谷歌视频推荐多任务排序模型——MMoE

原创 炮屁儿 独立团丶 4月6日

一、创新点

1、论文针对工业界视频推荐领域提供了一种端到端的大规模多目标排序模型

2、引入MMoE模型（对MoE模型的扩展）来提升系统的排序效果

3、采用类似Wide&Deep的架构来解决position bias

二、论文背景

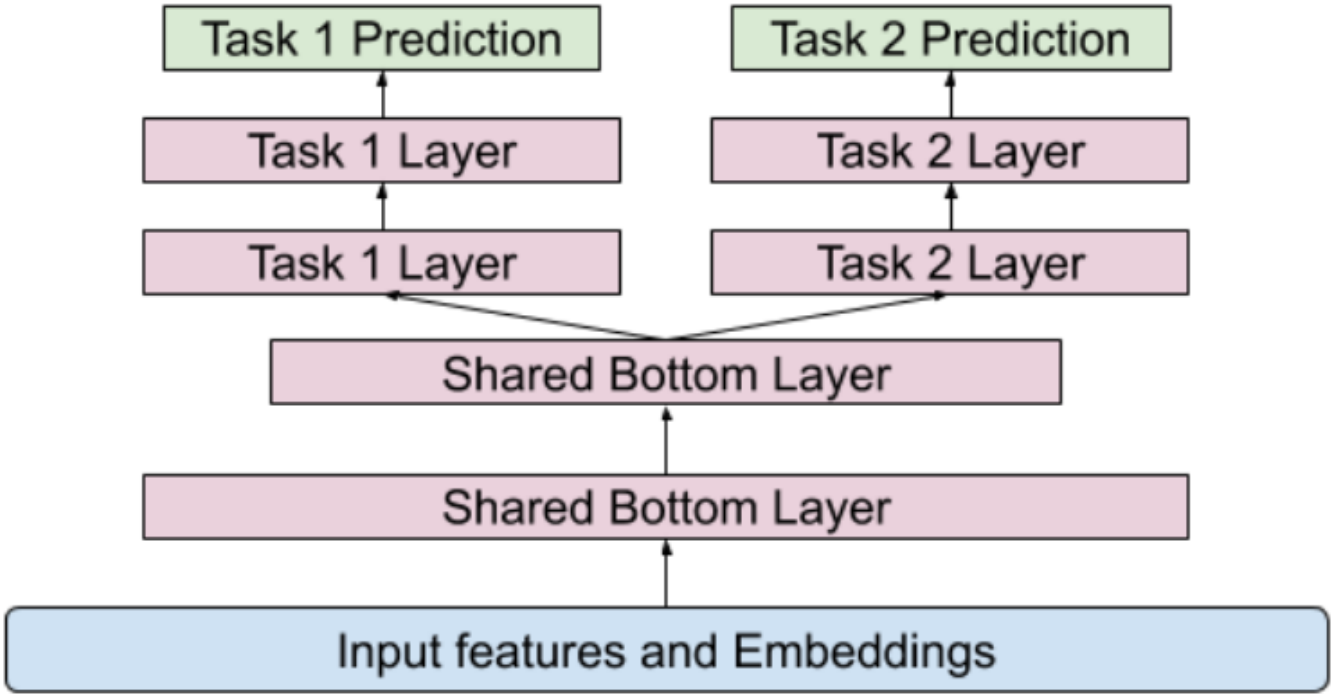
推荐系统在给用户推荐一些高点击的内容的同时，也需要对这些内容的质量有一定的保证，比如用户的使用时长、用户打分、用户评论等，所以通常推荐领域的模型同时会对多个目标进行优化，而如何对这多个目标进行同时的优化正是论文关注的重点，在以往的一些模型当中，通常优化一方面的效果就会损失另一方面的效果，所以这是一个trade-off的过程。除此之外无论是搜索、推荐还是广告，都存在一个的潜在的bias问题，下面以搜索广告为例，比如在百度中搜索"四川旅游"，搜索结果如下

展示的结果中分为多个区域，左侧、右侧、底部等，通过对业务数据的分析可以发现左侧的位置时相对来说最优的，其中越靠近上方的位置点击率就越高，其中首位的广告点击率要远高于其他位置的点击率，这与用户存在一个潜在的使用习惯有关，通常用户会更加偏好点击首位的广告（即使首位的广告相较于其他位置的广告来说没有那么match）。由于存在上述的问题，商业系统的Rank模型都是基于各种业务的日志数据进行训练的，由于日志中混入了这种bias，导致模型会被带偏（指偏离真实的数据分布），将这种模型部署到线上之后模型又会改变线上原本的分布，这就导致存在一个闭环（bias的影响会持续恶化），针对这个问题论文采用了类似于Wide&Deep的架构，引入了广告展现位置相关的特征来学习这种bias。

三、模型结构

1、Base模型

论文中提到的Base模型大致可以理解为多任务模型底层参数共享，上层针对不同任务再训练各自的隐藏层，具体结构可以参考下图

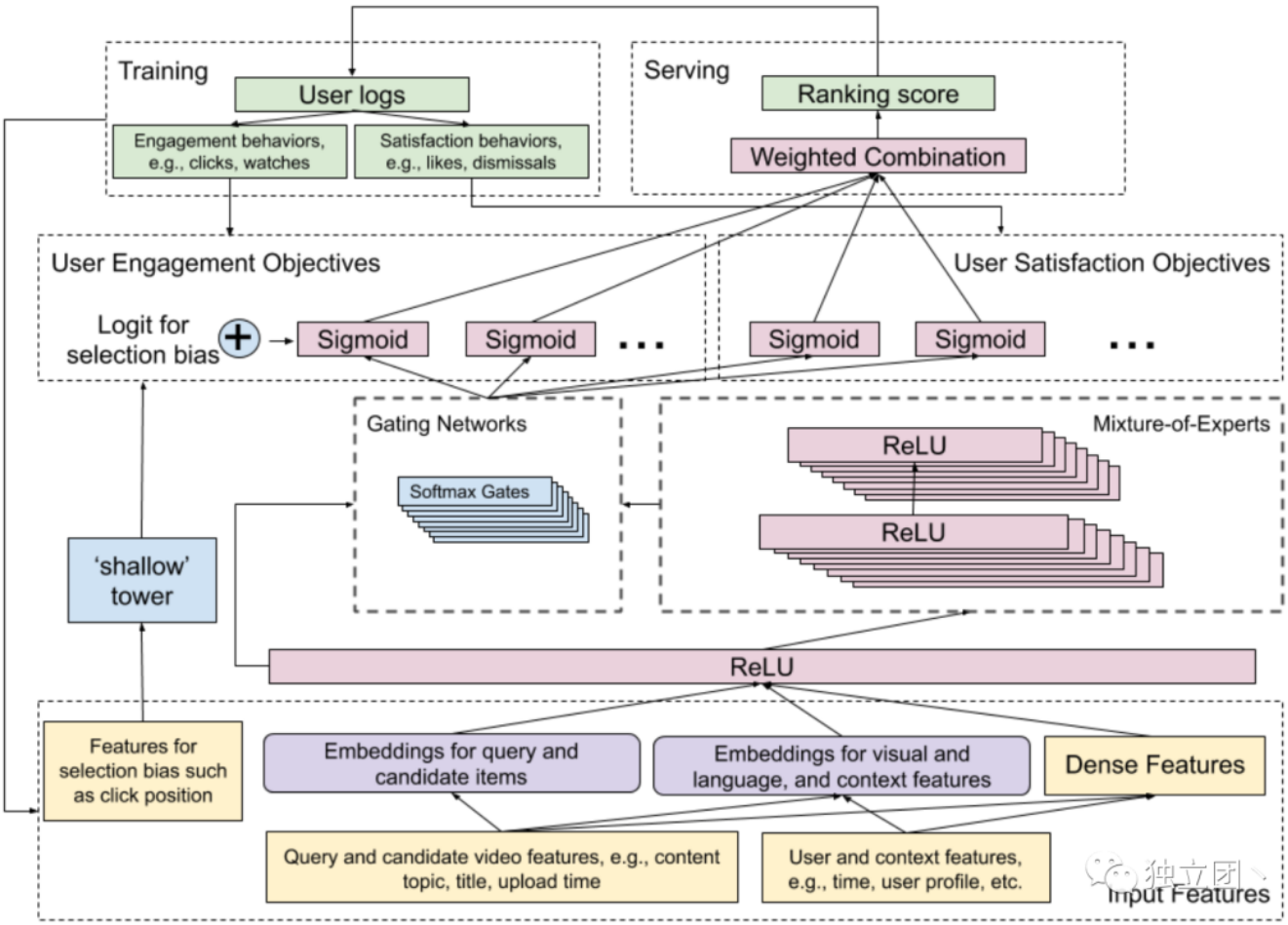


(a) Shared-Bottom Model with shared bottom hidden layers and separate towers for two tasks.

这种模型结构存在一个非常直观的问题，如果任务一和任务二之间的相关性不是特别高的话，多任务训练的结果可能不会非常好，如果两个任务分别是点赞和转发这种，那么这种任务是比较适合联合训练的；但是如果两个优化目标之间的相关性是比较差的时候，可能会导致各个目标学习的都不够充分，在互相影响之下都被带偏了导致最终整体效果不尽如人意，当然上面说的相关性不只限于直观感受上的相关性，同时也包括训练数据等方面导致看似相似任务的不相关。之前阿里提出的ESSM模型就是基于这种结构的CTR-CVR模型，我们组当时也是尝试了ESSM模型来联合训练CTR模型和CVR模型，但是线上效果不太好，也是调了非常久，最终分析可能原因之一就是可能训练数据导致两个任务之间的相关性存在一定问题最终影响了效果，最后这个模型也是没能最终应用到线上，如果哪个小伙伴有相关成熟的ESSM应用方案，也可以大家一起讨论下。

2、MMoE模型

首先来看MMoE模型线上的结构，如下图所示



整体流程（闭环）大致包括：

（1）从离线日志（user log）当中提取特征（分不同fields）

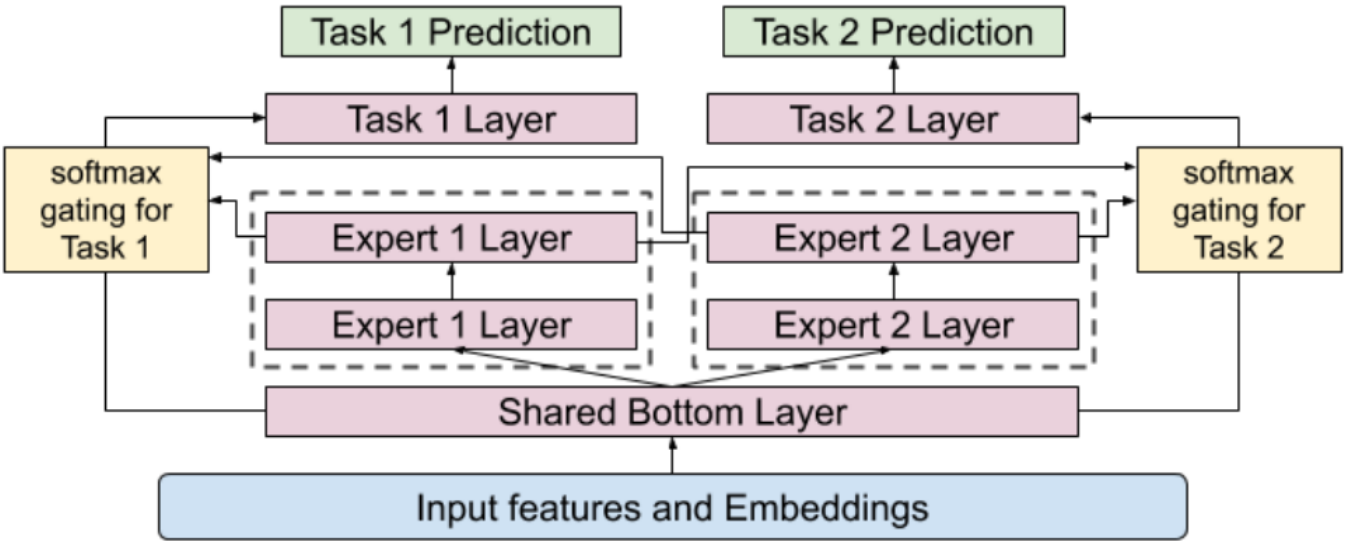
（2）将提取好的特征喂到整个网络（包括左侧shallow tower和右侧的main tower）中进行训练

（3）将训练好的模型部署到线上，对线上的请求进行响应，得到的结果会落到日志当中，供后续模型训练使用

从图中可以发现，MMoE包括两个部分，即左侧的shallow tower部分和右侧的main tower部分，论文中提到的采用类似Wide&Deep模型结构就是指这两个tower，其中shallow tower可以对应Wide部分，main tower对应的是Deep部分，接下来我们分别对这两部分进行讨论。

Main Tower部分

相较于Base模型的底层部分，Main Tower采用了一种模型融合的思想（加权求和），在MoE部分存在n个Expert网络，每个Expert网络的输出最终会经过Gating Network进行加权平均（比较简单的线性加权，Attention的思想）。在论文中每个Expert网络采用的就是普通的多层全连接网络，同时为了进一步提升模型的训练效率，其实不同的任务的n个Expert也是共享的（这部分应该是出于业务需求的考虑，具体可以根据自己的实际情况来进行相应的调整）。对于不同的任务通过相应的Gating Network来对不同的Expert赋予不同的权重，使得部分Expert“专注于各自擅长的任务”。具体Main Tower模型结构如下图所示



(b) Multi-gate Mixture-of-Expert Model with one shared bottom layer and separate hidden layers for two tasks.

Shallow Tower部分

前面我们提到了推荐、搜索、广告领域广泛存在的position bias问题，文章中shallow Tower部分主要作用就是消除Position bias的影响。这部分模型的输入是与position bias相关的一些特征（如广告展现时的排序位置、用户机型等），文章提到了用户机型也算在这部分feature当中，比较直观的认知是不同机型的尺寸的差异可能会对这些position bias有所影响。实际上我们线上的PC搜索模型当中也是采用同样的架构，即Main Tower + Shallow Tower这种类似Wide&Deep的模型结构，主要也是出于position bias的影响（听说头条对这部分也是有一定的改进，有了解的小伙伴可以一起交流下）。shalow tower的输出部分是当做main tower的engagement部分的bias，这里主要是因为真正受position bias影响的是user engagement部分（engagement行为定义为点击、观看这种；satisfaction行为定义为点赞、打分这种），也就是说主要是用户点击率受position bias影响，而用户真正点赞、打分这种受position bias影响是比较小的。除此之外需要注意的细节是，当我们在训练的时候加入的位置特征可以从用户日志中获得，但是在线上部署预测的时候，由于是根据我们ranking部分的结果排序后才能得到具体的展示位置，所以在线上预测时通常给该部分特征取一个约定值（0，1或者空值都是可以的）具体Shallow Tower模型结构如下图所示

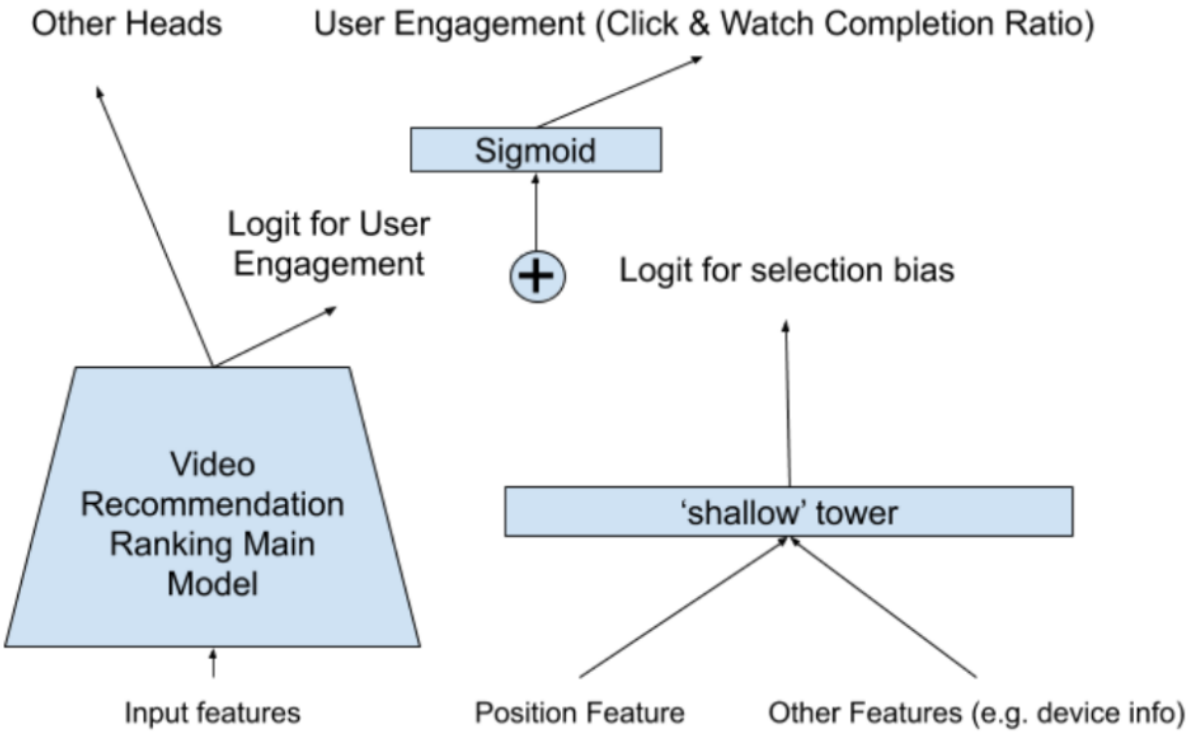


Figure 3: Adding a shallow side tower to learn selection bias (e.g., position bias).

四、实验评估

模型在评估的时候主要是应用在YouTube的视频推荐场景下，具体如下

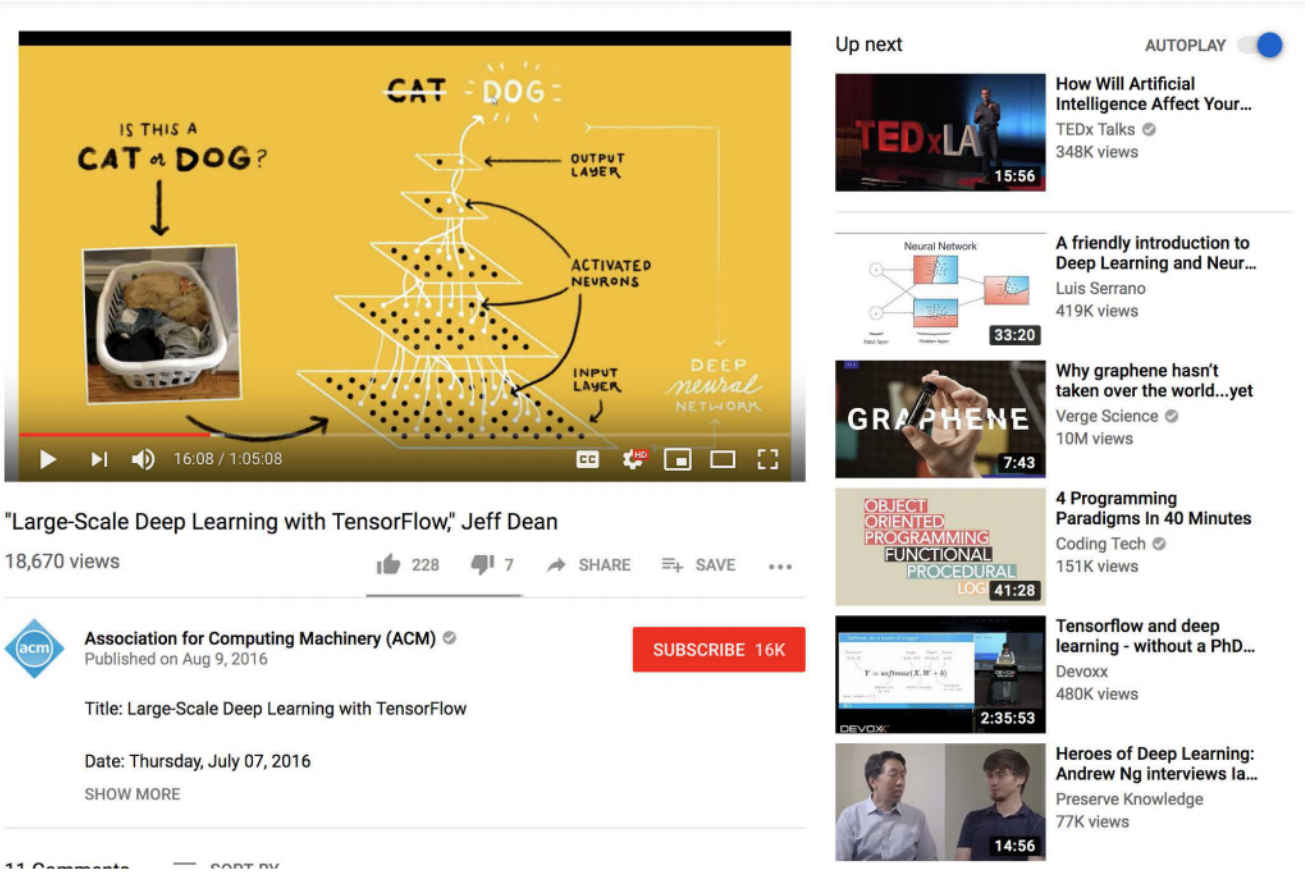


Figure 4: Recommending what to watch next on YouTube.

具体的实验结果如下所示

Model Architecture	Number of Multiplications	Engagement Metric	Satisfaction Metric
Shared-Bottom	3.7M	/	/
Shared-Bottom	6.1M	+0.1%	+ 1.89%
MMoE (4 experts)	3.7M	+0.20%	+ 1.22%
MMoE (8 Experts)	6.1M	+0.45%	+ 3.07%

Table 1: YouTube live experiment results for MMoE.

从上述结果来看MMoE模型在两个任务上的提升都是比较明显的，同时调整一些系统的超参数也能带来比较明显的提升，具体分析过程可以参考论文

五、延伸思考

文章在最后还对很多工程方面的问题与挑战进行了一些阐述，其中涉及到了工业界广泛面临的一些问题，比如大规模稀疏数据、模型的分布式训练、训练数据中存在的bias、有效性和效率之间的trade-off等。我觉得工业界和学术界之间的差别主要是体现在这些方面，学术界可以相对轻松一些的提出一个非常复杂的模型并且声称在某某测试集上达到了非常好的效果，完爆现有的state-of-art，但是实际上却很难部署到真正的工业环境中，从训练数据的获取到模型训练、线上部署都有很多值得深入思考的地方，所以这也的确是算法工程师需要深入思考的问题。文章的最后还给出了他们未来可能尝试的一些方向和思路，如尝试新的能够兼顾稳定性、效果以及开销的多任务模型；尝试能够自动发现潜在bias并解决的模型结构；模型压缩方案等

六、总结

MMoE是一篇工程性较强的Rank领域的论文，秉承谷歌出品必属于精品的思路还是非常值得深入研读的，本文也只是对该论文进行一个简单的介绍，论文中涉及到了非常多的细节内容，包括多路召回方面的一些阐述都能让人从中受益良多，所以还是需要自己精心去多读几遍，想来肯定是会大有益处的。

阅读原文