MMoE理论与实践

MMoE论文题目是：《Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-Experts》。MMoE是一种处理多任务的框架，是基于Mixture-of-Experts (MoE)模型，提出的一种显式建模任务相关性的模型。

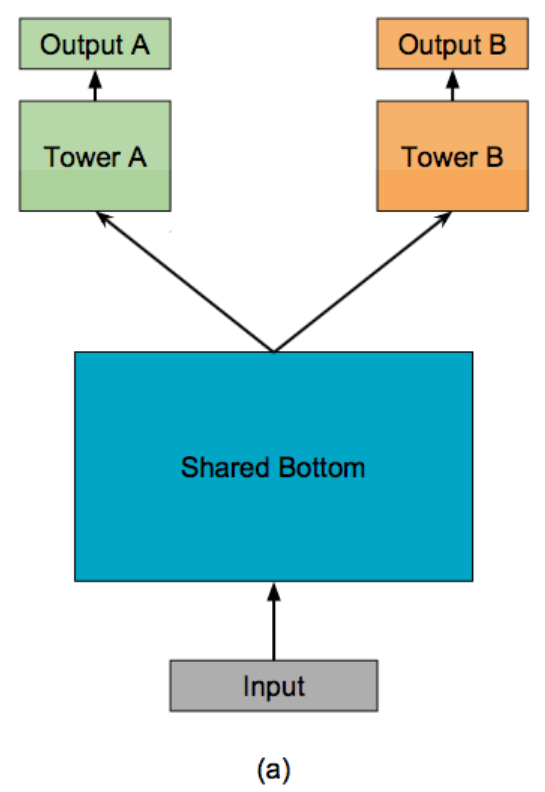
# 背景

近年来，深度神经网络的应用越来越广，如推荐系统。推荐系统通常需要同时优化多个目标，如电影推荐中不仅需要预测用户是否会购买，还需要预测用户对于电影的评分，再比如电商领域同时需要预测物品的点击率CTR和转化率CVR。因此，多任务学习模型成为研究领域的一大热点。

许多多任务学习模型取得了不错的效果，但是实践中多任务学习模型并不总比单任务模型效果更突出。这主要是因为不同任务之间的相关性低（如数据的分布不同等等）导致的。

# Shared-bottom多任务模型

Shared-bottom多任务模型是最一般的多任务学习模型，其模型框架如下：

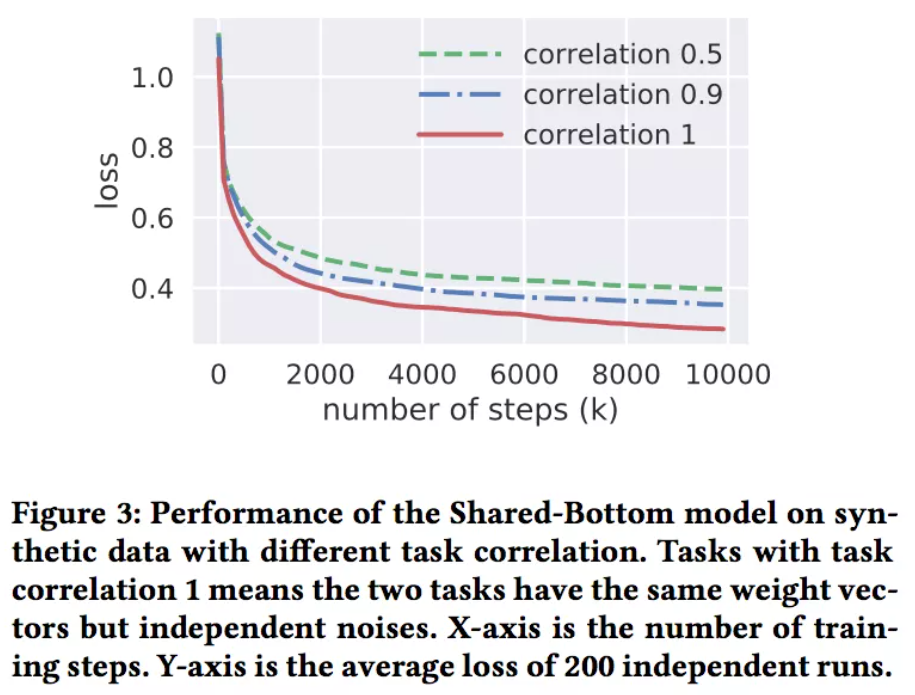


对于不同的任务，底层的参数和网络结构是共享的，然后再根据任务的个数在上面划分出多个tower network来分别学习不同的目标。假设底层输出是f(x)，那么第k个任务的输出yk为：



其中f表示shared-bottom network， hk表示第k个tower network，yk是最终的输出。

作者给出相关性不同的数据集上多任务的表现，其也阐述了，相关性越低，多任务学习的效果越差：



可以看到的是，随着任务相关性的提升，模型的loss越小，效果越好，从而印证了前面的猜想。

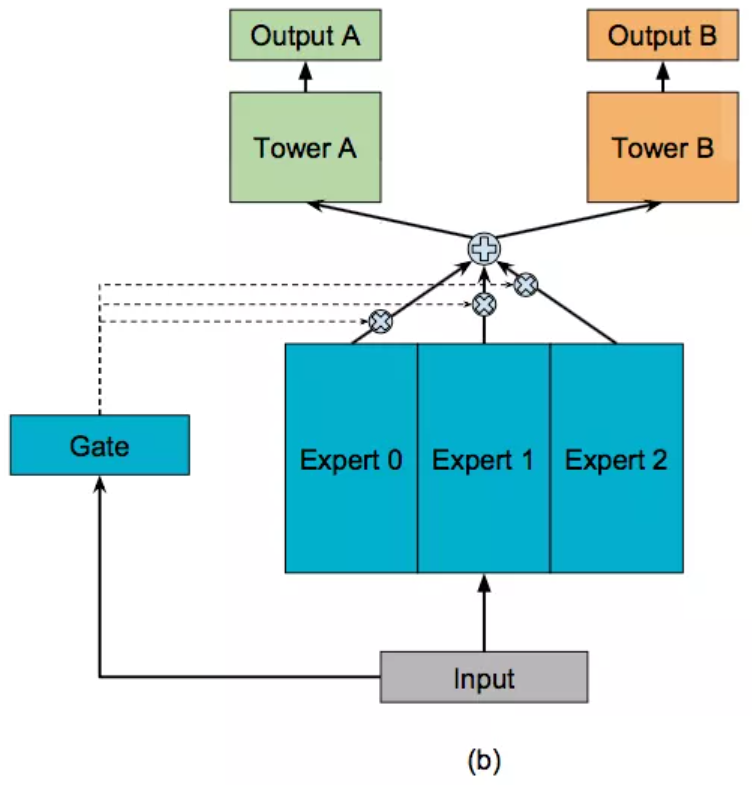
对于Shared-bottom多任务模型，它的优缺点如下：

优点：降低overfit风险，利用任务之间的关联性使模型学习效果更强

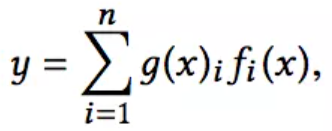
缺点：任务之间的相关性将严重影响模型效果。假如任务之间相关性较低，模型的效果相对会较差。

# MoE多任务模型

Mixture-of-Experts(MoE)模型（文中后面称作 One-gate Mixture-of-Experts (OMoE)）的模型结构如下图所示：



可以看到，在MoE中，相较于一般的多任务学习框架，共享的底层分为了多个expert，每个Expert其实就是一个feed forward network。同时设置了Gate，Gate为每个Expert输出一个标量，使不同的数据可以多样化的使用共享层。因此整个模型就像一种Ensemble的方式，由专家给出自己的“建议”，gate根据input去给每个专家一定的权重，然后最终根据加权后的专家意见给出最终的结果。此时共享层的输出可以表示为：



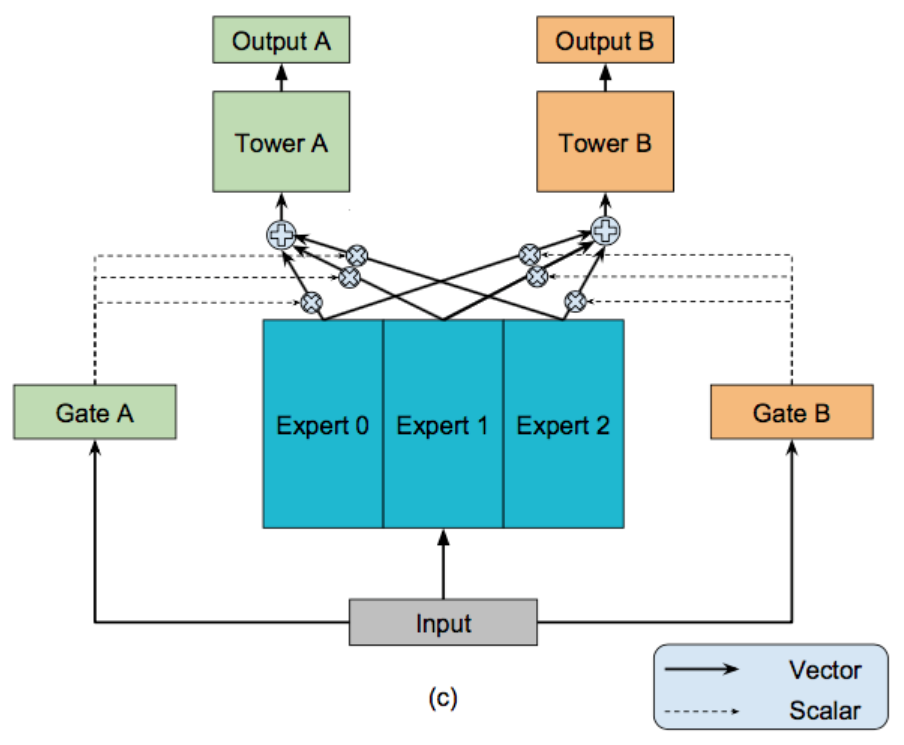
其中fi代表第i个expert的输出，gi代表第第i个expert对应的权重，是基于输入数据得到的，计算公式为g(x) = softmax(Wgx)。

MoE的主要目标是实现条件计算，对于每个数据而言，只有部分网络是活跃的，该模型可以通过限制输入的门控网络来选择专家网络的子集。

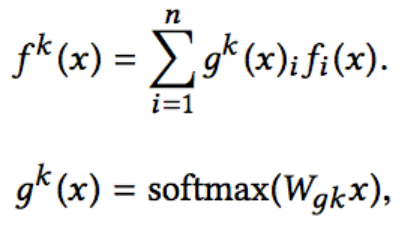
# MMoE模型

MoE能够实现不同数据多样化使用共享层，但针对不同任务而言，其使用的共享层是一致的。这种情况下，如果任务相关性较低，则会导致模型性能下降。

为了解决任务之间相关性降低导致模型效果下降的问题。paper作者在MoE的基础上进行了改进，在相对于shared-bottom结构不明显增加模型参数的前提下捕获任务的不同，提出了Multi-gate Mixture-of-Experts (MMoE)模型，MMoE模型为每一个task设置了一个gate，用这些gate控制不同任务不同专家的权重，旨在使得不同任务和不同数据可以多样化的使用共享层。其模型结构如下图所示：



此时每个任务的共享层的输出不同，第k个任务的共享层输出计算公式如下：



随后每个任务对应的共享层输出，经过多层全连接神经网络得到每个任务的输出：



从直观上考虑，每个Gate网络都可以根据不同任务来选择专家网络的子集，所以即使两个任务并不是十分相关，那么经过Gate后也可以得到不同的权重系数，此时，MMoE可以充分利用部分expert网络输出的信息，近似于单个任务学习模型；而如果两个任务相关性高，那么Gate的权重分布相差会不大，会类似于一般的多任务学习。

从某种角度上来讲，MMoE的厉害之处在于它的expert可以定义成任意一种单独的模型，可以说，MMoE是一个框架，而非简简单单的一个模型！这就使得我们在expert上可以灵活自由地实现自己想要的模型设计。

# 讨论

## 什么是多任务学习？

传统的机器学习，深度学习任务是只关注一个损失函数，但是在复杂的业务场景下，常常需要关注多个指标，逐一攻克各个指标难免捉襟见肘。比如在推荐场景下，推荐排序不但关心用户的点击率，也在乎后续的用户对物品的满意度指标。换句话说，好的推荐视频不但要让你点了（点击率），还要你看的爽（观看时长，评分）；好的推荐商品不仅要让你点了（点击率），还要让你买了（转化率）…

## 为什么多任务学习可以有效拟合多个目标？

* 多任务一起学习时，会互相增加噪声，从而提高模型的泛化能力；
* 多任务相关作用，逃离局部最优解；
* 多任务共同作用模型的更新，增加错误反馈；
* 降低了过拟合的风险；
* 类似 ESMM，解决了样本偏差和数据稀疏问题，未来也可以用来解决冷启动问题。

## MMoE模型的优势

一方面，因为gating networks通常是轻量级的，而且expert networks是所有任务共用，所以相对于论文中提到的一些baseline方法在计算量和参数量上具有优势。

另一方面，相对于所有任务公共一个门控网络(One-gate MoE model)，MMoE中每个任务使用单独的gating networks。每个任务的gating networks通过最终输出权重不同实现对experts的选择性利用。不同任务的gating networks可以学习到不同的组合experts的模式，因此模型考虑到了捕捉到任务的相关性和区别。

概括下来就是如下两点：

* MMoE模型增加了多个Gate kernels，可以有效的捕捉到子任务的相关性和区别；
* 相对于MoE模型，MMoE模型并没有增加太多的参数，从论文的效果来看有明显的提升。

## MTL的主要形式有哪些？

MLT（Multi Task Learning）主要有两种形式，一种是基于参数的共享，另一种是基于约束的共享。

### hard参数共享

参数共享的形式在基于神经网络的 MLT 中非常常见，其在所有任务中共享隐藏层并同时保留几个特定任务的输出层。

这种方式有助于降低过拟合风险，因为同时学习的任务越多，模型找到一个含有所有任务的表征就越困难，从而过拟合某个特定任务的可能性就越小，ESMM 就属于这种类型的。

### soft参数共享

每个任务都有自己的参数和模型，最后通过对不同任务的参数之间的差异施加约束。比如可以使用L2进行正则、迹范数（trace norm）等。

## MTL的适用条件

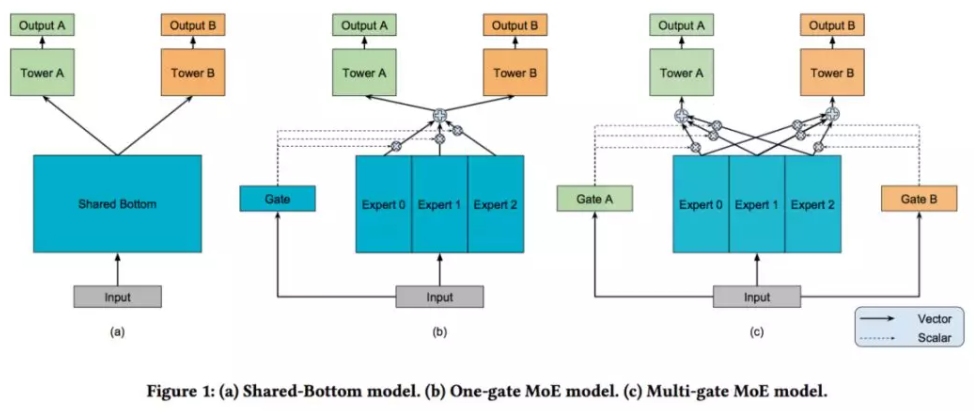
MTL的目标在于通过利用包含在相关任务训练信号中特定领域的信息来提高泛化能力。

什么是相关任务呢？有以下几个不严谨的解释：

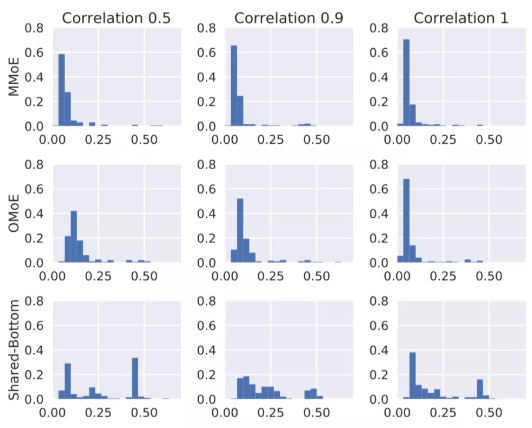
* 使用相同特征做判断的任务；
* 任务的分类边界接近；
* 预测同个个体属性的不同方面比预测不同个体属性的不同方面更相关；
* 共同训练时能够提供帮助并不一定相关，因为加入噪声有时也可以增加泛化能力。

# 实验结果

先回顾上面介绍的三种多任务学习的架构：



## MTL模型在不同相关性任务下的loss分布



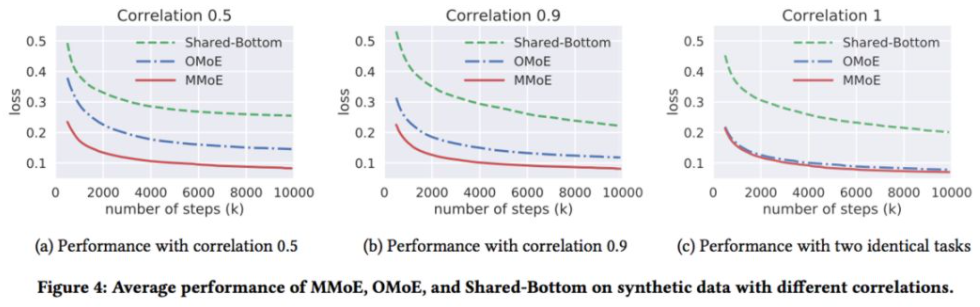
论文比较了不同MLT模型在不同相关性任务下的loss分布，其可以反应模型的鲁棒性，从图中可以看出：

* shared-bottom模型的表现方差高于OMoE和MMoE；
* 虽然在correlation=1的数据集下，OMoE和MMoE的表现接近，但是在correlation=0.5的数据集下，OMoE的表现有明显的下降，这也证明了MMoE的multi-gate structure能够有效的解决任务差异冲突带来的局部最小值。

实验分为三部分：人工合成数据集（即本文第二部分所介绍的人工生成的数据集）、UCI census-income dataset（UCI人口普查）和Large-scale Content Recommendation（Google大规模内容推荐）。

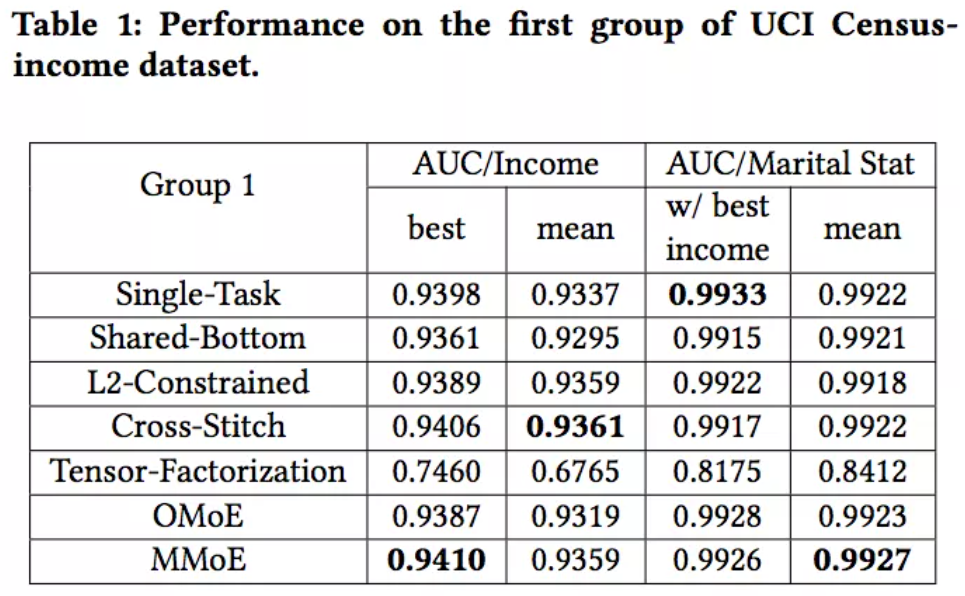
## 人工合成数据集-实验结果

在真实数据集中，我们无法改变任务之间的相关性，所以不太方便进行研究任务相关性对多任务模型的影响。论文中人工构建了两个回归任务的数据集，然后通过两个任务的标签的Pearson相关系数来作为任务相关性的度量。在工业界中通过人工构造的数据集来验证自己的假设是个有意思的做法。



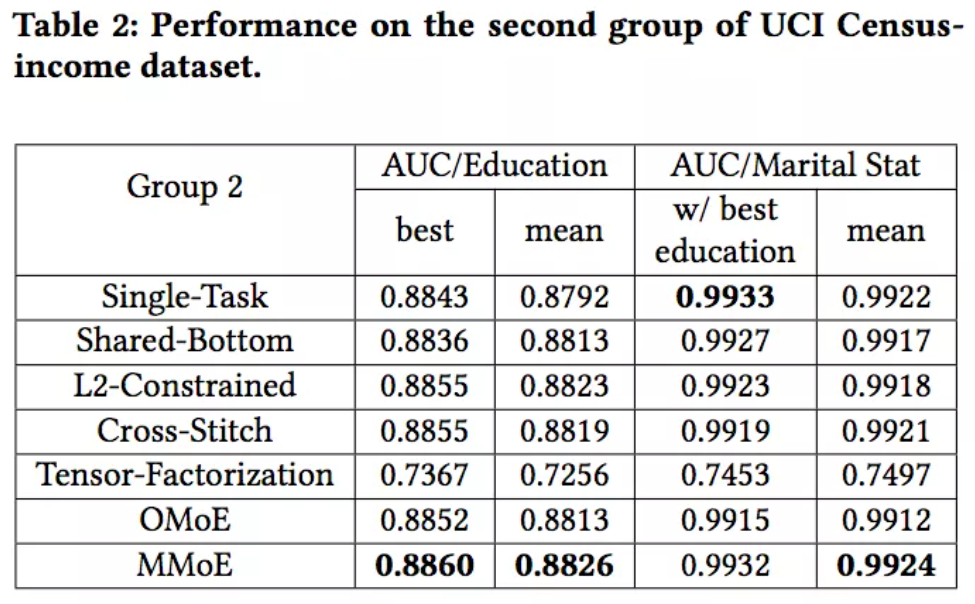
可以看到，不管多任务之间的相关度是0.5，0.9和1，MMoE的表现都要好于MoE，MoE的表现要好于Shared-bottom。

## UCI census-income dataset-实验结果

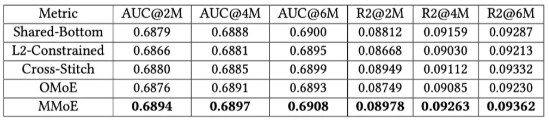


## Large-scale Content Recommendation-实验结果

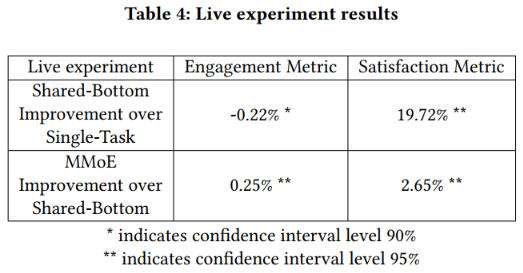
在Google大规模内容推荐这个实验中，作者使用了超过300亿的用户行为数据作为模型训练的输入，并且在100万用户行为进行离线实验评估，而所优化的任务分别是交互和满意度。



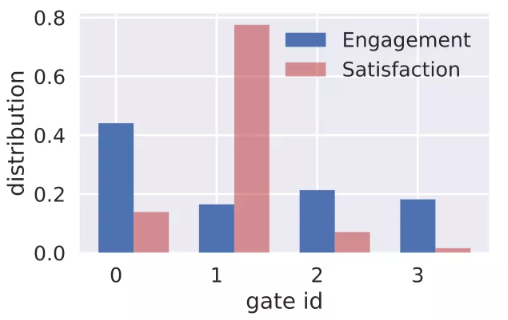
大型推荐系统的表现：



实验结果表明，MMoE在不同的参数设置上的实验AUC均超过了其他Baseline，这就证明了MMoE模型的有效性。并且作者对MMoE做了上线，也得到了比较不错的结果：



Gate网络在两个任务的不同分布：



论文展示了gate在两个任务上的分布，satisfaction任务的label比engagement任务的label更加稀疏，所以satisfaction的gate分布更加偏向于单个expert。

# 实践

参考：

https://github.com/jpegbert/code\_study/tree/master/MMoE

包含多个版本，有Tensorflow和Pytorch实现的版本

# 总结

作者提出了一种新颖的多任务学习方法——MMoE，其通过多个Gate网络来自适应学习不同数据在不同任务下的与专家子网的权重关系系数，从而在相关性较低的多任务学习中取得不错的成绩。

共享网络节省了大量计算资源，且Gate网络参数较少，所以MMoE模型很大程度上也保持了计算优势。

参考

<https://mp.weixin.qq.com/s/1y0biwfASqErpe5iak8l2A>

<https://mp.weixin.qq.com/s/13MyAqCn9RNxCXRsUC8p_Q>

<https://mp.weixin.qq.com/s/PuzIdXSDGzRIXxg9f6D_tg>

<https://mp.weixin.qq.com/s/7VtiPi-40hkG6YjMx2dgkw>

<https://mp.weixin.qq.com/s/AzQePtEpHNotVn647SQUwQ>

<https://mp.weixin.qq.com/s/AzQePtEpHNotVn647SQUwQ>

<https://mp.weixin.qq.com/s/cBy0Y5xDtkc6PxhF1HNomg>

<https://mp.weixin.qq.com/s/4vvjd9jWIcdIOQbqAaFmfw>

<https://github.com/ShaoQiBNU/Google_MTL>