Deep Neural Networks for YouTube Recommendations

ABSTRACT

是目前规模最大、最成熟的行业推荐系统之一。在本文中，我们对系统进行了高层次的描述，并针对深度学习带来的显著性能改进进行了讨论。本文按照经典的两阶段信息检索二分法进行了拆分：首先，我们详细描述了一个深层次候选生成模型，然后描述了一个独立的深层次排序模型。我们还提供了从设计、迭代和维护具有巨大面向用户影响的大规模推荐系统中获得的实践经验和见解。

1.INTRODUCTION

YouTube是世界上最大的视频内容创建、共享和发现平台。YouTube推荐网站负责帮助超过10亿用户从不断增长的视频库中发现个性化内容。在本文中，我们将重点介绍最近在YouTube视频推荐系统上进行的大量影响深度学习。图1说明了YouTube移动应用程序主页上的建议。

从三个主要角度来看，推荐YouTube视频极具挑战性：

规模：许多现有的推荐算法被证明可以很好地解决小问题，但无法在我们的规模上运行。高度专业化的分布式学习算法和高效的服务系统对于处理YouTube庞大的用户群和语料库至关重要。

新鲜度：YouTube有一个非常动态的语料库，每秒上传数小时的视频。推荐系统应该能够对新上传的内容以及用户所采取的最新行动做出足够的响应。从探索/开发的角度可以理解将新内容与成熟的视频进行平衡。

噪音：YouTube上的历史用户行为固有地难以预测，因为它的稀疏性和各种不可观测的外部因素。我们很少了解用户满意度的基本事实，而是对噪声隐式反馈信号进行建模。此外，与内容相关联的元数据在没有定义良好的本体的情况下结构不佳。我们的算法需要对训练数据的这些特殊特征具有鲁棒性。

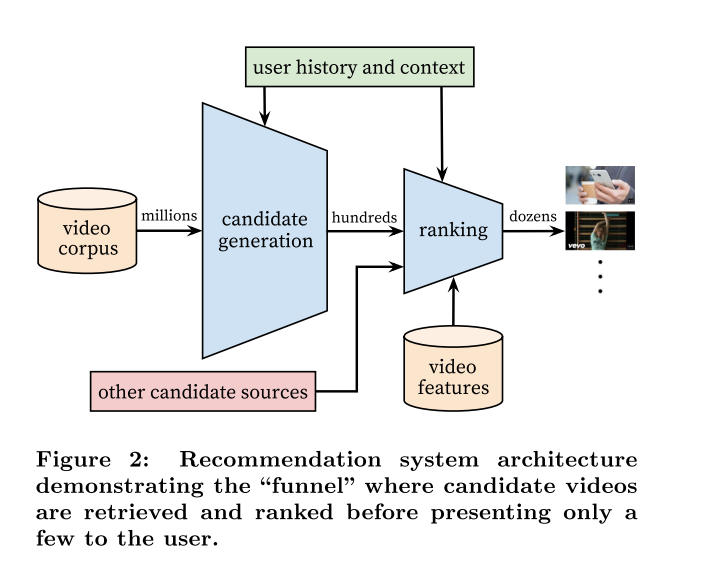
与谷歌其他产品领域相结合，YouTube已经经历了一个基本的模式转变，将深度学习作为几乎所有学习问题的通用解决方案。我们的系统是建立在Google Brain[4]之上的，它最近被开源为TensorFlow[1]。TensorFlow提供了一个灵活的框架，用于使用大规模分布式训练来试验各种深层神经网络结构。我们的模型学习了大约10亿个参数，并接受了上千亿个示例的培训。

与大量的矩阵分解方法的研究相比[19]，在推荐系统中使用深层神经网络的工作相对较少。神经网络-作品用于推荐新闻在[17]，引文在[8]和审查评级在[20]。协同过滤在[22]中被描述为一个深层神经网络，在[18]中被描述为自动编码器。Elkahky等。用于跨域用户建模的深度学习[5]。在基于内容的环境中，Burges等人使用深度神经网络推荐音乐[21]。

本文的组织结构如下：系统概述见第2节。第3节更详细地描述了候选生成模型，包括如何对其进行培训并用于提供建议。实验结果将显示该模型如何从隐藏单元的深层和额外的非均匀信号中获益。第4节详细介绍了排名模型，包括如何修改经典的逻辑回归来训练预测预期观察时间（而不是点击概率）的模型。实验结果表明，在这种情况下，隐藏层深度也是有帮助的。最后，第5节介绍了我们的结论和经验教训。

2.SYSTEM OVERVIEW

我们的推荐系统的总体结构如图2所示。该系统由两个神经网络组成：一个用于候选代，一个用于排名。



候选生成网络从用户的YouTube活动历史记录作为输入，并从大型语料库中检索一小部分（数百）视频。这些候选对象通常与用户相关，精度很高。候选生成网络仅通过协同过滤提供广泛的个性化。用户之间的相似性以粗略的特征表示，如视频监视的ID、搜索查询令牌和人口统计信息。

在列表中呈现一些“最佳”建议需要一种精细的表示法，用于区分具有高回忆能力的候选人之间的相对重要性。排名网络通过使用一组描述视频和用户的丰富功能，根据所需的目标函数为每个视频分配一个分数来完成此任务。得分最高的视频将按得分排列，并呈现给用户。

两阶段的推荐方法允许我们从大量视频（数百万）中提出建议，同时确保设备上出现的少量视频是个性化的，并吸引用户。此外，这种设计还可以混合其他来源生成的候选项，例如在早期的工作中定义的那些候选项[3]。

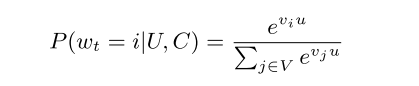
在开发过程中，我们广泛使用离线度量（精度、召回、排名损失等）来指导系统的迭代改进。然而，为了最终确定算法或模型的有效性，我们依赖于通过现场实验进行的A/B测试。在一个实况实验中，我们可以测量点击率、观看时间和许多其他测量用户测量的指标的细微变化。这一点很重要，因为实时A/B结果并不总是与离线实验相关。

3.CANDIDATE GENERATION

在候选生成期间，庞大的YouTube语料库被筛选成数百个与用户相关的视频。这里描述的推荐人用户的前身是一种在等级损失下训练的矩阵分解方法[23]。我们的神经网络模型的早期迭代用只嵌入用户以前观看记录的浅层神经网络模拟了这种分解行为。从这个角度来看，我们的方法可以看作是因子分解技术的非线性泛化。

3.1Recommendation as Classificatio

我们提出了一种极端的多类分类建议，即预测问题能够根据用户u和上下文c，在时间t的数百万个视频i（类）中准确地对特定的视频观看wt进行分类。



其中u∈RN表示用户的高维“嵌入”，上下文对和vj∈RN表示每个候选视频的嵌入。 在此设置中，嵌入仅仅是稀疏实体（单个视频，用户等）到RN中的密集向量的映射。 深度神经网络的任务是学习用户嵌入u作为用户历史和上下文的函数，这对于使用softmax分类器区分视频是有用的。

虽然YouTube上存在明确的反馈机制（拇指向上/向下，产品内调查等）我们使用手表的隐式反馈[16]来训练模型，用户完成视频是一个积极的例子。 这个选择基于更多隐含的用户历史记录的数量级，允许我们在显式反馈非常稀疏的情况下深入推出。

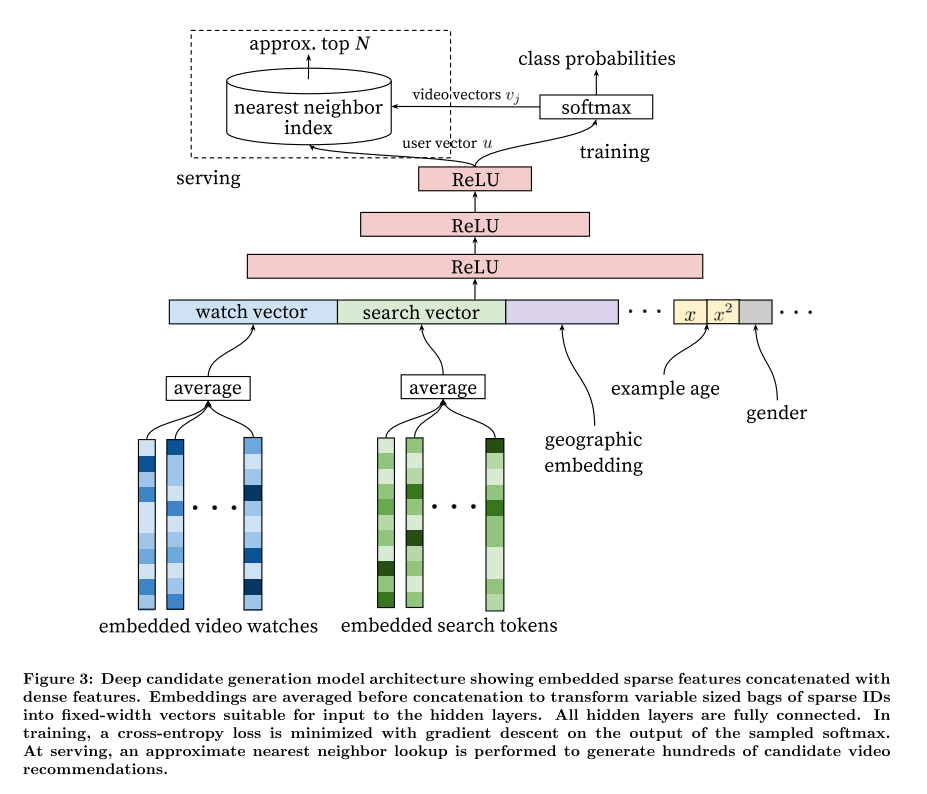
Efficient Extreme Multiclass高效的极端多类

为了有效地训练具有数百万个类的这种模型，我们依靠一种技术从背景分布中采样负类（“候选采样”），然后通过重要性加权来校正这种采样[10]。 对于每个示例，对于真实标签和采样的负类，交叉熵损失最小化。 在实践中，对数千个负片进行采样，相当于传统softmax的100倍以上。 一种流行的替代方法是分层softmax [15]，但我们无法达到相当的准确度。 在分层softmax中，遍历树中的每个节点涉及区分通常不相关的类集，使得分类问题更加困难并且降低性能。

在服务时，我们需要计算最可能的N.类（视频）以选择要呈现给用户的前N个。 在数十毫秒的严格服务延迟下对数百万项进行评分需要在类的数量上具有次要线性的近似评分方案。 YouTube以前的系统依赖于hashing[24]，这里描述的分类器使用类似的方法。 由于在服务时不需要来自softmax输出层的校准似然，因此评分问题减少到可以使用通用库的点积空间中的最近邻搜索[12]。 我们发现A / B结果对最近邻搜索算法的选择不是特别敏感。

3.2 Model Architecture

受连续词汇语言模型[14]的启发，我们在固定词汇表中为每个视频学习高维嵌入，并将这些嵌入提供给前馈神经网络。用户的观察历史由可变长度的稀疏视频ID序列表示，其通过嵌入映射到密集矢量表示。网络需要固定大小的密集输入，并且简单地平均嵌入在几种策略（总和，分量最大等）中表现最佳。重要的是，嵌入通过正常梯度下降反向传播更新与所有其他模型参数一起学习。功能连接成一个宽的第一层，然后是几层完全连接的Rectified线性单元（ReLU）[6]。图3显示了具有下面描述的其他非视频监视功能的一般网络架构。



3.3Heterogeneous Signals异构信号

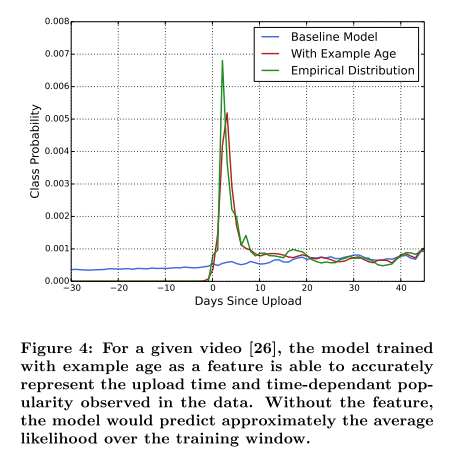
使用深度神经网络作为矩阵分解的推广的一个关键优势是可以轻松地将任意连续和分类特征添加到模型中。搜索历史记录的处理方式与监视历史记录类似，每个查询都被标记为unigrams和bigrams，并且每个标记都是嵌入的。平均后，用户的标记化嵌入式查询表示汇总的密集搜索历史记录。人口统计功能对于提供先行设备非常重要，因此建议可以为新用户提供合理的建议。用户的地理区域和设备是嵌入和连接的。简单的二进制和连续特征（例如用户的性别，登录状态和年龄）作为标准化为[0,1]的实际值直接输入到网络中。

“Example Age” Feature

每秒都会将数小时的视频上传到YouTube。推荐最近上传的（“新鲜”）内容对于YouTube作为产品非常重要。我们始终注意到用户更喜欢新鲜内容，但不以牺牲相关性为代价。除了简单推荐用户想要观看的新视频的第一顺序效果外，还有一个关键的次要现象是引导和传播病毒内容[11]。

机器学习系统通常表现出对过去的隐含偏见，因为它们经过训练以从历史实例中预测未来行为。视频流行度的分布是非常不稳定的，但是我们的推荐者产生的语料库上的多项分布将反映几周培训窗口中的平均观察可能性。为了纠正这个问题，我们在训练期间将训练样例的年龄作为特征。在服务时，此功能设置为零（或略微为负），以反映模型在培训窗口的最后进行预测。

图4展示了这种方法对任意选择的视频的有效性[26]。



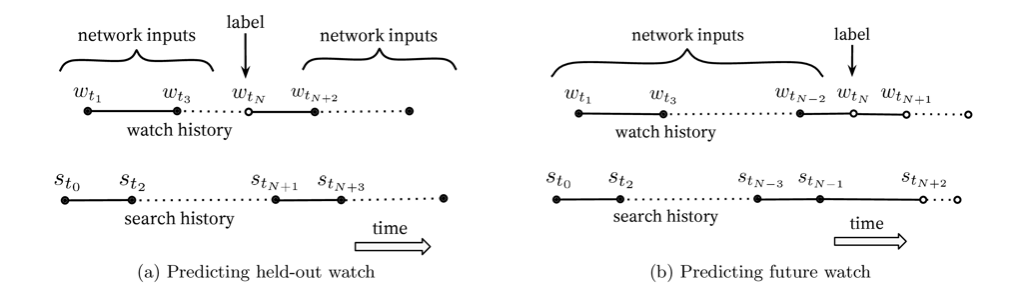
3.4 Label and Context Selection

重要的是要强调该建议通常涉及解决代理问题并将结果转移到特定背景。一个典型的例子就是假设准确预测评级会导致有效的电影推荐[2]。我们发现，这种替代学习问题的选择对于A / B测试的性能具有极大的重要性，但是非常难以用实验进行测量。

所有YouTube手表（甚至是嵌入在其他网站上的手表）都会生成培训示例，而不仅仅是观察我们制作的建议。否则，新内容浮出水面会非常困难，推荐者会过度偏向剥削。如果用户通过我们的推荐以外的方式发现视频，我们希望能够通过协作过滤将此发现快速传播给其他人。改进实时指标的另一个关键见解是为每个用户生成固定数量的培训示例，在损失函数中同等地对我们的用户进行加权。这阻止了一小群高度活跃的用户主导损失。

有点违反直觉，必须非常小心地从分类器中隐瞒信息，以防止模型利用站点结构并过度设置替代问题。以用户刚刚为“泰勒斯威夫特”发布搜索查询的情况为例。由于我们的问题是预测下一个观看的视频，因此给定此信息的分类器将预测最有可能观看的视频是出现在“泰勒斯威夫特”的相应搜索结果页面上的视频。毫无疑问，再现用户的最后一个搜索页面作为主页推荐表现非常糟糕。通过丢弃序列信息并用无序的代币包表示搜索查询，分类器不再直接知道标签的来源。

视频的自然消费模式通常会导致非 常不对称的共同观看概率。剧情系列通常是按顺序观看的，用户经常会发现一种类型的艺术家，从最广泛的流行开始，然后再关注较小的利基。因此，我们发现预测用户下一个手表的性能要好得多，而不是预测随机保持的手表（图5）。许多协作过滤系统通过提取随机项并从用户历史中的其他项预测它来隐式地选择标签和上下文（5a）。这会泄漏未来的信息并忽略任何不对称的消费模式。相反，我们通过选择随机监视“仅回滚”用户的历史记录，并且仅输出用户在保持标签监视（5b）之前所采取的动作。



3.5 Experiments with Features and Depth

添加特征和深度可显着提高保持数据的精度，如图6所示。在这些实验中，包含1M视频和1M搜索令牌的词汇，其中包含256个浮点数，最大包含50个近期手表和50个近期搜索。softmax层在相同的1M视频类上输出多项分布，其尺寸为256（可以将其视为单独的输出视频嵌入）。这些模型经过培训，直到与所有YouTube用户融合，对应于数据的几个时代。网络结构遵循一个共同的“塔”模式，其中网络的底部最宽，每个连续的隐藏层将单元的数量减半（类似于图3）。深度零网络实际上是一种线性分解方案，其执行方式与前一系统非常相似。增加宽度和深度，直到增量收益减少并且收敛变得不同：

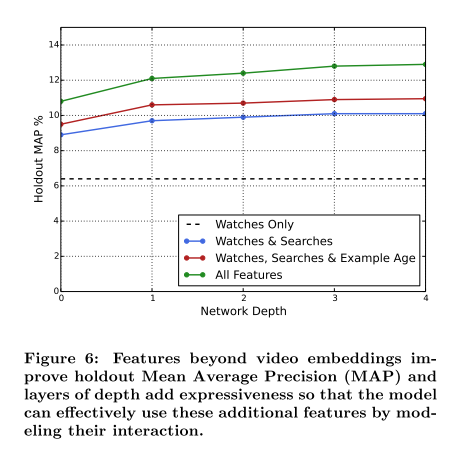
•深度0：线性层简单地转换连接层以匹配softmax维度256

•深度1：256 ReLU

•深度2：512 ReLU→256 ReLU

•深度3：1024 ReLU→512 ReLU→256 ReLU

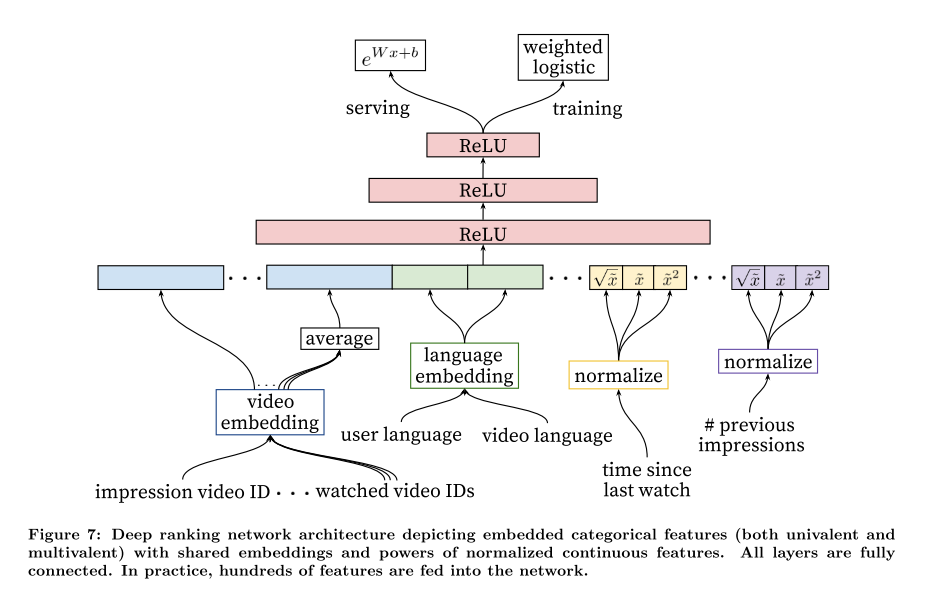
•深度4：2048 ReLU→1024 ReLU→512 ReLU→256 ReLU



4. RANKING

排名的主要作用是使用印象数据来专门化和校准特定用户界面的候选预测。例如，用户可能一般以高概率观看给定视频，但由于缩略图图像的选择，不太可能点击特定主页印象。在排名期间，我们可以访问更多描述视频的功能以及用户与视频的关系，因为只有几百个视频得分而不是候选人一代的数百万。排名对于集合不具有直接可比性的不同候选来源也是至关重要的。

我们使用具有类似架构的深度神经网络作为候选生成，使用逻辑回归为每个视频印象分配独立分数（图7）。然后按该分数对视频列表进行排序并返回给用户。我们的最终排名目标是根据实时A / B测试结果不断调整，但通常是每次展示的预期观看时间的简单函数。按点击率排名通常会促使用户无法完成的欺骗性视频（“clickbait”），而观看时间更能捕捉参与度[13,25]。



4.1 Feature Representation

我们的特征与传统的分类和连续/序数特征分类相隔离。我们使用的分类特征在其基数上变化很大 - 一些是二进制的（例如，用户是否登录）而其他人具有数百万个可能的值（例如用户的最后一个搜索查询）。根据功能是仅贡献单个值（“单价”）还是一组值（“多价”）进一步划分功能。单价分类特征的示例是正被评分的印象的视频ID，而相应的多价特征可以是用户已观看的最后N个视频ID的包。我们还根据功能是描述项目的属性（“印象”）还是用户/上下文的属性（“查询”）来对功能进行分类。每个请求计算一次查询功能，同时为每个评分项目计算展示功能。

Feature Engineering 特色工程

我们通常在排名模型中使用数百个功能，大致在分类和连续之间平均分配。尽管有望深入学习手工减轻工程特征的负担，但我们的原始数据的本质并不容易直接输入到前馈神经网络中。我们仍然花费大量的工程资源将用户和视频数据转换为有用的功能。主要挑战在于表示用户操作的时间顺序以及这些操作如何与正在评分的视频印象相关。

我们观察到最重要的信号是描述用户之前与项目本身和其他类似项目的交互的信号，与其他人在排名广告中的经验相匹配[7]。例如，考虑用户过去的历史记录以及上传正在评分的视频的频道 - 用户在此频道观看的视频数量是多少？用户最后一次观看有关此主题的视频是什么时候？描述过去用户对相关项目的操作的这些连续功能特别强大，因为它们可以在不同的项目中很好地概括。我们还发现将候选代的信息以特征的形式传播到排名中是至关重要的，例如，哪些消息来源提名这个视频候选人？他们分配了什么分数？描述过去视频展示次数的功能对于在推荐中引入“流失”也很重要（连续请求不会返回相同的列表）。如果用户最近推荐了一个视频，但没有观看，那么该模型会在下一页加载时自然降低这种印象。提供最新的印象和观看历史是本文范围之外的工程专长，但对于制作响应性建议至关重要。

Embedding Categorical Features 嵌入分类功能

与候选生成类似，我们使用嵌入将稀疏分类特征映射到适合神经网络的密集表示。每个唯一的ID空间（“词汇表”）具有单独的学习嵌入，其具有的维度大致与唯一值的数量的对数成比例地增加。这些词汇表是通过在训练之前传递数据一次构建的简单查找表。非常大的基数ID空间（例如视频ID或搜索查询术语）通过在排序后仅基于其在点击的展示中的频率仅包括前N来截断。词典外值简单地映射到零嵌入。与候选生成一样，多价分类特征嵌入在被馈送到网络之前被平均。

重要的是，同一ID空间中的分类特征也共享底层的emeddings。例如，存在许多不同特征使用的视频ID的单个全局嵌入（印象的视频ID，用户观看的最后视频ID，“播种”推荐的视频ID等）。尽管共享嵌入，但每个功能都单独馈送到网络中，以便上面的层可以学习每个功能的专用表示。共享嵌入对于改进泛化，加速培训和减少内存需求非常重要。绝大多数模型参数都在这些高基数嵌入空间中 - 例如，嵌入在32维空间中的一百万个ID具有比完全连接的层2048个单元宽度多7倍的参数。

Normalizing Continuous Features 规范连续特征

众所周知，神经网络对其输入的缩放和分布非常敏感[9]，而诸如决策树集合之类的替代方法对于单个特征的缩放是不变的。我们发现连续特征的正确归一化对于收敛是至关重要的。通过缩放值将具有分布f的连续特征x变换为〜x，使得使用累积分布~x = http://static.tongtianta.site/paper_image/c40b6f10-334c-11e9-8883-00163e08bb86/INNERLATEXT610.jpgdf将特征均等地分布在[0,1]中。在训练开始之前，在对数据的单次通过中计算的特征值的分位数上对该积分近似进行线性插值。

除了原始归一化特征〜x之外，我们还输入功率〜x2和√~x，通过允许网络容易地形成特征的超线性和子线性函数，赋予网络更强的表达能力。发现连续特征的馈电能力提高了精度。

4.2 Modeling Expected Watch Time

我们的目标是预测预期的观看时间，因为培训示例要么是正面的（点击视频印象），要么是负面的（没有点击印象）。正面示例用用户观看视频所花费的时间量来注释。为了预测预期的观察时间，我们使用加权逻辑回归技术，该技术是为此目的而开发的。

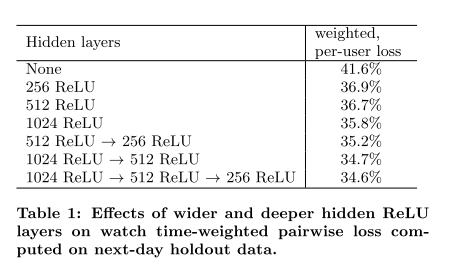
在交叉熵损失下用logistic回归训练该模型（图7）。但是，正片（点击）展示次数会按观察到的观看时间加权。负（未点击）印象都会获得单位重量。这样，逻辑回归学到的几率是N是训练样本的数量，http://static.tongtianta.site/paper_image/c40b6f10-334c-11e9-8883-00163e08bb86/INNERLATEXT612.jpg k是正印象数，而http://static.tongtianta.site/paper_image/c40b6f10-334c-11e9-8883-00163e08bb86/INNERLATEXT613.jpg是第i印象的观察时间。假设正印象的分数很小（在我们的例子中是正确的），学习的几率大约是E [T]（1 + P），其中P是点击概率，E [T]是预期的观察时间。印象。由于P很小，该产品接近E [T]。对于推理，我们使用指数函数http://static.tongtianta.site/paper_image/c40b6f10-334c-11e9-8883-00163e08bb86/INNERLATEXT614.jpg作为最终激活函数来产生这些几乎估计预期观看时间的赔率。

4.3 Experiments with Hidden Layers

表1显示了我们在具有不同隐藏层配置的次日保持数据中获得的结果。通过考虑在单个页面上向用户显示的正（单击）和否定（未单击）印象来获得针对每个配置显示的值（“加权的，每用户损失”）。我们首先使用我们的模型对这两个印象进行评分。如果负面印象得分高于正面印象，那么我们认为正面印象的观看时间是错误预测的观看时间。然后，加权，每用户损失是错误预测观看时间的总量，作为总观看时间的一小部分而不是保持印象对。

这些结果表明，增加隐藏层的宽度可以改善结果，增加深度也是如此。但是，交易-ff是推理所需的服务器CPU时间。1024宽ReLU的配置，然后是512宽ReLU，然后是256宽ReLU，为我们提供了最佳结果，同时使我们能够保持在我们的服务CPU预算范围内。

对于1024→512→256型号，我们尝试仅在没有功率的情况下馈送标准化的连续特征，这增加了0.2％的损失。使用相同的隐藏层配置，我们还训练了一个模型，其中正面和负面的例子被加权相等。不出所料，这使手表时间加权损失增加了4.1％。



5. CONCLUSIONS

我们已经描述了用于推荐YouTube视频的深度神经网络架构，分为两个不同的问题：候选生成和排名。

我们的深度协作过滤模型能够有效地吸收许多信号并模拟它们与深度层的相互作用，优于YouTube上使用的先前矩阵分解方法[23]。在为推荐选择替代问题时，艺术比科学更多，我们发现通过捕捉不对称的共同观察行为并防止未来信息泄露，对未来的手表进行分类以在现场指标上表现良好。扣留来自分类器的描述性信号对于获得良好结果也是必不可少的 - 否则该模型将超过替代问题而不能很好地转移到主页。

我们证明了使用训练示例的年龄作为输入特征消除了对过去的固有偏差，并允许模型表示流行视频的时间依赖行为。这提高了精确度，并且最近在A / B测试中上传的视频显着增加了观看时间。

排名是一种更经典的机器学习问题，但我们的深度学习方法优于先前的线性和基于树的观察时间预测方法。推荐系统特别受益于描述过去用户对项目行为的特殊功能。深度神经网络需要分类和连续特征的特殊表示，我们分别用嵌入和分位数归一化进行变换。深度层被证明可以有效地模拟数百个特征之间的非线性相互作用。

Logistic回归通过加权训练样本进行修改，其中观察时间为正例，而负面例子则为统一，使我们能够学习紧密模拟预期观看时间的几率。与直接预测点击率相比，此方法在观看时间加权排名评估指标上表现更好。