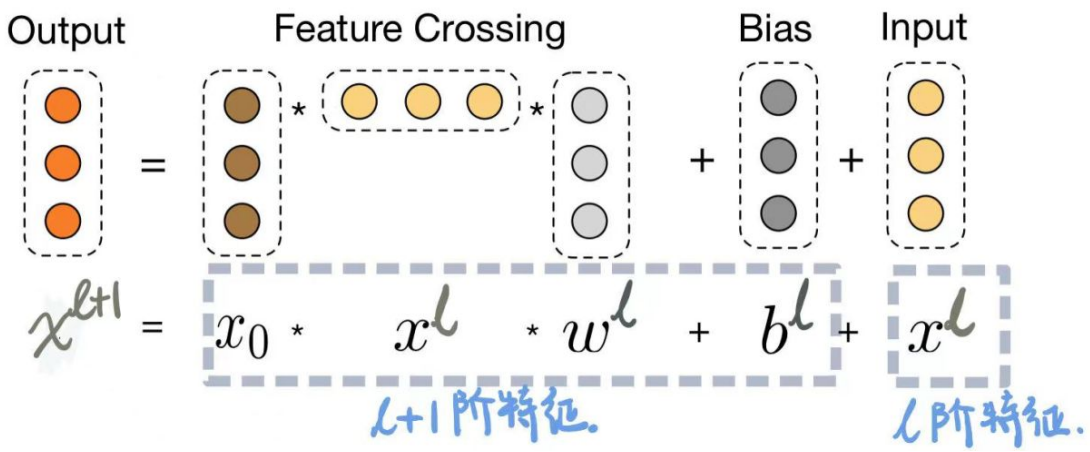
DCN-V2: Improved Cross&Deep Network [2020]

1. 相比DCN-V1的改进

DCN-V1结构图：

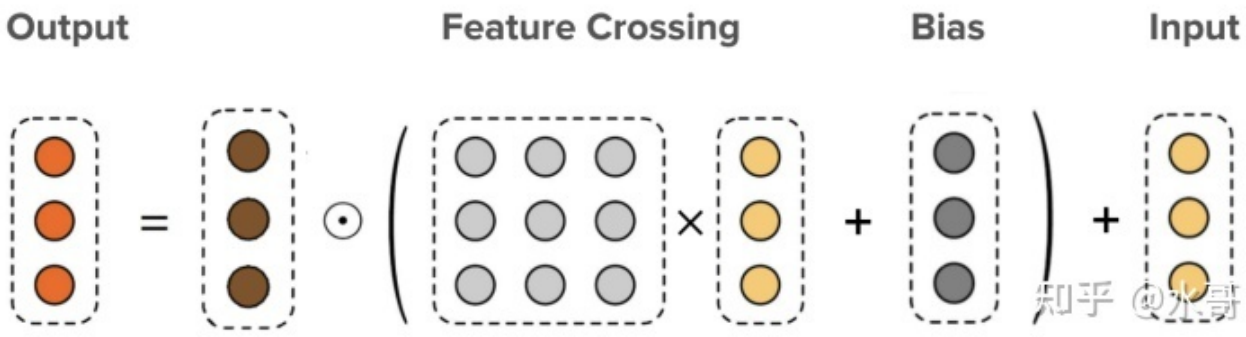


它的核心表达式为：

$$x_{(l+1)} = x_0 x_{(l)}^T w + b + x_{(l)}$$

上次我们讲过，DCN中的Cross端参数量很少，每一层只有一个向量足矣。但是这个带来的问题就是，Cross端参数和Deep端极不平衡，大量参数都集中在deep端。当训练集为亿级的时候，模型的capacity不够。

而V2的结构为：



它的表达式可以写为：

$$x_{l+1} = x_0 \odot (W_l x_l + b_l) + x_l$$

可以看出，最大的变化是将原来的向量 $\boldsymbol{w}$ 变成了**矩阵**（这个计算方式和Fibinet中的双线性交叉很像，都是先乘上一个矩阵，然后就哈达玛积），而这一个改动就解决了前面的问题。一个矩阵 $\boldsymbol{W}$ 拥有足够多的参数来保留高阶交叉信息，或者挑选需要的交叉结果。因此这个工作也实现了真正的高阶交叉。

要注意的一个DCN-V2和xDeepFM的很大区别是，DCN-V2仍然不是vector-wise的操作。根源在于，DCN-V2把所有特征的embedding **concat起来**一起输入网络，所以在 $\boldsymbol{W_l}$ 那里无法保持同一个特征的embedding同进退，同一段embedding自己内部也存在交叉。

3.2 Mixture of Low-Rank DCN

$\boldsymbol{W_l} \in \mathbb{R}^{\boldsymbol{d}\times\boldsymbol{d}}$, 当field很多，embedding size很大的时候，参数量就非常大。因此作者引入了【**低秩分解**】来处理，即把 $\boldsymbol{W_l}$ 变成两个"瘦"矩阵的乘，即 $\boldsymbol{U,V} \in \boldsymbol{R^{d\times r}}$, $\boldsymbol{UV^T} = \boldsymbol{W}$。当 $\boldsymbol{r} \leq \boldsymbol{d/2}$的时候，就能够达到【压缩】的目的。这样做的intuition是，W_l通常是低秩的，即特征值的decay非常明显，只有较少的非零特征值。

此时有：

$$\boldsymbol{x_{(l+1)}} = \boldsymbol{x_0} \odot (\boldsymbol{U_l}(\boldsymbol{V_l^T}\boldsymbol{x_{(l)}}$$

+b_l)+x_l，

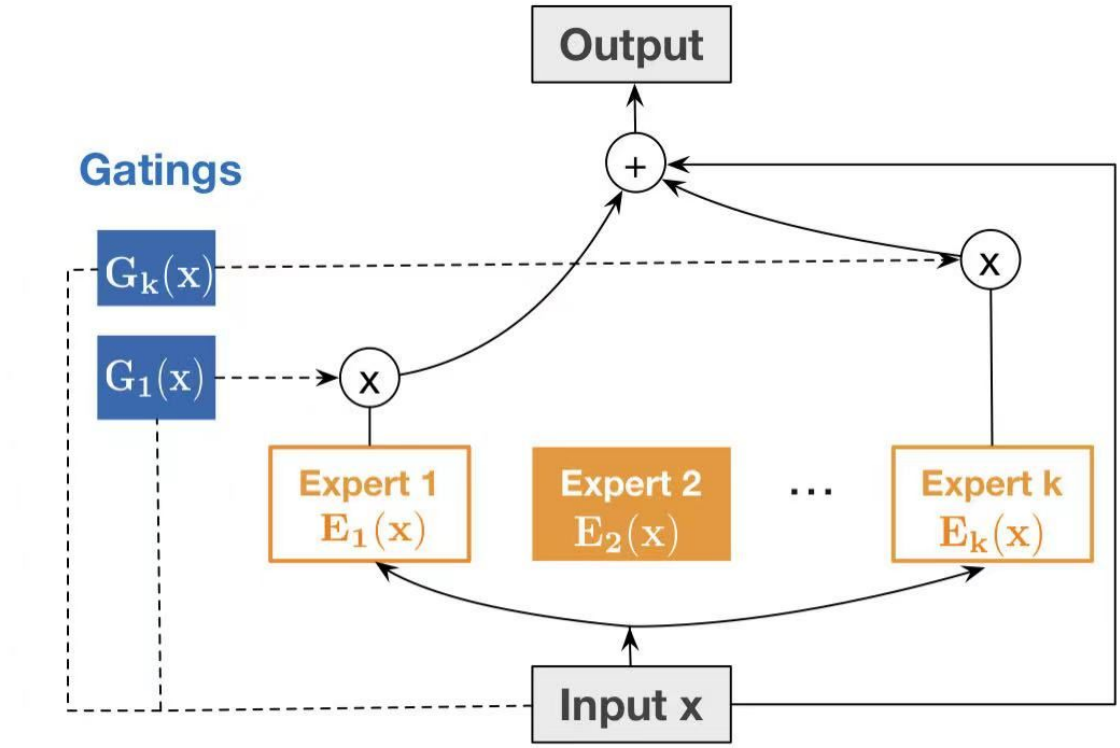
其中$U_l V_l^T$即是近似的矩阵W_l.

DCN-mix中结合了MoE(multi-of-Expert)的思想，认为矩阵的低秩分解其实是在不同特征空间上的映射，所以可以采用多个特征空间。因此，做不同的低秩分解就是不同的Expert。然后，用门控网络（即注意力网络）进行加权求和：

$$\mathrm{x}_{l+1} = \sum_{i=1}^{K} G_i(\mathrm{x}_l) E_i(\mathrm{x}_l) + \mathrm{x}_l$$

$$E_i(\mathrm{x}_l) = \mathrm{x}_0 \odot \left(U_l^i \left(V_l^{i\top} \mathrm{x}_l \right) + \mathrm{b}_l \right)$$

其中，$E_i(x_l)$即为在不同低秩分解下的$l+1$阶交叉特征，门控网络是通过输入x_l得到的，表示对不同低秩分解得到的表示的一种加权注意力。



(b) Mixture of Low-rank Experts

效果：DCNv2效果还是比较明显的，要优于autoint。