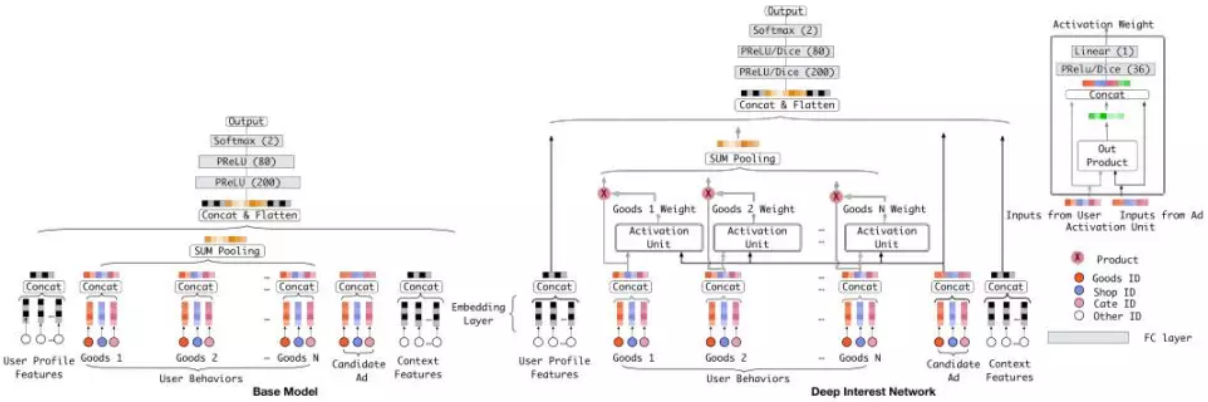

Figure 2: Network Architecture. The left part illustrates the network of base model (Embedding&MLP). Embeddings of cate_id, shop_id and goods_id belong to one goods are concatenated to represent one visited goods in user's behaviors. Right part is our proposed DIN model. It introduces a local activation unit, with which the representation of user interests v… a… given different candidate ads.

figure 2中左边为Base Model，不同的用户有不同数量的历史行为，这就导致用户历史行为特征的数量是可变的，而全连接层只能接受定长的输入。比较常见的做法是采用池化层将变长的输入转化为固定长度。

$$\boldsymbol{e}_i = \text{pooling}(\boldsymbol{e}_{i_1}, \boldsymbol{e}_{i_2}, ...\boldsymbol{e}_{i_k}). \quad (1)$$

嵌入层和池化层共同将原始稀疏特征转化为固定长度的表示向量。然后所有的向量可以被拼接到一起作为实例的表示向量。

4.DIN结构：

figure2中右边的部分就是DIN的结构。与base model相比，DIN引入了局部激活单元，其他部分和base model完全一样。激活单元被用在用户历史行为特征中，作为池化层的权重来计算用户表示向量。如公式（3）所示：

$$\boldsymbol{v}_U(A) = f(\boldsymbol{v}_A, \boldsymbol{e}_1, \boldsymbol{e}_2, .., \boldsymbol{e}_H) = \sum_{j=1}^{H} a(\boldsymbol{e}_j, \boldsymbol{v}_A)\boldsymbol{e}_j = \sum_{j=1}^{H} \boldsymbol{w}_j \boldsymbol{e}_j, \quad (3)$$

其中，{e1, e2, ..., eH } 为用户行为的embedding向量。vA是ad A的embedding向量。

局部激活单元的想法和attention机制类似。但是和传统的attention机制不同，公式3中的的权重(wj)之和不再必须等于1.这是为了保证用户兴趣的强度。相比传统的attention机制，对于输出a(·)的使用softmax规范化被抛弃了。我们也尝试过用LSTM对用户历史行为进行序列化建模，但是效果并不好。和NLP任务不同，用户历史行为可能包含很多同时存在的兴趣。如果突然中断这些兴趣，可能会产生噪声。

5.训练技术：

5.1.Mini-batch Aware Regularization 自适应正则化

在工业界中深度学习网络中很经常发生过拟合现象，然而对于像阿里这样的存在上亿

参数需要计算的深度网络，如果使用L1，L2的正则是显然是不科学的，举个例子：如果L2的方法对网络参数进行调整，L2的计算需要依赖整个模型参数，然而正则起作用的只有一些非零的参数。所以我们引入公式如下：

$$L_2(\mathbf{W}) = \|\mathbf{W}\|_2^2 = \sum_{j=1}^{K} \|\boldsymbol{w}_j\|_2^2 = \sum_{(\boldsymbol{x},y)\in\mathcal{S}} \sum_{j=1}^{K} \frac{I(\boldsymbol{x}_j \neq 0)}{n_j} \|\boldsymbol{w}_j\|_2^2, \quad (4)$$

其中wj 表示第j个embedding向量，I(xj，0)表示第x个实例是否含有特征id j，nj 表示所有样本中特征j出现的次数。在mini bantch中公式4可以被转化为公式5：

$$L_2(\mathbf{W}) = \sum_{j=1}^{K} \sum_{m=1}^{B} \sum_{(\boldsymbol{x},y)\in\mathcal{B}_m} \frac{I(\boldsymbol{x}_j \neq 0)}{n_j} \|\boldsymbol{w}_j\|_2^2, \quad (5)$$

其中，Bm 表示第m个mini bantch。我们可以看出两点：1）正则约束对出现的特征频次有关，出现频次越多正则化化约束越强，反正相同。2）该方法通过自适应的学习方式减少了部分的计算开销。

$$a_{mj} = max(x, y) \in B_m$$

a mj表示mini bantch中是否有最少一个样本含有特征j，将a mj代入公式5，得到公式6：

$$L_2(\mathbf{W}) \approx \sum_{j=1}^{K} \sum_{m=1}^{B} \frac{\alpha_{mj}}{n_j} \|\boldsymbol{w}_j\|_2^2. \quad (6)$$

因此，第m个bantch 中embedding的权重w的更新方式应该为：

$$\boldsymbol{w}_j \leftarrow \boldsymbol{w}_j - \eta \left[\frac{1}{|\mathcal{B}_m|} \sum_{(\boldsymbol{x},y)\in\mathcal{B}_m} \frac{\partial L(p(\boldsymbol{x}), y)}{\partial \boldsymbol{w}_j} + \lambda \frac{\alpha_{mj}}{n_j} \boldsymbol{w}_j \right], \quad (7)$$

5.2.数据自适应激活函数

PRULU是RELU激活函数的改良版。relu的公式为：y = max(0, wx+b)由于当输入小于0时，relu的取值为0，这可能导致网络停止更新。PRelu对公式进行了修改，使得输入小于0时输出不等于0.

$$f(s) = \begin{cases} s & \text{if } s > 0 \\ \alpha s & \text{if } s \leq 0. \end{cases} = p(s) \cdot s + (1 - p(s)) \cdot \alpha s, \quad (8)$$

研究表明，PReLU能提高准确率但是也稍微增加了过拟合的风险。无论是ReLU还是PReLU突变点都在0，论文里认为，对于所有输入不应该都选择0点为突变点而是应该依赖于数据的。于是提出了一种**data dependent**的方法：**Dice激活函数**。形式如下：

$$f(s) = p(s) \cdot s + (1 - p(s)) \cdot \alpha s, \quad p(s) = \frac{1}{1 + e^{-\frac{s - E[s]}{\sqrt{Var[s] + \epsilon}}}} \quad (9)$$

E[s]和Var[s]分别为mini bantch输入值的均值和方差。ϵ 是一个很小的常量，在文中被设定为10的-8次方。

pi的计算分为两步：

1. 首先，对x进行均值归一化处理，这使得整流点是在数据的均值处，实现了data dependent的想法；
2. 其次，经过一个sigmoid函数的计算，得到了一个0到1的概率值。巧合的是最近google提出的Swish函数形式为x*sigmoid(x) 在多个实验上证明了比ReLU函数x*Max(x,0)表现更优。

另外，期望和方差使用每次训练的mini batch data直接计算，并类似于Momentum使用了指数加权平均。alpha是一个超参数，推荐值为0.99。

6.结果可视化：

文中以一个年轻妈妈为例，挑选了9个种类，每个种类100个商品作为ad的候选集。embedding向量的聚类结果如图所示：

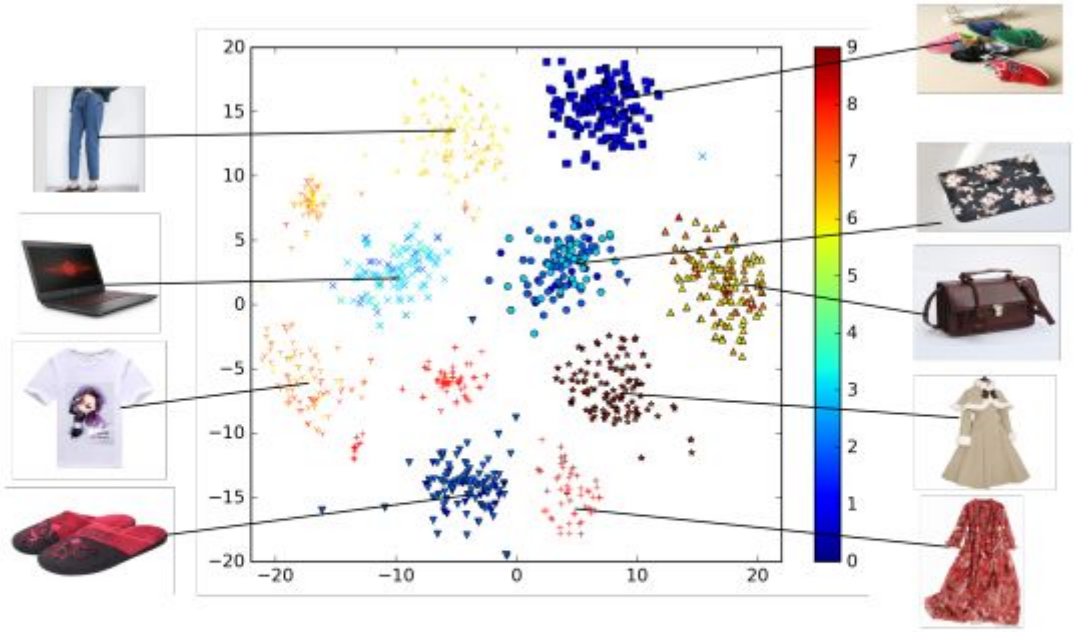

Figure 6: Visualization of embeddings of goods in DIN. Shape of points represents category of goods. Color of points corresponds to CTR prediction value.

我们可以看到相同的种类的物品几乎都被聚到了同一个类中，说明DIN的局部激活单元的效果还不错。

7.总结

1）用户有多个兴趣爱好，访问了多个good_id，shop_id。为了降低纬度并使得商品店铺间的算术运算有意义，我们先对其进行Embedding嵌入。那么我们如何对用户多种多样的兴趣建模那？使用**Pooling对Embedding Vector求和或者求平均**。同时这

也解决了不同用户输入长度不同的问题，得到了一个固定长度的向量。这个向量就是用户表示，是用户兴趣的代表。

2）但是，直接求sum或average损失了很多信息。所以稍加改进，针对不同的behavior id赋予不同的权重，这个权重是由当前behavior id和候选广告共同决定的。这就是Attention机制，实现了Local Activation。

3）DIN使用activation unit来捕获local activation的特征，使用weighted sum pooling来捕获diversity结构。

4）在模型学习优化上，DIN提出了Dice激活函数、自适应正则 ，显著的提升了模型性能与收敛速度。