NFM理论与实践

# 背景

在CTR预估、广告、搜索和推荐等场景中，通常有很多类别特征，要使模型具有更好的效果，通常需要采用特征组合的方式揭示人工无法直接发现的隐藏于特征交叉背后的特征，而过多的类别直接导致了特征组合也是非常多的。传统的做法是通过人工特征工程或者利用决策树来进行特征选择，选择出比较重要的特征。但是这样的做法都有一个缺点，就是：无法学习训练集中没有出现的特征组合。

近几年，Embedding-based方法开始成为主流，通过把高维稀疏的输入embed到低维度的稠密的隐向量空间中，模型可以学习到训练集中没有出现过的特征组合。

Embedding-based大致可以分为两类：

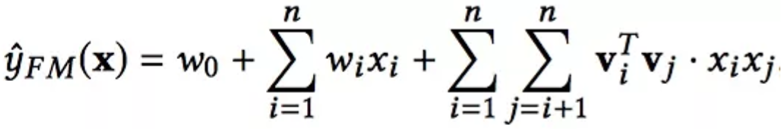
（1）factorization machine-based linear models；

（2）neural network-based non-linear models

下面分别简单介绍下

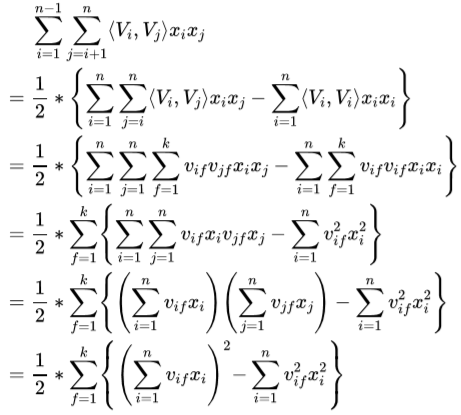
# FM

FM全称为Factorization Machine通过隐向量内积来对每一对特征组合进行建模。形式化为：

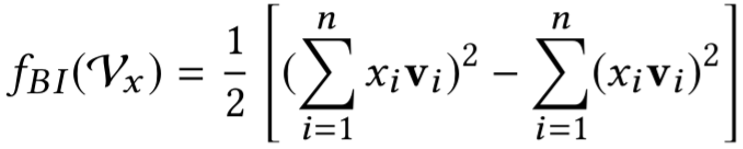


w0是全局偏置；wi是单特征的权重；vi和vj分别代表特征xi和xj的隐向量。类别型特征经过one-hot得到每一位代表一个特征。比如gender经过one-hot之后，gender=male是一个特征，gender=female是另一个特征。

由于NFM二阶部分与FM二阶部分十分类似，所以这里先回顾下FM二阶部分化简过程：



可以看到，交叉项得到的是一个值，不考虑最外层的求和，我们可以得到一个k维的向量。这个k维的向量就是NFM所谓的"Bi-Interaction Layer"的结果（即NFM二阶交叉项）:



注：NFM二阶交叉项部分可以看做是FM二阶交叉项不考虑最外层的求和，得到的一个k维的向量

FM很好的解决了高纬度高稀疏输入特征组合的问题。通过隐向量内积来建模权重，针对在训练集中没有出现过的特征组合或者出现次数很多的特征组合，都能有效的学习。FM虽然考虑了组合特征，但是FM依然属于线性模型，表达能力仍然有限，而且它只能对二阶组合特征进行建模。

## Embedding + DNN

在CTR预估、搜索和推荐等场景中，业界通常会使用DNN（Deep Neural Network）增加模型的非线性表达能力。对于过于稀疏的类别特征采用Embedding的当的方式得到对应的向量，再把得到的向量输入DNN。虽然多层DNN已经被证明可以有效的学习高阶特征组合。但是DNN网络优化（网络学习）比较困难、对于参数较为敏感，训练难度很大。业内大部分DNN的架构都是：把特征的嵌入向量简单拼接起来，输入到神经网络中学习，将捕获高阶特征交叉信息的任务交给了DNN结构。这样简单的拼接嵌入向量，并没有将二阶交叉特征的信息完全表征出来**，**因为缺失了很多组合特征的信息(carry too little information about feature interactions in low level)效果并不好，那么只能寄希望于后面的MLP弥补不足。但是为了提高NN的学习能力就需要增加网络层数，复杂的网络结构会出现梯度消失/爆炸、过拟合、degradation（模型效果随网络结构复杂性的提高不升反降）等问题，网络的学习（优化）非常困难。

有研究表明，如果不对Embedding（嵌入层）预训练，Wide&Deep和DeepCross的性能比FM还差，而且DeepCross严重过拟合，Wide&Deep遇到了degradation问题。如果使用FM预训练初始化Embedding（嵌入层），Wide&Deep和DeepCross性能都提升了，甚至超过 了FM。Wide&Deep的degradation问题也解决了，因为训练集的性能得到了提升。但是过拟合的问题依然存在。这进一步说明DNN的训练学习真的存在困难。

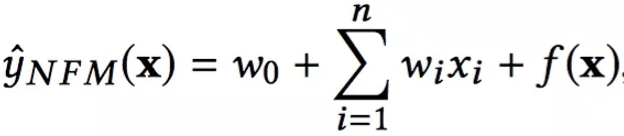
NFM针对FM的缺点，在二阶特征组合的隐向量空间中，引入了非线性变换来提升模型非线性表达能力；同时，也可以学习到高阶的组合特征。

NFM摒弃了直接把嵌入向量拼接输入到神经网络的做法，在嵌入层之后增加了 Bi-Interaction Layer（Bi-linear interaction）对二阶交叉特征进行处理。这使得low level的输入表达的信息更加的丰富，交叉特征的信息能更好的被DNN学习，降低DNN学习更高阶交叉特征信息的难度。减轻DNN的负担，意味着不再需要更深的网络结构，从而模型参数量也减少了，模型训练更便捷。这极大的提高了后面隐藏层学习高阶非线性组合特征的能力。

# NFM模型

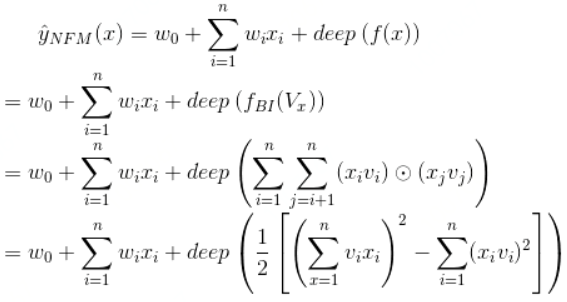
## NFM 结构定义

NFM全称Neural Factorization Machine，它用来解决的问题是Sparse Prediction。NFM公式化定义如下：



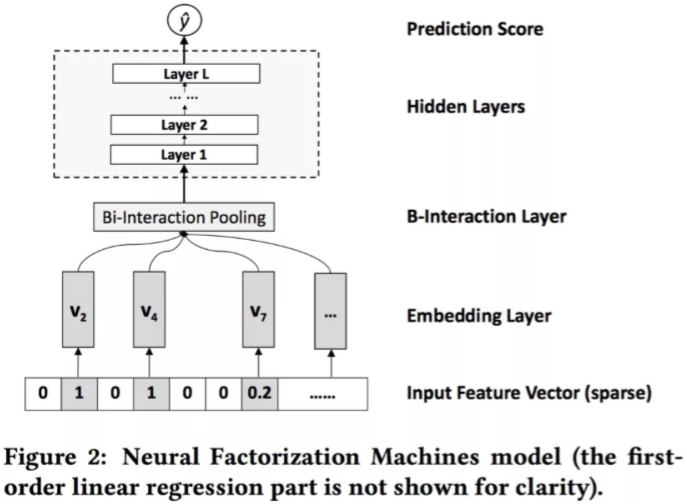
其中，f(x)是NFM的核心，用来学习二阶组合特征和高阶的组合特征模式。

在数学形式上，NFM模型的主要思路是用一个表达能力更强的函数替代原FM中二阶隐向量内积部分。如下：



## 模型结构

模型结构图如下所示，下图并未涵盖一阶项与偏置，完整的NFM是将三者涵盖其中的。



NFM网络架构的特点非常明显，就是在Embedding层和MLP隐层之间加入特征交叉池化层（Bi-Interaction Pooling Layer）。

从结构图可以看出NFM共有五个部分：输入层、Embedding层、特征交叉池化层、隐藏层和输出预测层。最主要的层是特征交叉池化层。将多个embedding向量进行一个池化操作。下面分别对于每个部分进行详细讲解。

## 输入层

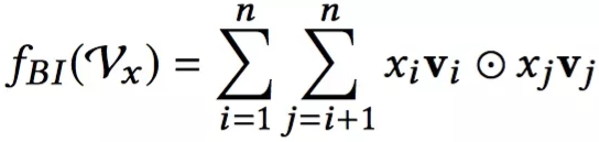
输入层的特征需要离散化处理（实际场景中，分为数值特征与分类特征，分类特征需要先离散化处理）。

## Embedding Layer（嵌入层）

Embedding Layer将高纬的稀疏特征转换为低维度的稠密向量表示。embedding vector的计算可通过lookup table获取。在NFM中，最终输入特征向量是由输入特征值 xi与embedding vector vi相乘得到，vi就是第i个特征的embedding向量，用表示输入向量。

## Bi-Interaction Layer

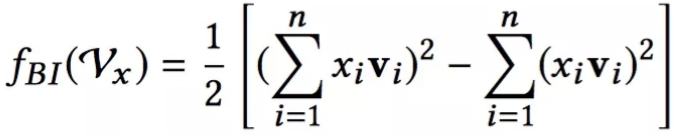
Bi-Interaction Layer是NFM的核心，其本质是一个pooling操作，将embedding vector集合Vx通过池化操作转换为一个向量。其形式化如下：



其中，xi表示不为0的输入特征，表示两个向量对应元素相乘，其结果为一个向量。示例：https://mmbiz.qpic.cn/mmbiz_png/SkYRBSJiaQWQad9JcRZpI0ajiaEXh7TMt0W8S6gtoAicIiazRx3bO2iczINmNcFD8VexGf6HtBpZs9JOAyKibmLgfQgw/640?wx_fmt=png&tp=webp&wxfrom=5&wx_lazy=1&wx_co=1。所以，Bi-Interation Layer将embedding vectors 进行两两交叉运算，然后将所有向量进行对应元素求和，最终为pooling之后的一个向量。

fBI的输入是整个的嵌入向量，xi和xj是特征取值，vi和vj是特征对应的嵌入向量。中间的操作表示对应位置相乘。所以原始的嵌入向量任意两个都进行组合，对应位置相乘结果得到一个新向量；然后把这些新向量相加，就得到了Bi-Interaction的输出。这个输出只有一 个向量。

的计算时间复杂度为，k为embedding vector维度，n为特征的维度，参考FM的优化方法，可以对进一步改写为：



它的计算复杂度是O(kN)。其中k是嵌入向量的维度，N是输入x中非零特征的个数。

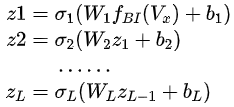
总结，Bi-Interaction Layer实现了对二阶组合特征的建模，与FM中的二阶交叉项相比，没有引入额外的开销，包括参数数量和计算复杂度，能以线性时间复杂度进行训练。

## Hidden Layer

由多个全连接层构成，属于DNN。不过在此之前，作者还采用了Dropout和Batch Normalization，原因如下：

1. Dropout：是神经网络的正则化技术，为了防止过拟合；
2. Batch Normalization：该层是对输入的每个小批量（min-batch）标准化为零均值的单位方差的高斯分布（zero-mean unit-variance Gaussian distribution）。作者使用BN，是为了避免embedding向量的更新过程中修改输入层的分布；

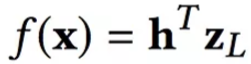
DNN部分的定义如下：



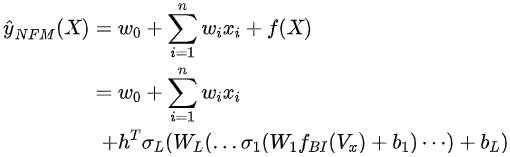
其中，分别表示参数矩阵与偏置向量，表示激活函数，可以取sigmoid，tanh，relu等。关于隐藏层的结构，可以使用类似于FNN中的几种结构：tower，constant，diamond等。

## Prediction Layer

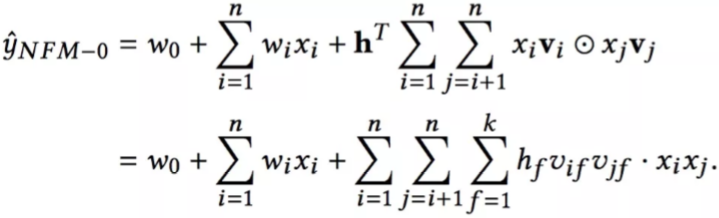
最后一层隐藏层加上一个线性变换，作为结果输出，形式化表达如下：



其中h是中间的网络参数。考虑到前面的各层隐藏层权重矩阵，f(x)形式化如下：



这里的参数为，相比于FM其实多出的参数就是隐藏层的参数，所以FM也可以看做是一个神经网络架构，就是去掉隐藏层的NFM。我们把去掉隐藏层的NFM称为NFM-0，形式化如下：



如果h为全1向量，那么此时NFM就是FM，可以认为NFM就是FM的推广。

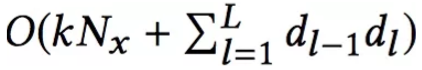
注：实验发现在正则化FM 的时候，使用Dropout比传统的L2正则化还要有效。

h不会增强FM的表征能力，因为该参数可以吸收到特征的embedding 向量中。换言之，就算 h不是全为1的向量，我们也可将其视为FM的等价模型。

如果将全局偏置w0与一阶项综合考虑，NFM的结构图与Wide&Deep极为相似，但NFM中的二阶项与DNN是 **串联结构**。NFM左侧同样可以看做是一个LR模型，但不同于Wide&Deep，NFM左侧的LR模型仅输入单特征，并没有将组合特征送入LR模型，所以也就无需进行额外的特征工程工作。

## 复杂度

NFM相比于FM，复杂度增加在MLP部分。所以NFM的复杂度和Wide&Deep、DeepCross是相同的，形式化如如下：



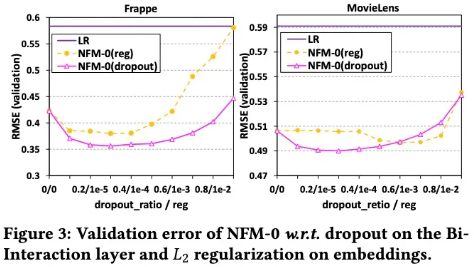
# 性能分析

论文作者围绕三个问题对NFM的表现能力进行分析

## 问题一

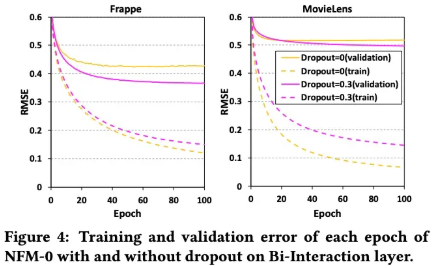
**Q：Bi-Interaction pooling 能否有效捕获二阶交叉特征信息？dropout与batch normalization能否起作用？**

首先将NFM去除隐层，得到NFM-0模型，此时的NFM-0与FM等价。以LR作为对比模型，在不同dropout比例、不同的L2正则化强度下，NFM-0的表现如下：



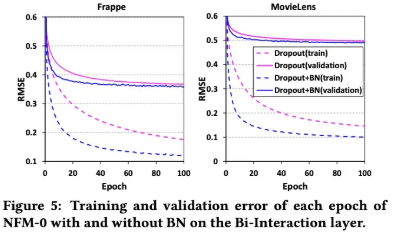
从上述实验可以看出，NFM-0比LR表现好很多，说明Bi-Interaction pooling能够捕获二阶交叉信息。dropout与L2正则对比可以说明，NFM-0更适合使用dropout技术。

下面对比实验，观察dropout是否能够有效缓解模型过拟合问题。



从上图可以看出，dropout能够有效缓解模型过拟合问题。

接下来试验观察batch normalization的效果：



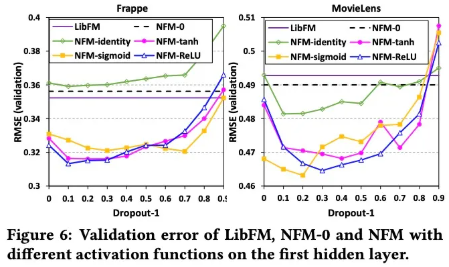
从上图可以看出，batch normalization能够明显加速模型收敛。同时可以看出，结合dropout与batch normalization技术能够有效避免过拟合，提高模型泛化能力。

## 问题二

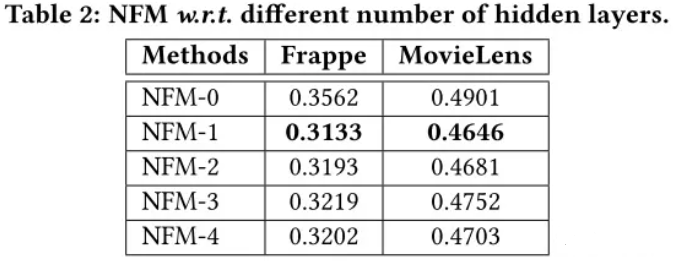
**Q：NFM隐层能否有效的捕获更高阶交叉特征信息？**

NFM使用一层隐层，同时将维度与Bi-Interaction的输出维度保持一致，对比NFM-0、FM（二者其实是等价的）有巨大提升。说明，NFM中的DNN部分能够学习更高阶交叉特征信息，帮助模型提效。

同时，对比不同的激活函数发现，引入非线性激活函数比线性函数（identity）表现更加优异。这也说明了非线性结构（DNN）的必要性。



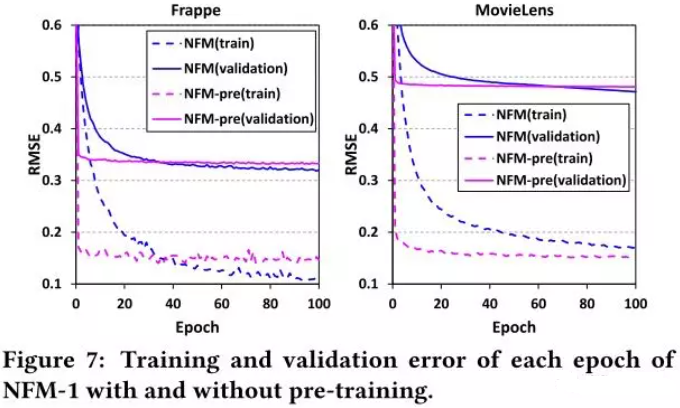
除此之外，作者还对隐层结构进行实验对比。首先比较了不同隐层带来的效果，如table2所示，NFM-1是最佳状态，继续叠加隐层反而效果会有所下降。说明NFM的Bi-Interaction能充分的捕获二阶交叉特征信息，所以只需要接入较浅的隐层结构就能捕获更高阶的信息，从而取得不错的效果。



为了进一步验证Bi-Interaction捕获信息的高效性，作者将Bi-Interaction替换为concatenation。实验结果发现，随着隐层结构越来越深，效果也会越来越好，但是最佳效果仍不如NFM-1。

因为Wide&Deep、DeepCrossing使用FM预训练参数初始化时能够有更好的表现，为了更进一步展示NFM的优越性，作者还考察了NFM参数是否对于预训练敏感。

将FM训练得到的参数作为NFM的初始化参数，对比预训练模式与随机初始化模式的效果。从下图可以看出，预训练只能起到加速收敛的作用，对于最终的结果影响不大。而NFM对于参数初始化不敏感，更容易进行训练优化。

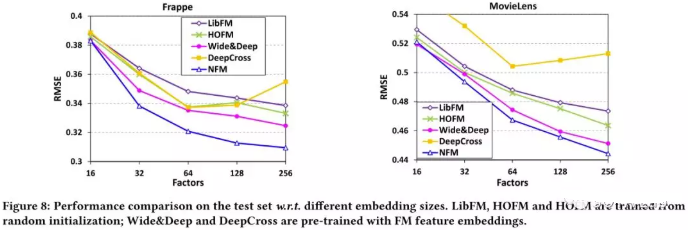


## 问题三

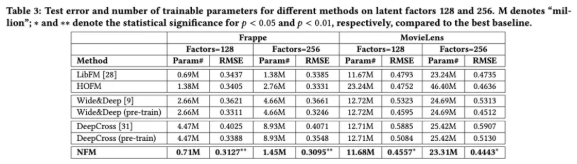
**Q：NFM与目前（2017年）最好的模型（Wide&Deep、DeepCrossing）相比如何呢？**

NFM使用一层隐层，激活函数为relu，与对比模型保持一致，dropout设置为0.5。Wide&Deep、DeepCrossing使用原论文中最佳参数。

通过实验选择最佳的embedding 维度，如下图所示，除了DeepCrossing，其他模型的表现基本一致，所以最终选择128、256作为对比参数。



从最终的对比结果Table3来看，NFM的确表现最佳。



除此之外也能够发现一些其他有意思的现象。

HOFM（使用了更高阶交叉特征的FM）与FM对比可以发现，更高阶的特征交叉能够带来提升，但HOFM采用线性方式拟合高阶特征，这种做法带来的提升效果有限，且模型参数更多。对比HOFM与NFM可以看出，使用非线性方式拟合高阶特征才能带来巨大提升效果，且NFM所引入的参数远远低于HOFM。

Wide&Deep与DeepCrossing在使用预训练的参数进行初始化之后，能够取得更好的效果。DeepCrossing使用了10层网络结构，但是最终的效果却不及FM，这说明DeepCrossing难以训练优化，网络并不是越深越好。

总结，NFM结构简单，参数少，训练速度快（Bi-Interaction Layer的复杂度为 ），效果好。

# 实践

参考：

<https://github.com/jpegbert/code_study/tree/master/NFM>

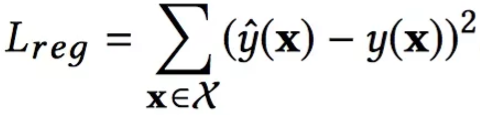
# 实践Tricks

## 损失函数

NFM可以用于分类、回归、ranking问题，对应着不同的损失函数。

1. 回归。采用square loss；
2. 分类。Hinge Loss、log loss或交叉熵损失函数；
3. ranking。Contrastive max-margin loss

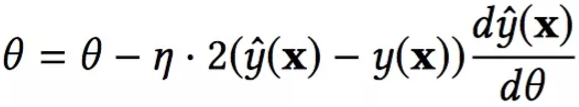
注：论文中以回归问题为例，使用square loss，形式化如下：



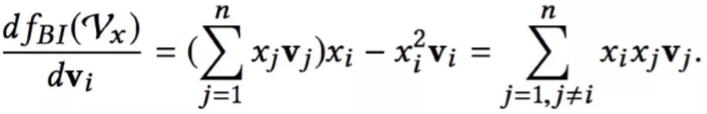
这里并没有正则化项，因为作者发现在NFM中使用Dropout能够得到更好的效果。

## 参数估计

在NFM模型中，使用mini-batch Adagrad来进行参数估计，Adagrad是SGD的变体，特点是每个参数都有自己的学习速率。然后让参数沿着目标函数负梯度的方向进行更新，是下降快的方向，形式化如下：



这里唯一需要指出的是Bi-Interaction在求梯度时是怎么做的：



所以，NFM的训练依旧可以是端到端的训练，只需要把Bi-Interaction插入到网络中即可。

## 过拟合

NFM面临过拟合的风险，NFM作者使用了dropout与batch normalization技术来缓解过拟合问题。实验表明，结合使用这两种技术能有效的避免过拟合。

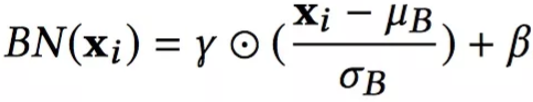
### Dropout

Dropout在训练过程中随机丢掉一些神经元，那么再一次参数更新中也就只会更新部分参数，增加了模型的抗过拟合能力。在NFM 中，dropout一般是用于Bi-Interaction Layer输出之后，对于后续的每一层隐层都可使用。

注：在测试阶段，Dropout是不使用的，所有的神经元都会激活。

### Batch Normalization

DNN的训练面临很多问题。其中一个就是协方差偏移(covariance shift)，意思是：由于参数的更新，隐藏层的输入分布不断的在变化，那么模型参数就需要去学习这些变化，这减慢了模型的收敛速度。Batch Normalization可以对输入进行处理（把每层神经网络任意神经元的输入值的分布强行拉回到均值为0方差为1的标准正态分布），从而加速模型收敛，在NFM中，Batch Normalization用于Bi-Interaction的输出，以及后面所有隐藏层的输入。Batch Normalization可形式化如下：



其中：μ\_B 是该mini-batch的均值，σ\_B 是mini-batch的方差，对于隐藏层的输入，BN在mini-batch数据上，把输入转换成均值为0，方差为1的高斯分布。gamma和beta是两个超参数，用来控制normalization的缩放和平移，扩大模型的表达能力，如果模型发现不应用BN操作更好，那么就可以通过学习这两个参数来消除BN的影响。对于NFM，对Bi-interaction层的输出也进行BN操作。

注：在测试预测的时候，BN操作同样有效，这时的均值和方差在整个测试集上来进行计算。

注：出现网络收敛越来越慢的原因：深层神经网络在传入激活前的输入（W\_i \*z\_{i-1}+b\_i），在训练过程中分布会逐渐往非线性方向偏移，造成低层神经网络梯度消失。

### 附加层相对顺序

同时使用dropout，与batch normalization技术，需要注意两者的使用顺序。在Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift一文中，作者提到: 需要为激活函数提供需要的数据分布。所以，应该在Bi-Interaction Layer之后接入batch normalization，然后直接进行dropout。需要注意的是，Bi-Interaction Layer是没有激活函数的。后续隐层需要进行batch normalization调整数据分布，然后再加上激活函数，最后使用dropout技术。

## 原paper中得出的结论

论文中对比了FM、Wide&Deep模型。此处，记录下作者得出的一些重要的结论：

（1）Dropout在NFM中可以有效的抑制过拟合

（2）Batch Normalization在NFM中可以加快训练速度；

（3）NFM使用一个隐藏层得到了好的效果；

（4）如果用FM来pre-train（预训练）嵌入层（Embedding），NFM会收敛的非常快，但是NFM终的效果并没有变好。说明NFM对参数有很好的鲁棒性；

（5）模型性能基本上随着Factor（Embedding向量的维度）的增加而提升。

# 总结

NFM最主要的创新点在于Bi-Interaction池化部分来代替其他模型使用的拼接，具体来说：

1. 引入了Bi-Interaction Layer，采用Pooling操作将多个embedding vector转化成 一个vector；
2. 加入了NN，用于发现高阶非线性组合特征

对比于Wide&Deep模型与Deep Crossing模型完全依赖深度学习有意义的特征交互，NFM使用特征交叉池化的方式捕获了较低级别的二阶特征交互，这比拼接提供的信息更多。这极大地促进了NFM的后续隐藏层以更容易的方式学习有用的高阶特征交互。

参考

https://mp.weixin.qq.com/s/qmkMJJkMqumbtynM4cLatw

https://mp.weixin.qq.com/s/u2qk33WI2wxZQZ1wpxulvQ

https://mp.weixin.qq.com/s/4hOHyDN-fDVXi3KV8UX2Cw

https://mp.weixin.qq.com/s/1en7EyP3C2TP3-d4Ha0rSQ

https://mp.weixin.qq.com/s/h2sB1cne4JlA4huc6N5ihA

<https://mp.weixin.qq.com/s/1sWYlzIydiLAPMBnr-a5sQ>

<https://mp.weixin.qq.com/s/29mEv-trmrvIx_wqfu5EYw>

https://mp.weixin.qq.com/s/In1F5OksL5vE1voeSC6tPg