

协作排名

苏赫里德·巴拉克里希南
AT&T实验室-研究180公
园大道。
新泽西州福勒姆公园

suhrid@research.att.com

苏米特·肖普拉AT&T
实验室-研究180公
园大道。
新泽西州福勒姆公园

Schopra@research.att.com

摘要

典型的推荐系统使用预测和实际评级之间的均方根误差(RMSE)作为评价指标。我们认为RMSE不是这项任务的最佳选择，特别是当我们只向任何用户推荐几个(顶部)项目时。相反，我们建议使用排序度量，即归一化折现累积增益(NDCG)，作为这项任务的更好的评估度量。借鉴学习的思想，对网络搜索社区进行排序，提出了新的模型，对推荐任务进行了近似优化。我们的MOD-ELS本质上是矩阵分解模型的变化，在这些模型中，我们还学习与用户相关的特性和排名任务的项。一些标准的协同过滤数据集的实验结果验证了我们的说法。结果还表明了我们的模型的准确性和效率，以及学习features对排名的好处。

类别及主题描述

H.2.8[数据库应用]: 数据挖掘-协作过滤和排序;
I.2.6[学习]: 连接-ISM和神经网络

一般条款

算法

关键词关键词

推荐系统, 学习排名, 协作排名, NDCG, RMSE

1. 引言和专题

在过去的十年里, 大量的基于网络的recommendation系统在不同的领域得到了巨大的成功。它们的激增可归因于各种因素。其中, 一个关键因素是相当无限的选择

允许将本作品的全部或部分用于个人或课堂使用的数字或硬拷贝免费发放, 但不得为教授或商业利益制作或分发副本, 且副本载有本通知和first页面上的完整引文。否则复制、重新发布、在服务器上发布或重新分配到列表, 需要事先的特定许可和/或费用。

WSDM'12, 2012年2月8日至12日, 美国华盛顿州西雅图。版权2012ACM978-1-4503-0747-5/12/02.\$10.00。

可供用户在多种应用程序中使用的项目。一部分非常受欢迎的基于网络的推荐服务列表是: Netflix, 电影, Amazon.com, 产品, 潘多拉, 最后。调频, 我调谐天才的音乐, 你管的推荐给你, 在线视频, Facebook的其他人你可能知道, 为社交网络, 我应该阅读下一步, 为书籍等。

在典型的推荐系统能够为用户提出个性化推荐之前, 它需要征求用户的偏好。这是通过向用户征求数字(现在也是普遍存在的)“明星”评级来完成的, 以获得选定的项目。例如, Netflix一旦用户在他们的网站上创建了一个帐户, 就把这作为第一步; 他们要求一个新用户在5点星的范围内对一些预选的电影/电视节目进行评分, 从1=讨厌它到5=喜欢它。经过足够多的评分, 协作过滤(CF)[17]技术被应用到整个数据集(所有用户, 所有项目)。CF模型是大多数推荐系统的核心。它们通过从目标用户收集的偏好信息和所有其他用户的偏好推断未观察到的用户项偏好来工作。最后, 提出了建议, 用户可以显示估计是她最喜欢的项目。这通常是通过使用CF模型来首先估计所有项目的首选项来完成的。接下来, 根据这些估计的首选项对项目进行排序; 最后, 将顶部项目的(小)子集显示给用户作为她的建议。

由于许多推荐系统都是在这种使用显式评级作为用户偏好替代的模式下运行的, 因此出现了一种自然的方法来训练和评估这种系统。在此视图中, 推荐任务减少到预测未见用户项对的评级。为了评估这些系统的性能, 从所有用户收集的评级数据被划分为一个训练portion(训练集)和一个不相交/不重叠的测试部分(测试集)。模型是在训练集上学习的, 并在测试集上进行评估(在训练过程中没有看到)。由于推荐任务的框架是预测未见用户项对的评级, 因此评估包括量化推荐者预测测试用户项评级的程度。这本质上是一个回归任务, 模型预测和实际测试评级之间的平均平方误差(MSE)是一个评估度量的natural选择。测试MSE较低的模型平均比MSE较高的模型有更好的预测。此外, 一旦测试MSE是评估

在训练这些模型的同时，训练MSE也自动成为损失函数的首选。

虽然这是建立推荐系统的一种完全有效和成功的方法，但我们和其他几位研究人员[22、6、16、21]认为，使用MSE标准来评估这类系统是不太适合推荐任务的。原因有两方面。首先，正如前面提到的，大多数推荐者在实践中使用的方式是生成一个项目的topk列表来显示

每个用户。因此，只建议系统TEM估计，被她高度评价的人将永远展示给用户。用户从来没有显示任何项目，系统认为她不会给予高评级。由于MSE对所有评级都给予了同样的重视，无论是高的还是低的，因此最小化MSE的结果是训练模型，这些模型可以像高的一样准确地预测低评级。因此，在试图解决一个比我们需要的更困难的问题时，似乎有很多额外的工作是不必要的。作为一个例子，在Netflix的情况下，由于我们事先知道我们只会向用户显示（估计）4/5星的项目，因此要求我们的系统善于准确预测1星级评级似乎浪费了建模能力。对MSE作为训练标准的第二个主要批评是，在大多数情况下，交流预测的评级值本身甚至没有直接显示给用户(Netflix是一个显著的例外)。如果预测数值的唯一用途是在一个topk列表中，这些评级的预测值的精度似乎也不是必要的。相反，你得到了

正确的项目顺序似乎至关重要。特别是，列表顶部的正确顺序似乎是关键。

由于这些原因，我们认为，不使用MSE，排名度量将更适合于评估重新修正系统。特别是，在本文中，我们表明归一化折扣累积增益(NDCG)是一个特别好的适合作为一个度量来评估这种系统。NDCG（数学细节见2.1节）是以信息检索中的排名作为tar-get应用程序[12]开发的。给定关联值（通常在序数尺度上）的一组项目（网页）作为重新搜索查询返回，NDCG可以用来评分这些项目的任何排列列表。设计NDCG的目的是使在排名最高的职位中具有最高相关性项目的列表是获得最高分数的列表。由于推荐系统将只用于向用户展示几个建议的项目，我们的重点也正是在列表顶部填充的项目上。只要低排名项目可以与高排名项目区分开来，估计低评分对NDCG和排名任务几乎没有影响。我们将此排序视图称为推荐协作排序。

在网络搜索学习到等级领域的工作的推动下，我们提出了两类协作排序问题的模型：点态模型和配对模型。这两种模型背后的思想是学习一个参数函数，它为每个输入分配一个相关分数，这是一个用户项对。这两种模型的区别在于如何学习函数的参数来优化排序度量。

在推荐设置中使用这些模型的一个关键问题，而不是网络搜索排序，是缺乏显式输入特性的可用性。为此，我们提出了一个新的解决方案，即学习这些特征，同时学习参数

在损失函数优化过程中的排序函数。我们通过许多标准的真实世界协同过滤数据集上运行我们的模型来验证我们的声明。结果表明了我们的模型的效率和准确性。我们还展示了学习输入特性的好处，这些特性被调整到手头的排名任务中。

在第二节中，我们简要讨论了学习到等级社区的状态和人们用于网络搜索排名的模型类型。第三节详细讨论了我們提出的协作排序问题的模型。然后，我们在第4节中讨论相关工作和评估。接下来，我们将在第5节中讨论我们的实验和结果。最后，我们以我们的结论结束，并在第6节中讨论未来的工作。

2. 学习到兰克

网络搜索任务涉及在互联网上查找信息。在标准设置中，用户向搜索引擎提供一组搜索术语（查询），搜索引擎返回相关超链接的列表（列表或im-预置）。然后，用户可以遵循看起来相关的超链接，或完善她的搜索等。研究工作侧重于这项任务的许多具体方面。最近研究关注的一个突出领域是自动网络搜索结果排名。特别是，由于大多数用户将注意力集中在通常放在搜索结果第一页顶部的少数印象上，一个理想的特点是，搜索引擎返回的印象列表应根据它们与搜索查询的相关性进行排序。在网络搜索文献中，这项任务涉及通过强烈考虑到它们的排名来找到返回列表的模型，被称为“学习排名”(LTR)问题。

在大多数情况下，在LTR中，排序问题被视为监督学习问题，其中假定访问标记的训练数据D。一个典型的训练数据集D由一个查询-印象对列表组成 $q = \{(q, i, y_i) : i \in [1, \dots, l]\}$ ，其中q是查询，i是印象，y_i是相关性分数。相关性分数y_i通常由一些序数尺度Y（例如，y=1，与y=5无关，非常相关）的编辑判断，查询印象对用固定长度特征向量x表示 $x_i \in \mathbb{R}^d$ 。搜索引擎使用的确切特征是专有的，但通常包括测量查询和列表主体之间文本重叠的数量

文本、锚文本、URL和文档标题。它们还可以包括特定于查询的数量，如查询的长度，或特定于列表的数量：它的Page Rank、链接特性等。学习包括估计一个模型的参数，它以特征x作为输入 q_i 与查询印象对(q, l)和pro相关联

介绍正确的相关性分数y。

2.1 NDCG

回想一下，我们的目标是产生排名机制，使相关印象在一个排名列表的顶部非常好。如前所述，众所周知，前几个印象得到更多的关注（点击和视图），而不是印象进一步向下的页面[19]。更进一步，列表的注意力下降得更快，许多用户很少超过结果的第一页[19]。反映了这些关切，这是一个评价指标

已经成为非常流行的LTR社区是正常折扣累积增益或NDCG[12]。

可以计算一组关联标签已知的项目的任何排列的NDCG分数。NDCG有一个用户定义的参数和两个功能，使它在排名设置中是可取的。参数k是一个切割

关闭参数，并确定排名中有多少项要考虑的名单。这两个函数是增益函数和折扣函数。增益函数允许用户设置每个关联级别的意义。折扣功能使项目较低的排名列表贡献较少的NDCG评分。更具体地说，让y是一个项目序列的相关值的向量(例如，与一个查询相关联的im-presions)。让 π 表示每亩

关于y中项的序列。例如， π 将由经过训练的排序算法返回的列表的顺序。然后 π_{qq} 是 π 和y中qth项的索引。 π_{qq} 是本项目的实际关联值。该置换 π 的折扣累积增益定义为：

$$DCG@k(y, \pi) = \sum_{q=1}^k \frac{2^{\pi_{qq}}}{2 + q}$$

增益函数是这个定义中的两个幂(mi-nus1)，当我们向下移动排名列表时，折扣函数具有对数衰减。然后，归一化折扣累积增益可定义为：

$$NDCG@k(y, \pi) = \frac{DCG@k(y, \pi)}{DCG@k(y, \pi^*)}, \quad (1)$$

哪里 π^* 是对应于任何基于相关性分数的完美排序的项目序列的排列；在相关性分数较低的项目中没有排序

在排序中出现的时间比任何具有较高相关性分数的项目都要早(换句话说，项目按y排序，并且领带被任意破坏)。

理想情况下，人们希望学习排名模型的参数，以便它直接最大化预测排名的NDCG。然而，度量是连续的，在任何地方都会带来重大的挑战，同时优化它。此外，与之相关的固有“排序”使得这一度量的平滑松弛/近似的设计同样困难。为此，几位作者[5, 3, 15]提出了LTR问题的一些解决方案，其中涉及一个代理损失函数，其minimization将松散地逼近NDCG度量的最大化。我们在

下一个高水平。

2.2 学习排名方法

LTR范式中的模型可以分为三大类：点状、成对和所谓的列表明智方法。不同之处主要在于损失函数的形式和使用的训练数据。

点模型估计参数函数 $h(x)_{q/}$ ，将与每个查询印象项目相关联的特征作为输入，并作为输出产生关联分数。在

为了训练该函数的参数，当预测的相关性分数为时，可以使用回归损失。

连续的 $\mathcal{O} \rightarrow R$ [5]，或分类损失时

相关性分数是离散的($h: R \rightarrow Y$)[15]。在训练时，在排名指标上的表现，如NDCG，是通过明智的选择来模仿的

目标反应。例如，在[5]中，作者使用实际相关值的指数作为回归的目标。这将相关值y重新标度为更接近NDCG度量的分子的值。因此

明智的模型本质上是Web搜索查询印象、相关性数据的回归/分类模型。请注意，在点向模型中，查询之间的区别只通过适当的查询-印象特性x来典型地表示 $q/$ 。换句话说，查询q和印象1

只定义适当的特性，这些示例及其

特征 $x_{q/}$ 随后，相关的响应被独立地调制。建模的基础是查询印象特征 $x_{q/}$ 信息丰富，足以区分它们给定的相关性水平。对于模型本身，通常使用灵活和强大的(非线性)回归/分类方法，如梯度增强回归树。这种点的优势

模型是学习模型相对简单，更重要的是，这些方法很好地扩展到大尺寸问题。

配对模型也估计函数 $g(x)_{q/}$ 每个查询印象项目得分 $g: R_{q/} \rightarrow R$ 。但是，

成对模型的训练损失是基于具有相同查询的对印象，{(q) 我是 x 我是 我知道 我是 是的 我是 (q) j x 我是 我知道 j 是的 j)} 其中q 我是 = q j 但是我 我是 我知道 j。两两损失取决于 对中项目的关联标签的顺序。损失

当预测的顺序是正确的，或者换句话说，当一对中更相关的项目被预测比不太相关的项目有更高的分数时，就会很低。在我们的例子中，用y j > 你 我是 g(x)的损失将很低 我是 我知道 j > g(x) 我是 我是 我知道 我是。

当订单出现错误时，损失将很高。

例如，在我们以前的例子中，当y j > 你 我是 但是， g(x) 我是 我知道 j < g(x) 我是 我是 我知道 我是)。成对方法的例子有排名Boost[8]、排名网和兰姆达排名[3]。直观

对于这些模型，可以很好地排序的函数也应该能够对项目进行良好的排序。

我们还简要地给出了列表方法的指针，例如AdaRank和PermuRank[23, 24]，它们超出了我们的工作范围(可能是未来工作的一个有趣的途径)。这些方法已经被设计成更优化的专业排名绩效指标，如NDCG，平均平均精度等。

接下来，我们将概述这些LTR方法(点和成对)如何应用于重新推荐问题。一些问题自然会出现。我们如何以类似于排名的方式设置推荐问题？协作过滤设置中对查询的响应概念是什么？我们可以使用什么功能？？什么选择的参数功能将表现良好？我们如何修改点态和成对损耗来优化NDCG的性能？我们接下来要处理这些问题和其他问题。

3. 合作谈判

回想一下，推荐系统的目标是在一组m用户U和n项I上给出评级数据R的情况下提出有效的建议。在我们的符号中，我们将使用索引i ∈ U来索引用户和索引j ∈ I来索引项目。因此，我们可以参考

¹ 也有类似的基于两两的分类模型，但我们将在本文中讨论回归模型。

第*i*项和第*j*项用户的个人评级 r_{ij} 。自练习一个典型的用户费率只有一小组项目，评级数据*R*通常非常稀疏。

我们首先概述潜在因素模型[14]，一个标准类非常流行和有效的协作过滤模型。在潜在因子模型中，我们学习表示每个用户和每个项目的“因子”的*d*维向量。通常，*d*比*m*或*n*小得多，它改变了我们的信念，即少量未观察到的因素足以提供准确的评级。我们先做准备

评级为*R*的小说 \hat{r}_{ij} ，以... 项目*i*和用户*j*的因素。数学上：

$$\hat{r}_{ij} = \mathbf{v}_i^T \mathbf{u}_j$$

其中，每个项*i*和用户*j*的因子是*d*维向量*v* 我是 和你 \mathbf{u}_j 。我们亦会参考使用*U*收集所有用户因素，并类似地使用*V*收集所有项目因素。

当MSE作为评价和训练准则时，这就导致了参数估计，利用对低次目标函数的修改：

$$\min_{\mathbf{v}, \mathbf{u}} \sum_{(i,j) \in R} (r_{ij} - \mathbf{v}_i^T \mathbf{u}_j)^2 + \lambda (\|\mathbf{v}_i\|^2 + \|\mathbf{u}_j\|^2) \quad (2)$$

各种潜在因素模型的差异通常表现为因素的正则化形式。在上述公式中，从概率矩阵分解(PMF, [20])因子正则化的形式是平方 l_2 关于因素的规范。我们重载了*R*的符号也表示在

稀疏矩阵($\{i, j\} \in R$)。此外，测试MSE是由

$$MSE = \frac{1}{|R|} \sum_{(i,j) \in R} (r_{ij} - \mathbf{v}_i^T \mathbf{u}_j)^2$$

测试额定值矩阵*R*的计算*。特别是*R**|表示测试用户项对的数目。

如前所述，我们认为，对于 topk recommendations，预测和实际评级的平方损失是次优的。这是因为MSE的损失是错误的

预测的评级相等。相反，我们建议使用平均 NDCG@*k* (方程1) 作为评价指标。我们从 LTRCOM-Munity 中借鉴了合适的排名损失，以学习更准确的排名任务模型。我们称这种范式为协作秩 (如[22])。遵循LTR方法 (见第2节)，我们提出了问题的点和成对解决方案。

3.1 逐点模型

我们提出的第一组方法类似于LTR社区中使用的基于回归的点态方法。关键洞察力是在[5]和[15]中提出的一个想法；完美回归 (或分类) 也将导致完美的 NDCG。分析 DCG、Cossock 和 Zhang 可以用回归误差约束 DCG 误差[5]。同样，李等。在[15]中，AI 能够通过分类误差约束 DCG 误差。实际上，这意味着回归 (或分类) 是优化等级的可行路径。这些结果是有利的，因为回归和分类都是非常好的研究和可扩展的。

回想一下，在LTR设置中，有查询、输入和相应的关联标签，以及该查询

² 请注意，错误项通常也包含在模型规范中。高斯噪声通常是使用的误差形式。

印象对产生了这些模型中的特征。然而，对于我们的推荐任务，我们只有稀疏的评级数据*R*。因此，目前尚不清楚如何直接应用LTR中使用的上述回归 (或分类) 类型方法

协作排序任务。特别是，我们需要清楚地定义查询的类似概念、查询印象特征和相关值。回想一下，LTR问题本质上涉及对响应查询的一组输入排序。类似地，可以查看推荐的问题，例如为特定用户排序一组项。因此，推荐任务中的用户对应于Web搜索中的查询。接下来，虽然我们希望使用平均测试NDCG作为评估度量，但与LTR不同，我们没有明确的用户项特征来训练模型。很方便，这正是我们通过训练一个潜在因素模型得到的结果。最后，随着查询等同于用户，并定义了特征，查询印象的相关值，直观地与我们观察到的用户项评分值相关。这导致了我们的第一个建议。这是两步法。

第一步是利用RMSE训练一个潜在因素模型，以获得用户和项目因素。我们的前任我们使用了方程2中描述的PMF模型。模型训练后，第*i*项和用户*j*的因子

项目用户对(*i*, *j*)。这些形式的“特征”的每一个评级的项目-用户对。在第二步，利用这些特征

从第一步作为输入，我们应用基于回归的点向LTR算法进行排序优化。我们把这种简单的两阶段方法称为CR Δ 最惠国 在那里， Δ 符号 BOL 表示它是点向技术，

MF代表使用矩阵分解估计的特征。

有足够的能与CR合作良好 Δ 最惠国。我们以前与各种非线性回归器进行了实验，取得了大致相同的成功。特别是，我们尝试了梯度增强树[9, 15]，随机森林[2, 18]和多层神经网络[3]。

更详细地说，CR培训程序 Δ 最惠国 模型模型 基于g的神经网络如下：

1. 给定额定值数据*R*，我们首先训练一个*d*维因子模型，如PMF，它最小化了方程2中的目标函数。这导致了一组项目因素*V*和*U*。
2. 然后，我们为基于等级的培训创建一个新的培训数据集*D*如下：对于每一次观察到的培训评级 r_{ij} 从*i*项和用户*j*中创建一个single固定长度特征向量 $\mathbf{x}_{ij} = [\mathbf{v}_i \text{ 我是 } \mathbf{u}_j]$ 的长度 $2k$ 。符号“ Δ ”表示按列排列的凹凸阿提。此外，对于每个目标等级 r_{ij} 我们也是吃了修改后的响应 y_{ij} 由 y 提供 $y_{ij} = 2^{\mathbf{r}_{ij}} - 1$ 。这种响应的重新标度，以更好地反映NDCG增益在[5]中提出，并被发现工作良好在[15]中。因此，我们有一个数据集 $D = \{(\mathbf{x}_{ij}, y_{ij}) \mid (i, j) \in R\}$ ，索引所有培训评级。
3. 使用这个标记的数据集*D*，我们然后学习一个适合的回归函数*g*，参数 θ ，将用户项特征映射到重新标度的评级值。因此*g*是一个函数，使得 $g: \mathbb{R}^{2k} \rightarrow \mathbb{R}$ 学习最小化重新标度上的平方误差

答复：

$$L_{zz} = \frac{1}{2} \sum_{(i,j) \in D} (y_{ij} - g(x_{ij}))^2 \quad (3)$$

这种最小化是使用随机Gra-dient下降来实现的..特别是对于每个项目用户对 (i, j) ，其特征向量 $x_{ij} = [v_i, u_j]$ 向前推进 通过函数 g 来计算输出和损耗。损失的梯度 对函数的参数使用反向传播计算，并更新 θ 的参数。我们使用学习速率退火和早期停止 作为一种方法来规范参数。有关模型的示意图，请参见图1。

- 在测试时，给定一个用户项对来进行预测，构造相应的用户项特征，并利用学习函数 g 获得分数。
- 一旦我们的模型预测了所有项目-用户对的分数，任何用户的电影排名列表都可以通过根据他们对该用户的分数对电影进行排序来获得。

这个模型的一个问题是它的两个步骤性质。在里面 第一阶段CR的特点 使用MSE学习 最小化，也不清楚这些是最优的 用于排序任务的特性。理想情况下，人们希望使用最适合手头最后任务的功能。使用神经网络作为排序函数为我们提供了一个有吸引力的机会来学习网络的参数，同时学习针对排序任务调谐的项用户特征表示。这导致了这个问题的统一模型，它完全是基于排序并从数据中估计的。本工作的特征学习是受自然语言处理问题模型的后发[1, 4]。

我们称之为统一型号 α_{LF} (学习因素LF)。在高层，CR α_{LF} 就像一个典型的因子模型，因为它对所有的训练都有d维因子U和V 用户和项目。另外，CR α_{LF} 有一套 θ 功能上决定如何计算的参数 预测的反应，这是规模评级。对于排序任务，同时学习 fac-tors 和模型参数 θ .. 我们使用以下类似EM的算法来训练这个模型（见图1）：

- 我们把收视率重新定为 $y_{ij} = 2^{r_{ij}} - 1$ 。
- 随机初始化因子U和V.. 人们可以潜在地以其他方式初始化因素，例如使用矩阵分解过程的输出。
- 接下来，我们修正了因子U和V，并估计了排序函数的pa-rameters θ ，它将这些特征作为输入，并产生一个重新标度的评级。这种训练是通过最小化重标度目标响应上的平方误差和回归函数的输出，使用随机梯度下降过程（类似于我们所描述的）。CR α_{LF} 。
- 经过步骤3中一个阶段的参数训练，我们修复了参数 θ 并训练了特征U和V。

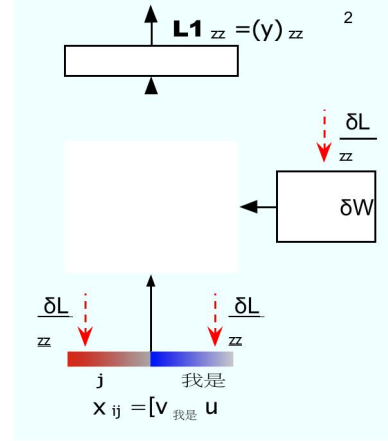
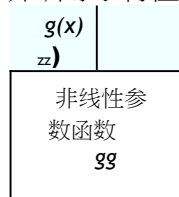


图1：CR图案 α_{LF} 及CR α_{LF} 。用户因素 v_i 项目因素 u_j 形成参数 $x_{ij} = [v_i, u_j]$ 该模型用于计算预测 $g(x_{ij})$ 。每个样本的培训 反向传播每个样本的梯度 平方损失 $L_{zz} = \frac{1}{2} \sum_{(i,j) \in D} (y_{ij} - g(x_{ij}))^2$ 只更新模型参数 W (适用于CR) α_{LF} ，此外，项目用户特性 x_{ij} (CR) α_{LF} 。

该步骤还使用随机梯度下降。我们不计算 θ 参数的梯度，而是计算损失的梯度。对特征向量 $x_{ij} = [v_i, u_j]$ [英语背诵文选 传播，然后更新特征向量。

- 我们迭代步骤3和步骤4，直到验证错误不减少，并返回学习的U、V和 θ 。

我们的实验表明，这是一种准确有效的协作排序技术。接下来，我们将转向合作排名的配对方法。

3.2 配对模型

从[11]开始，研究人员将注意力集中在排名问题的配对训练损失上。在LTR社区中，Joachims的SVMrank[13]、Burges等人的RankNet和LambdaRank[3]一直是特别重要的。配对技术是有吸引力的，因为引出明确的评级（或相关性分数）遭受一个被称为校准[10]的抽回，这可以用两个例子来说明。首先，用户通常在相同的规模上有不兼容的评级。例如，在1至5的评级范围内，对某些用户A的评级为4，可能相当于对另一个用户B的评级为5。其次，有时用户会以过于慷慨（或苛刻）的方式开始对项目进行评级，并在看到一些项目后对最初的评级感到遗憾。两两技术不是精确地预测单个项目的相关性分数，而是绕过与校准相关的问题，直接专注于正确学习一对项目之间的等级顺序。

与点向模型一样，为LTR问题引入的成对模型背后的思想可以直接应用于解决协作排序问题，只要我们能够定义适当的用户项特征。成对方法的核心是学习一种

参数函数 $g(X): \mathbb{R}^d \rightarrow \mathbb{R}$ ，由 θ 参数化，其作用于单个用户项特征向量 x_{ij} 并返回一个分数。然而，这种方法是成对的，因为 θ 的参数是使用对示例进行训练的。这导致了一个模型，可以快速地进行测试预测。

因为它在单个用户项特征向量上操作；对示例只在训练中使用，在测试时不需要它们。

我们的第一个建议是类似的成对扩展到两阶段点向模型 $CR_{\text{最惠国}}$ 。我们用符号 CR 表示这个模型 \Rightarrow 最惠国， $\langle \rangle$ 指成对培训。详细程序如下：

1. 给定评级数据 R ，我们首先训练一个 d 维因子模型，以获得项目和用户因子 V 和 U 。
2. 然后，我们创建一个数据集 D ，其特征与点向特征相同，即每个项目用户 (i, j) 我们生成一个特征向量 x_{ij} 获得的将与项目 i 和用户 j ： $x_{ij} = [v_i, u_j]$ 。目标反应 y_{ij} 对于此项，用户对设置为原始评级 $y_{ij} = R_{ij}$ （没有在这种情况下重新标度）。在这一步的最后，我们

训练数据集 $D = \{(x)_{zz} \text{ 是 } zz \text{ 与 } z = \{i, j\} \in R\}$ 。

3. 然后，我们通过设置 AP-本体成对的概率回归来学习我们的评分函数 $g(X)$ 在 D 对例子上的参数 θ [3]。让一对来自特定用户的评级，在两个不同的项目按 z 索引 1 和 z_2 。定义 $o_{zz1} = g(x)_{zz1}$ 作为模型输出（预测的评级分数），以及 $o_{zz1} - o_{zz2} = g(x)_{zz1} - g(x)_{zz2}$ 作为预测来自同一用户的一对项目的分数。让我们 $zz1, zz2$ 表示模型下 z 项的概率，排名高于项目 z_2 为这个用户。在 [3] 之后，我们使用 Logistic 函数 $\sigma(Q) = \frac{1}{1+e^{-Q}}$ 建立以模型为条件的概率模型产出。因此：

$$P_{zz1, zz2} = \frac{e^{o_{zz1} - o_{zz2}}}{1 + e^{o_{zz1} - o_{zz2}}} = \frac{e^{g(x)_{zz1} - g(x)_{zz2}}}{1 + e^{g(x)_{zz1} - g(x)_{zz2}}} \quad (4)$$

学习模型包括估计分数函数 g 的参数 θ ，从而使模型产生高概率 $P_{zz1, zz2}$ 一对样本，其中 $y_{zz1} > y_{zz2}$ 当 y 时，概率较低 $zz1 < y_{zz2}$ 。这是通过最小化交叉熵来实现的损失。尤其是让 $Y_{zz1, zz2} = I(y_{zz1} - y_{zz2})$ 做某人指示函数，接受 $\{0, 1\}$ 中的值取决于你是否 $zz1 > y_{zz2}$ 或者不是。后来损失在一对单一的例子中，给出了：

$$L_{12} = \log(1 + e^{o_{zz1} - o_{zz2} - Y_{zz1, zz2}}) \quad (5)$$

在我们的实验中，我们使用了一个多层神经网络来处理 scoring 函数 g ，并利用随机梯度下降法最小化交叉熵损失。请参见图 2 中的示意图。

模型和训练过程。

4. 对于每个用户的所有对示例重复步骤 3，直到验证集上的错误停止减少为止。整个数据集的损失是由

$$L = \sum_{zz1, zz2 \in D} L_{12} = \sum_{zz1, zz2 \in D} \log(1 + e^{o_{zz1} - o_{zz2} - Y_{zz1, zz2}})$$

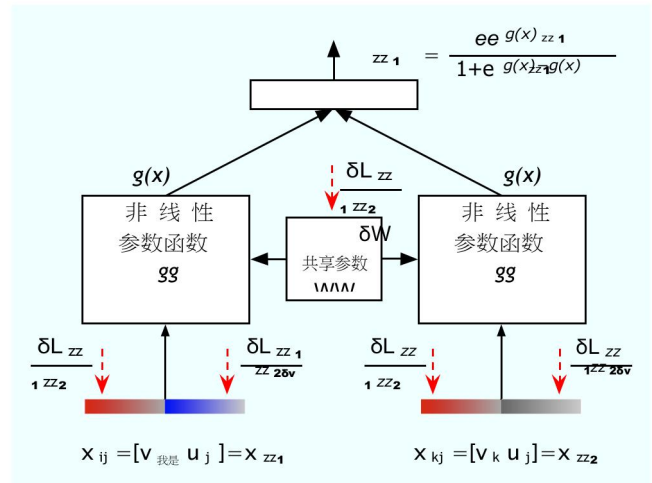


图 2：图示成对模型

$CR_{\text{最惠国}}$ 及 CR_{LF} 。在这些模型中，一对老鼠 $Ins1$ 和 k ，来自同一个用户熟练地处理我们的输入特征 x_{zz1} 和 x_{zz2} 。相同的函数 g 为两个 z 匹配一个分数 o_{zz1} 和 o_{zz2} 。每个样本损失 $L_{zz1, zz2}$ （由方程 5 给出）是这些分数的差异和 I 和 K 的真实评级。每一对样本的训练都涉及反向传播这种损失的梯度，仅限于模型参数 $W(CR_{\text{最惠国}})$ ，以及对特征 x 的补充 $zz1$ 和 $x_{zz2}(CR_{\text{LF}})$ 。

5. 在测试时，给定一个用户项对进行预词，构造相应的用户项特征，并通过输出函数 g 与学习参数 θ 。然后对每个用户的预测分数进行排序。

因为我们使用神经网络来实现函数 g ，在 sim 中

2. 我们有能力做到这一点

在成对设置中学习一个统一的模型。我们注意到这一建议，我们还从 CR_{LF} 该模型的 EM 样式算法如下类似的配方：修复因子 U, V 并估计 θ ；然后修复 θ 并估计因子 U, V （参见图 2）。

3.2.1 配对模型的后发式

仔细的读者会注意到，配对程序的复杂性似乎相当大；关于配对评级的培训是繁重的，特别是当大多数 $collab$ 时。

指示过滤数据集有数百万观察到的评级。

我们确实指出，我们只从评级中构建对于我们的稀疏评级矩阵，这种对的数量远远小于 $O(I \times R^2)$ 。然而，即使是每个用户的所有评级对都会被忽略

延长，因为许多数据集包含许多活动/沉重的用户，他们每个都有数千个评级。

与成对模型相关的另一个问题，在 Lambda Rank 文件 [3] 中提出，与在明智的模型中使用 MSE 评估度量的问题相似。特别是，当训练时，给每个用户的所有对提供相等的 em_phase 可能是次优的，当 NDCG 作为评价指标。这是因为

(6)

³ 请注意，Lambda Rank 是一个网络搜索 LTR 提案

NDCG更关心更高的评级值，包括增益函数和折扣函数（见方程1）。换句话说，试图让 g 学习一对评级为“1”和“2”的项目之间的排序并不是很有用。在羔羊里

达兰克，作者通过修改梯度更新步骤，同时循环所有（每个用户）对来解决这个缺点。特别是，它们为每个用户创建所有可能的对，但为每对分配权重，以便涉及较高级别的对，如（‘5’、‘4’）被分配比涉及较低级别的对更高的权重，如（‘2’、‘1’）。在参数的梯度更新过程中，这会导致更大的梯度步骤（从而导致 g 的更大变化）。

对涉及更高的评级和较小的步骤对

涉及较低的评级。换句话说，评级较高的对对 g 有更大的影响。他们还表明，EM的盗版，这一过程导致局部最优NDCG[7]。相反，我们把这两个问题都解决了，

大量的对和处理非信息对的NDCG，用一个简单的启发式。对于每个用户，我们不创建所有对的项目，而是创建至少有一个顶级评级类的对，用户已经评级了任何项目。例如，如果特定用户使用了值‘5’、‘4’、‘3’、‘2’来对各种项目进行评级，那么对于这个用户，我们只形成一对，其中至少有一个项目被评为a‘5’或‘4’。程序如下：

1. 对于每个用户，我们收集所有的项目，她已经评级，并根据他们的评级排序。我们让我们用 S 表示这组项 j 对于用户 j 。
2. 我们从 S 中挑选最高等级的产品 j 把它配对与设置中的所有项目，这些项目的评级低于所选项目的评级。
3. 然后，我们将此项从集合 S 中删除 j 。
4. 我们重复步骤2和3，直到我们已经耗尽了对于应于该用户的前2个评级类的所有项目。

这背后的直觉是，当我们评估NDCG时，包含每个用户前两个评级类的对（比如‘4’和‘5’）是我们需要关注的。正如[5]和[15]中所指出的，为了具有良好的测试NDCG性能，所学函数必然不需要具有完美的回归（或分类）性能。例如，在每个评级分数中添加相同的常数不会影响NDCG的性能。事实上，重要的只是两件事：(A)正确区分高评级项目的的能力，例如，在一个项目评级‘4’和一个项目评级‘5’之间。准确区分高评级项目和低评级项目的的能力，例如一个项目被评为“4”和一个项目被评为“1”。我们的启发式捕获了这两种需求。A. 通过制作带有表单评级的项目对（‘5’，‘4’）来解决。B. 只由至少有一个高评级的项目的气质对编码， I 。e. 对形式（‘5’，‘1’）或（‘4’，‘2’）。由于能够订购一个（‘1’，‘2’）对可能根本不用于测试NDCG，我们不创建任何这样的对与我们的启发式。

而不是关于建议。我们已经自由了将有关的想法转化为我们的背景，以获得清晰和连续性。

4. 有关工作和评价

我们现在谈谈文献中的相关建议。事实上，没有多少其他研究人员从排名的角度关注矩阵分解。正如我们在引言中提到的，事实标准是使用MSE/RMSE进行ma-Trix分解。但是，在我们讨论一些相关的建议之前，我们在这里指出，评估排名任务的推荐者是复杂的，因为我们有固定数量的评级，用于培训模型和评估模型（例如，相对于评估部署在现场用户上的系统）。我们下一步讨论这个问题。

4.1 评价

特别是，具有单一固定评级数据集的培训和实验评估在以下两者之间存在相当直接的权衡：培训评级增加，从而减少模型预测中的偏差，测试评级增加，以及减少评估中的偏差⁴。我们的选择度量，NDCG，对这种权衡来说特别糟糕。

例如，考虑Netflix数据集的标准培训验证（探针）拆分，其中验证数据集由每个用户大约三个评级组成。在这种情况下，在培训数据集上进行培训后，我们甚至无法在验证数据集上评估NDCG@10。作为一个更相关的例子，使用一些固定数量的随机抽样评级进行验证/测试，例如每个用户20将导致NDCG@10的高方差估计。

由于我们正在为协作排序做准备，我们更喜欢更准确/更低的方差评估，因此我们首选的列车测试拆分比培训评分更多。事实上，这种形式的实验设置（大测试数据集，较小的训练数据集）也是Cofi Rank[22]所青睐的。Cremonesi等人。[6]追求不同形式的评价。在接下来的内容中，我们简要地概述了这两个建议，并强调了与我们的方法相比的相似之处和差异。

4.2 Cremonesi等人。

在他们最近的论文中，Cremonesi等人。指出MSE/RMSE作为评价指标的存在性[6]。和我们一样，他们认为基于topk的推荐方案更适合推荐人的方式

最终会被使用。在他们的论文中，他们表明，训练的方法在MSE/RMSE上表现良好，在topk推荐任务上表现不佳。他们提出了简单的稀疏SVD(因为SVD需要)

全秩矩阵，未见评级的归零值)作为topk推荐任务的竞争方法。然而，他们的工作和我们的工作有很大的不同。最大的区别是他们如何评估他们的topk推荐者。Cremonesi等人。不要考虑NDCG；他们评估使用精度/召回只是少数高评级，也有大量的项目SAM-PED随机。4.他们决定接受这项计划，以便进行培训

在一套更大的评级上。虽然这是合理的，但我们更喜欢使用NDCG进行评估，这避免了随机抽样，而且几乎没有偏见。另一个问题，克里-莫内西等人。提到流行的东西似乎很多

⁴LTR方法不会受到这个问题的影响，因为它们只是在测试/独立查询上进行评估

影响算法的性能对其精度/召回评估指标的影响。同样，我们认为这主要是由于它们的评估度量的性质；在我们不报告的实验中，稀疏SVD和top-pop（一个非个性化的推荐器，它只是返回按发生次数排序的项目）与我们评估的方法相比，都是非竞争性的。

4.3 Cof级

一个更接近我们工作的建议是Cofi Rank[22]。科菲秩的动机和发展与我们的模型完全相同。他们认为NDCG是一个打赌的训练/评估指标的建议，我们遵循的实验协议基本上是相同的（以允许比较）。在Cofi Rank中，作者拟合了一个最大裕度矩阵因式分解模型（矩阵因式分解，在fac-tors上有一个跟踪范数正则化），以最小化包括NDCG@k在内的各种损失。我们的建议有几个非常重要的区别。也许最大的区别在于我们的两两和点两两训练损失，这与Cofi Rank直接最小化训练NDCG@k的建议不同。同样，在这种情况下很难进行评估，目前还不清楚最小化训练NDCG@k是否会给出一个在测试NDCG@k上形成良好的模型，特别是当训练和测试数据大小不同时。这可能是为什么在Cofi Rank文件中，作者报告说，具有regression（平方）损失(Cofi-reg)的Cofi在使用NDCG@k(Cofi-NDCG)训练的Cofi模型中表现出竞争力/更好。在我们的实验中，我们也一致发现Cofi-reg优于Cofi-NDCG。科菲秩和我们之间的其他微小差异与矩阵因式分解模型中使用的形式正则化有关。当Cofi使用MMMF时，我们使用PMF[20]，它具有稍微不同的正则化形式。

5. 专家

我们现在描述我们为评估我们的方法而进行的实验。我们评估了我们对几个著名的协同过滤数据集的建议。这些数据集主要用于推荐电影(和DVD上的电视节目)，由一组用户对一组电影的评级集合组成。特别是，我们使用了电影镜头数据和每个电影数据。这些数据集的大小和组成各不相同。表1给出了与每一个相关的一些基本统计数据。

我们的实验评估与Cofi Rank论文中的实验评估密切相关。对于每个用户，我们随机抽样一个固定数量的N的评级，并将它们放置在培训集中。该用户的剩余评级被放置在测试集中。我们用N=10进行实验，

每个用户20次，50次培训评级。因为我们评估NDCG@10，我们要求用户至少评级为20，分别为30项和60项。不满足这些要求的用户将从培训和测试数据集中删除。此外，我们要求，要考虑一个项目，它应该已经被至少5个用户在数据集中（培训+测试）。不符合这一标准的项目也被删除。这种过滤导致数据集中的用户和项目数量略有减少。用户和项目的数量在表1的前两列中报告。对于不同值N对应的数据集，fl后的平均用户数和项目数

表1的最后两列报告了Tering。对于这两个数据集，对于不同的N值，我们生成10个复制数据集，每个数据集的训练和测试集具有不同的随机样本。我们报告平均NDCG@10超过这10个复制为每个N。其他截断点的NDCG给了我们不同的数值，但定性的结果完全相同。

5.1 结果

在我们所有的模型中，参数评分函数g是一个两层完全连接的神经网络。第一层（隐藏层）是一个标准感知器层，由一个线性模块组成，然后是一个tanh非共线性AP-plied分量，直到线性模块的输出。第二层（输出层）为全连通线性层，采用隐层输出作为输入，产生分数.. 将所有实验的隐藏层中的单元数固定，并设置为400。由于它是一个评分函数，输出层是单个单元的。与模型相关的其他超参数，如学习速率和用于早期停止的训练历元的num-ber，每个数据集使用具有相似特性的单独训练/验证数据集选择一次。我们没有做进一步的参数优化。所显示的结果是用这些固定参数学习的模型。

作为基线模型，我们选择了标准概率矩阵因式分解(PMF)，它是通过最小化回归MSE（方程2）来拟合的。我们还将我们的模型与Cofi Rank[22]进行了比较。我们在所有的实验中都设置了因子维数d=50。我们进行了培训和评估。点模型CR_{最惠国}及CR_{FL}。我们也训练过并成对模型CR进行了评价。最惠国及CR_{FL}。CR的特点。最惠国及CR_{最惠国}从基线PMF模型。

我们关于电影镜头的结果，以及每个电影数据集，见表2和表3。总的来说，结果显示了一些直观的模式。所有方法都能更好地提出排名建议，增加每个用户可用的培训评级N。此外，我们看到基线结果是相当合理的。换句话说，就是根据我们的期望，尽量减少培训MSE并不是一个可怕的代理培训提供排名推荐。由于最小MSE的培训是事实标准，我们的结果为使用这种方法提供了保证。在我们的评估中，Cofi-Rank做得不好。我们训练了[22]中讨论的所有三个版本的CofiRank，即Cofi-NDCG、Cofi-ord（序数损失）和Cofi-reg。在我们的结果中，我们重新移植了性能最好的三种模型。我们使用未调谐的超参数设置，如[22]中所述.. 在我们的实验中，我们发现许多情况下，Cofi-reg是最好的方法..

然而，我们在本文中提出的模型是明确的赢家。它们提供了优越的排名推荐性能。我们在NDCG中显示出了很大的收益，即使我们使用了通过最小化MSE而获得的固定因子，仅仅使用了一个不同但简单的预测

模型，例如CR_{最惠国}。这证明了我们的假设 MSE最小化不是最优的重新推荐。此外，根据手头的任务调整因素是至关重要的，并可能导致更多的改进。这一点从我们所学知识的表现中可以明显看出

特征模型，CR_{FL}及CR_{FL}。我们的实验不是

为两两或点明智模型的优越性提供证据。因此，作为一个实际的建议，我们建议使用点态模型，而不是它们的成对模型，因为它们的效率和可伸缩性。

6. 结论和今后的工作

在本文中，我们主张合作排名作为一种打赌的方式提出建议。受LTR社区最近工作的启发，我们提出了协作排序问题的新模型，以及它们伴随的有效算法。我们的实验表明，我们的建议是准确的，学习特征的排名任务产生最好的表现模型。在未来的工作中，我们希望将我们的模型扩展到更丰富的模型，如列表明智的LTR方法。未来工作的其他途径包括分析和形式化我们的成对启发式，并提高成对算法的可扩展性。最后，我们计划将这些结果扩展到更大的数据集，如Netflix，以及来自不同领域的数据集，如音乐推荐和图书推荐。

7. 附注

我们要感谢与鲍勃·贝尔的讨论，这总是伟大的想法和灵感的源泉。

8. 参考资料

- [1] Bengio Y., R.Ducharme, P. 文森特和詹文。一种神经概率语言模型.. *机器学习研究杂志*, 3: 1137-1155, 2003年。
- [2] 布莱曼L.。随机森林。在机器学习中, 第45 (1) 卷, 2001年。
- [3] C.洞穴。从ranknet到lambdarank到lambdamart: 概述。在微软研究技术报告 MSR-TR-2010-82, 2010年。
- [4] 科洛伯特R.和J. 威斯顿。一种统一的自然语言处理体系结构: 具有多任务学习的深层神经网络。在ICML '08: 第25次机器学习国际会议记录, 第160-167页, 纽约, 美国, 2008年。ACM.
- [5] 科索克D.和张先生。贝叶斯最优子集排序的统计分析。 *IEEE信息理论交易*, 54 (11): 5140-5154, 2008年。
- [6] P. 克莱蒙西, 韩永和R. 吐鲁林。推荐算法在top-n推荐任务中的性能。在第四次ACM推荐系统会议记录中, RecSys '10, 第39-46页, 纽约, 美国, 2010年。ACM.
- [7] P. 唐麦斯, 斯沃雷和布格斯。关于lambdarank的局部最优性.. 在第32届国际ACMS IGIR信息检索研究与开发会议记录中, SIGIR '09, 第460-467页, 纽约, 美国, 2009年。ACM.
- [8] Y. 弗伦德, 伊耶, 沙皮尔和Y. 歌手。一种有效的增强算法, 用于组合偏好。 *J. 马赫。学习。决议*, 4: 933 - 969, 2003.
- [9] J.H. 弗里德曼。贪婪函数逼近: 梯度升压机。 *统计年鉴*, 29: 1189-1232, 2000年。
- [10] 哈克和冯安S.。Matchin: 用在线游戏激发用户偏好。在CHI '09: 第27次计算机系统的人类因素国际会议记录, 第1207-1216页, 纽约, 美国, 2009年。ACM.
- [11] R.赫布里奇, 格雷贝尔和奥贝梅尔。序数回归的大裕度秩边界。在监狱里。J. 巴特利特, B. Scho伊科普夫, D. 舒尔曼和A. J. 斯莫拉, 编辑, 大保证金分类器的进展, 第115-132页。麻省理工学院出版社, 2000年。
- [12] K. (a) 拉维林和J. 凯卡。《消除》伊宁。累积增益 红外技术的评价。ACM Trans. 信息。 赛斯特。, 2002年10月20日: 422-446。
- [13] 乔希姆斯T.。利用点击数据优化搜索引擎。在第八届ACMS IG KDD知识发现和数据挖掘国际会议记录中, KDD '02, 第133-142页, 纽约, 美国, 2002年。ACM.
- [14] Y.Korn, R.Bell和C.. 伏林斯基。推荐系统矩阵分解技术。计算机, 42 (8): 30-37, 2009年。
- [15] P. 李, C.J.C.Burges, 和Q.. 吴。麦克兰克: 学习使用多重分类和梯度提升排序, 2007年。
- [16] B.马林和泽梅尔。与非随机缺失数据协同预测和排序.. 第三次ACM推荐系统会议记录 (RecSys '09), 2009年。
- [17] P. 梅尔维尔和辛德瓦尼。推荐系统。2010.
- [18] 莫汉A.、陈中强和K.Q. 温伯格。具有初始化梯度的Web搜索排序增强了回归树。机器学习研究杂志, 车间和会议记录, 14: 77-89, 2011年。
- [19] 理查森M., 多米尼克, 拉格诺。预测点击量: 估计新广告的点击率.. 在万维网第16次国际会议记录中, WWW '07, 第521-530页, 纽约, 美国, 2007年。ACM.
- [20] Salakhutdinov R.和A.Mnih。概率矩阵因式分解.. 神经信息处理系统(NIPS), 2007年。
- [21] H.史泰克和泽梅尔。用于训练top-k推荐系统的广义概率框架及其变体。 *推荐系统、算法和技术的实际使用讲习班(PRSAT, 2010年)*, 与 RecSys 合作, 2010年。
- [22] M. 魏默, A. 卡拉佐格卢, Q. 五. 乐, 和A.. 斯摩拉。协作排序: 协作排序的最大裕度矩阵因式分解。神经信息处理系统(NIPS), 2007年。
- [23] 徐J.和李宏伟。Adarank: 一种用于信息检索的增强算法。在第30届年度国际ACMS IGIR信息检索研究与开发会议记录中, SIGIR '07, 第391-398页, 纽约, 美国, 2007年。ACM.
- [24] 徐J., 泰。刘先生, 陆先生, 李先生, 和W-Y. 妈妈。直接优化学习中的评价措施进行排名.. 在第三十一届年度国际ACMS IGIR信息检索研究与开发会议记录中, SIGIR '08, 第107-114页, 纽约, 美国, 2008年。ACM.

表1：显示与我们在实验中使用的三个数据集相关的统计数据的表格。专栏“Avg”。用户（10/20/50）“（和“Avg”。项目（10/20/50）“）报告在随机抽样每个用户10、20和50评级后生成的三个数据集中的平均用户数（和项目），作为培训数据的一部分。

数据集	没有。用户	没有。项目	没有。评级	评级比额表	贪婪。用户 (10/20/50)	贪婪。项目 (10/20/50)
电影镜头 每个电影	943 61265	1682 1623	100000 2811718	1 - 5 0 - 1	941/743/496 36656/28170/14350	1094/1205/1279 1526/1564/1592

表2：该表使用不同的训练集大小在电影镜头数据集上通过各种模型报告NDCG@10。报告了10倍以上的平均值和标准差。

方法	N=10	N=20	N=50
基线	0.6916±0.0051	0.7087±0.0031	0.7317±0.0053
Cof等级	0.6994±0.0079	0.7125±0.0031	0.7204±0.0040
CR \rightarrow 最惠	0.7026±0.0022	0.7103±0.0021	0.7242±0.0051
国 CR \rightarrow FL	0.7220±0.0043	0.7202±0.0031	0.7316±0.0070
CR \rightarrow 最惠	0.7023±0.0040	0.7129±0.0033	0.7312±0.0053
国 CR \rightarrow FL	0.7184±0.0031	0.7221±0.0034	0.7360±0.0058

表3：该表使用不同的训练集大小，在每个电影数据集上通过各种模型报告NDCG@10。报告了10倍以上的平均值和标准差。

方法	N=10	N=20	N=50
基线	0.6950±0.0019	0.7028±0.0006	0.7269±0.0007
Cof等级	0.7109±0.0050	0.7271±0.0074	0.7346±0.0077
CR \rightarrow 最惠	0.6959±0.0021	0.7409±0.0013	0.7469±0.0013
国 CR \rightarrow FL	0.7205±0.0010	0.7620±0.0005	0.7593±0.0013
CR \rightarrow 最惠	0.7038±0.0016	0.7359±0.0007	0.7367±0.0010
国 CR \rightarrow FL	0.7200±0.0012	0.7442±0.0004	0.7503±0.0006