FNN

基于**FM预训练**获取特征embedding表示，然后拼接起来，输入MLP来进行CTR的预估。使用DNN来对FM的embedding进行再交叉，从而产生高阶的特征组合（隐式），加强模型对数据模式的学习能力。

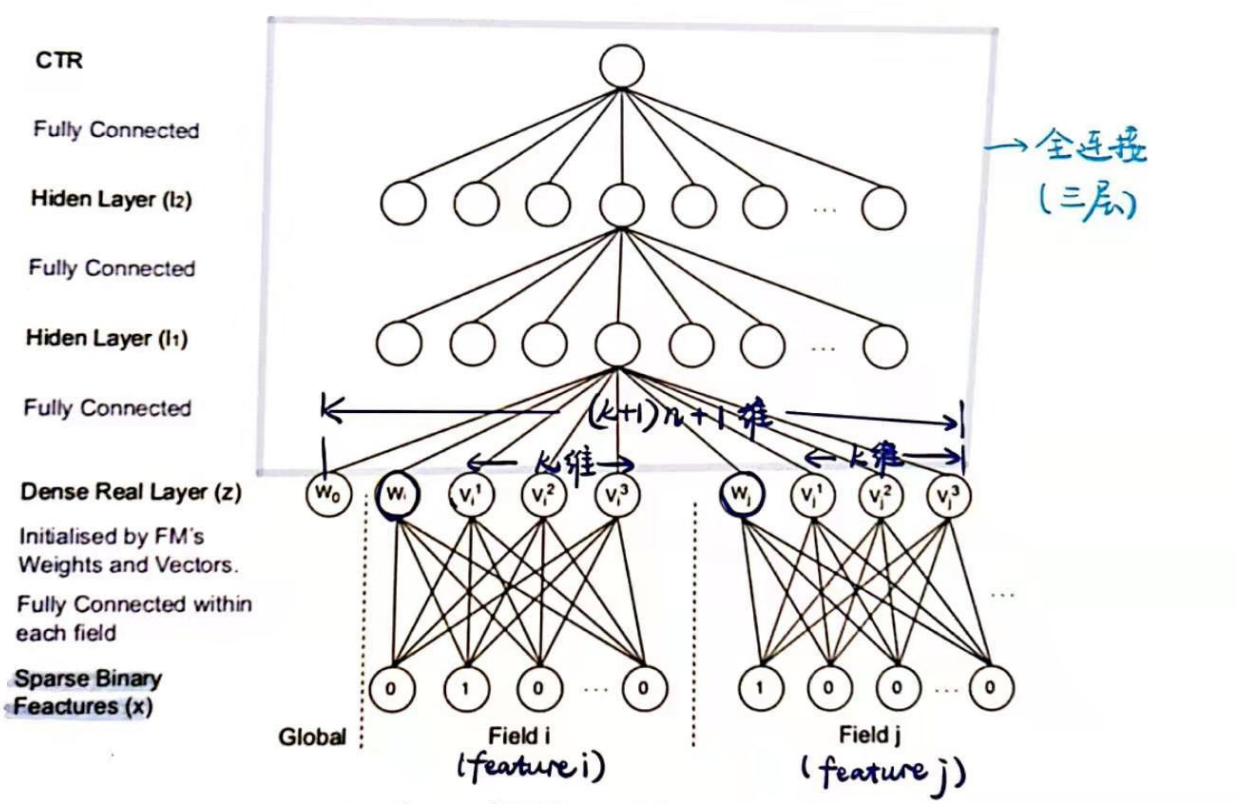


Fig. 1. A 4-layer FNN model structure.

使用FM预训练的embedding，在FNN中之后并没有通过fine tuning来调整参数，所以FNN不是end-to-end训练的，而是一种“贪心”的训练方法。FNN有如下几个问题：

1）FM中进行特征组合，使用的是隐向量**点积**。将FM得到的隐向量移植到DNN中接入全连接层，全连接本质是将输入向量的所有元素进行加权**求和**，且不会对特征Field进行区分（是bit-wise的，因为全部concat起来了），也就是说FNN中高阶特征组合使用的是全部隐向量元素相加的方式。说到底，在理解特征组合的层面上FNN与FM是不同的(FM: 点积，FNN：加权平均)，而这一点也正是PNN对其进行改进的动力。

2）在神经网络的调参过程中，参数学习率是很重要的。况且FNN中底层参数是通过FM预训练而来，如果在进行反向传播更新参数的时候学习率过大，很容易将FM得到的信息抹去。FNN至少应该采用Layer-wise learning rate(**不同层的学习率不同**)，底层的学习率小一点，上层可以稍微大一点，在保留FM的二阶交叉信息的同时，在DNN上层进行更高阶的组合。

AFM (Attentional Factorization Machines) [2017]

AFM解决的问题是，FM中虽然计算了二阶交叉特征，但是它并没有区分不同交叉特征的权重。回忆一下FM的公式，每个交叉特征$x_i x_j$的权重就是$< v_i, v_j >$直接算出来的。但是实际上，有些特征很重要、有些不那么重要，所以应该用注意力机制来分配给他们不同的权重才是。同时，注意力机制也提供了可解释性 -- 我们可以知道那些交叉特征更为重要一些。

> ...FM lacks such capability of differentiating the importance of feature interactions, which may result in suboptimal prediction. ...we enable feature interactions contribute differently to the prediction.

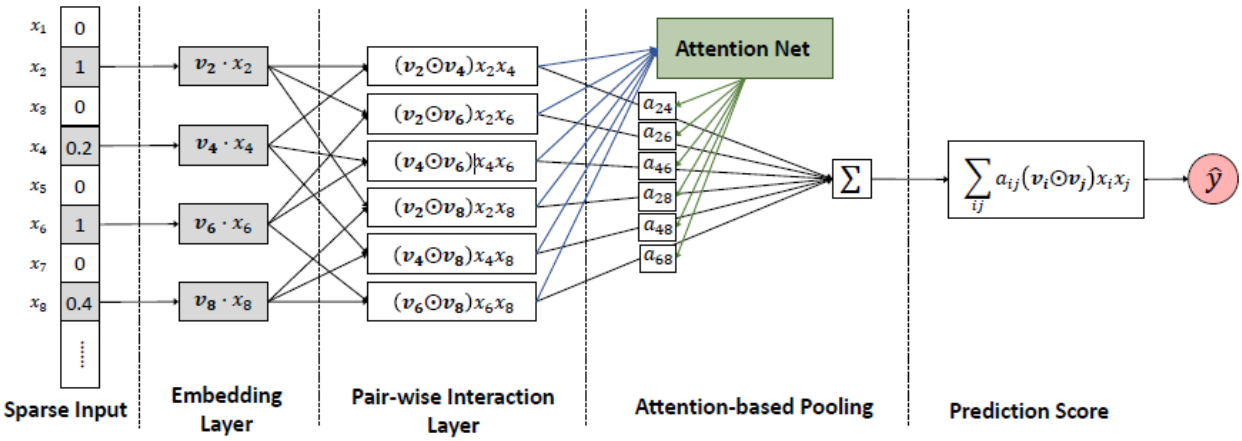


Figure 1: The neural network architecture of our proposed Attentional Factorization Machine model.

- Spare Input -> Embedding Layer: 和FM中一样，对特征做embedding。
- Pair-wise Interaction layer: FM 是直接求两个特征embedding的点积，把好不容易建立好的embedding压缩成了一个数，这样做是会损失模型的表达能力的。所以，在这里我们使用的是element-wise的哈达玛积，能够继续保持embedding形式，适用于神经网络。(注意这里只有x_i, x_j都不为0，才有值。所以最后应该有C^2_{field}个交互特征，field是域的个数。如果每个域都是one-hot而不是multi-hot的话)。如果不考虑attention的话，这里就是所有二阶特征embedding求哈达玛积的sum-pooling，然后经过全连接得到最终得分$\hat{y}$.
- Attention net: 为了解决FM不能区分交叉特征重要性的问题，论文中在二阶交互特征计算完成后加入了注意力网络（Attention Net），使用多层感知器（MLP）得到注意力权重：

$$a'_{ij} = \mathbf{h}^T ReLU(\mathbf{W}(\mathbf{v}_i \odot \mathbf{v}_j)x_i x_j + \mathbf{b}) \tag{5}$$

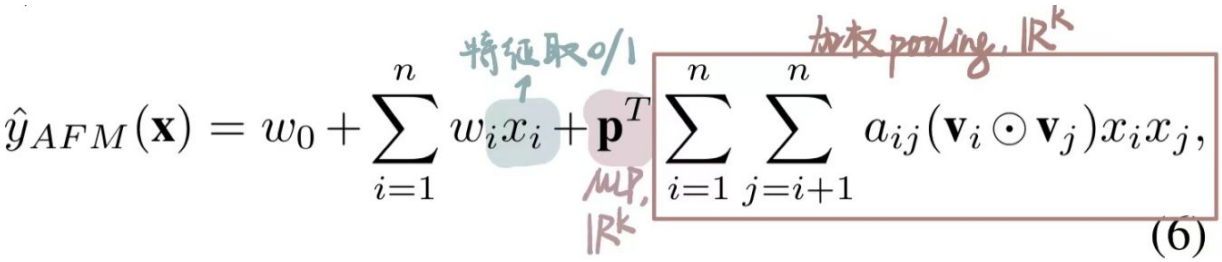
$$a_{ij} = \frac{\exp(a'_{ij})}{\sum_{(i,j)\in\mathcal{R}_x} \exp(a'_{ij})}, \text{(softmax)}$$

a_{ij}表示第i个特征和第j个特征的组合特征对于结果的重要程度。

- **Attention-based Pooling**: 就是将Pair-wise Interaction Layer中所有的特征用attention net中算出来的特征重要性a_{ij}进行加权求和。得到$\sum_{ij} a_{ij}(v_i \odot v_j)x_i x_j$. 这是一个k维向量。这里通过使用注意力机制，增强了模型的可解释性。但是AFM的局限是，只能提供二阶特征的可解释性，并不能够对高阶特征提供可解释性（这个在Autoint和InterHAt中得到了部分解决）。

- **prediction score:** 使用一阶特征和二阶特征进行得分的预测：

$$\hat{y}_{AFM}(\mathbf{x}) = w_0 + \sum_{i=1}^{n} w_i x_i + \mathbf{p}^T \sum_{i=1}^{n} \sum_{j=i+1}^{n} a_{ij}(\mathbf{v}_i \odot \mathbf{v}_j) x_i x_j, \tag{6}$$



避免过拟合的trick：使用特征dropout来避免复杂的特征共现(prevent complex co-adaption). 由于所有出现特征都来计算pair-wise interaction，这样的特征是非常之多的，然而有些特征未必有用。所以，可以drop掉一些特征。

> Since AFM models all pair-wise interactions between features while not all interactions are useful, the neurons of the pair-wise interaction layer may easily co-adapt with each other and result in overfitting. As such we employ dropout on the pair-wise interaction layer to avoid co-adaptions.