Kernel-CF：借助社交网络分析和核平滑，实现真正有效的协同过滤

Hao Wang Ratidar.com,
北京，中国*
haow85@live.com

摘要

协同过滤是推荐系统领域中最简单、历史最悠久的机器学习算法。尽管历史悠久，它仍然是学术研究中的热门议题。通常，人们会使用与目标客户相似度分数大于 0 的用户或物品来计算算法。然而，经过仔细审视后，这未必是最优方案。本文将推荐系统的输入数据转换为二维社交网络，并应用核平滑来计算用户-物品评分矩阵中未知值对应的偏好。我们统一了推荐系统与非参数统计的理论框架，并提供一套算法流程和最优参数选择方法，以实现这一目标。

1 引言

推荐系统是一项蓬勃发展的技术，凭借在亚马逊和 TikTok 等公司的广泛应用，深刻影响着我们的日常生活。关于这项技术的文献浩如烟海，商业化相关理念也吸引了巨额投资。推荐系统的基本思想是利用数据挖掘算法，根据用户过去的偏好和历史数据向其推荐物品。

协同过滤是历史最悠久的推荐系统技术之一，至今仍在学术研究领域被广泛讨论。尽管世界上大多数公司都已在其在线系统中采用深度学习模型，研究出版物中仍不断出现各种细微改进。

协同过滤至今仍在应用，原因在于它简单且易于解释。基于用户的协同过滤模型会利用某个用户与其他偏好相似用户之间的相似度作为权重，为该用户历史数据中未出现的物品计算偏好分数，并向该用户推荐偏好分数最高的物品。基于物品的协同过滤理念也十分相似，因此这里不再赘述。

协同过滤既可以作为独立模型使用，也可以作为混合模型的首个候选项选择步骤，之后再由算法采用排序方法。作为候选项选择模型，协同过滤可以大幅减少后续排序阶段需要处理的候选物品数量，使整个系统能够处理规模大得多的数据集，并且比独立模型具有更高的准确率。

在基于用户的协同过滤中，相似度计算步骤会根据相似度得分是否为正来筛选候选物品。然而，理论上这种做法未必是最优选择。候选集规模过小，除了会降低准确率，还可能导致稀疏性问题。另一方面，如果候选集规模过大，系统中就会引入噪声，进而降低算法性能。对基于物品的协同过滤进行的分析与此类似。

近年来，人们尝试将深度学习与协同过滤结合，并对算法进行其他改进，我们将在下一节详细阐述。研究人员也运用统计学概念及其变体来改进浅层模型。然而，这些改进几乎都只是小幅调整。

在我们的研究中，我们发现协同过滤方法可以推广到比其当前形式更广泛的范畴。因此，我们旨在借助社会网络分析理论和非参数统计（尤其是核平滑），提出一种更全面的理论。我们在研究中进行了详细分析，并通过理论论证向读者证明，协同过滤与非参数统计之间存在一个统一的理论框架，可通过可视化将二者联系起来。

2 相关工作

协同过滤是一种经典的推荐系统技术，长期以来一直应用于工业界和学术界。最早的协同过滤方法包括基于用户的协同过滤[1] 和基于物品的协同过滤 [2]。为了使协同过滤更具适应性和效率，研究人员采用了许多不同的方法 [3][4]来提升算法的整体性能。

在深度学习时代， [5] [6] ，协同过滤可能显得有些过时。然而，关于这项技术，仍有许多基本问题尚未得到解决。2018年，Wang 等人 [7] 首次分析了协同过滤中的流行度偏差，并量化了

其影响。其他问题，例如最优邻域选择，以及将协同过滤框架与其他理论统一起来，仍有待解决，本文将在以下章节中回答这些问题。

本文所介绍方法的中间步骤之一是图可视化。Gephi 等热门软件在其工具中集成了许多可视化功能。Force Atlas 2 [8] ,等算法已广泛应用于图可视化领域。

本文采用的另一项重要技术是非参数统计 [9] [10]。具体而言，我们应用核平滑方法重新定义协同过滤，并为协同过滤选择最佳邻域。

3 协同过滤

协同过滤分为两种：基于用户的协同过滤（user-CF）和基于物品的协同过滤（item-CF）。基于用户的协同过滤通过比较用户的历史数据来计算用户之间的相似度，并使用加权平均公式向用户推荐物品。基于用户的协同过滤的形式化定义如下：

$$R_{i,j} = \frac{\sum_{k=1}^{N} sim(user_i, user_k) R_{k,j}}{\sum_{k=1}^{N} sim(user_i, user_k)} \quad (1)$$

计算这些值的关键在于了解 。计算相似度值的常用方法包括 Jaccard距离和余弦距离，以及其他多种选择。

从公式（1）可以明显看出，相似度值为正是 的一个阈值。一个不易察觉的问题是，这样的阈值是否适用于基于用户的协同过滤算法。阈值选择不当会导致欠拟合或过拟合问题。

与基于用户的协同过滤一样，基于物品的协同过滤也需要计算相似度值。与基于用户的协同过滤不同，基于物品的协同过滤关注的是物品数据之间的相似度。该算法的形式化定义如下：

$$R_{i,j} = \frac{\sum_{k=1}^{N} sim(item_i, item_j) R_{i,k}}{\sum_{k=1}^{N} sim(item_i, item_j)} \quad (2)$$

同样，计算物品对之间相似度值的常用方法是Jaccard距离和余弦距离。不出所料，基于物品的协同过滤也需要处理过拟合／欠拟合问题。

能否将协同过滤算法定义为核平滑问题？这些公式看起来无论如何都很像 Nadaraya-Watson 估计量。答案是，如果我们想将用户-物品评分矩阵网格转换到另一个空间，就需要进行一些变换，才能将协同过滤算法转化为核平滑问题。

原始的用户－物品评分网格由 (i, j) 对定义，其中 i 是用户 ID，j 是物品 ID，而 (i,j) -> 是定义在该网格上的标量函数。这个网格无法进行平滑处理，因为 (i,j) 的邻域，即 (i+1,j)、(j+1,i) 等，并非协同过滤算法中 (i,j) 的真实邻域。为了解决这个问题，我们借助社交网络可视化算法，将用户－物品评分网格转换到另一个空间，使得 (i, j) 的邻域成为协同过滤公式所定义的邻域。

4 社交网络可视化

社交网络可视化是一项商业化技术。Gephi 等知名社交网络分析软件为用户提供了多种可视化技术。本文使用 ForceAtlas 2 对基于用户的协同过滤中的用户间相似度矩阵进行可视化。通过有效地将相似度矩阵可视化，我们能够获取用户的点坐标，并据此计算用户的真实邻域。不出所料，使用新的坐标系后，协同过滤算法只不过是 Nadaraya-Watson估计量的一种特殊类型。这个新的坐标系使我们能够使用 Nadaraya-Watson估计量之外的非参数统计方法。

经典的 Force Atlas 2 算法将社交网络数据的可视化布局视为两种力之间的相互作用：吸引力和排斥力。吸引力定义如下：

$$F_a(n_1, n_2) = d(n_1, n_2) \quad (3)$$

排斥力定义如下：

$$F_r(n_1, n_2) = k_r \frac{(deg(n_1)+1)(deg(n_2)+1)}{d(n_1, n_2)} \quad (4)$$

公式（3）和（4）中的变量定义如下：和分别表示用户，d(表示距离，deg() 表示该用户在社交网络图中的度。该算法的灵感来自现实世界中带电粒子相互作用的物理现象。

请注意，我们也可以使用多维尺度分析或等距映射算法来可视化推荐系统相似度矩阵。这两种算法都能在低维空间中保留社交网络节点之间的距离／相似度。不过，它们的速度比 Force Atlas 2 慢得多，因此在大型互联网公司的真实商业环境中，不可能对数亿用户运行这些算法。

图1展示了在 LDOS-CoMoDa 数据集上运行 ForceAtlas 2 模拟的快照：

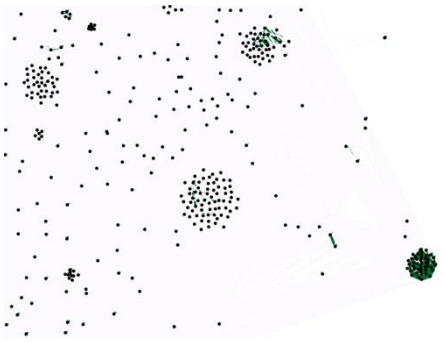

图 1 LDOS-CoMoDa 数据集上的 Force Atlas 2 模拟

从图1中，我们可以观察到用户-用户相似度图中的用户具有聚类特性。在可视化中，相似的用户聚集在一起，而不相似的用户则彼此分开。一个聚类中彼此相邻的点（用户）自然构成了图中某个点（用户）的邻域。协同过滤算法本质上就是可视化图中的Nadaraya-Watson估计量，而该图充当一种新的坐标空间，由原始用户-物品评分矩阵网格转换而来。

5 核平滑

在上一节中，为了计算用户-用户／物品-物品相似度的可视化图，我们需要知道图中两点之间的距离。然而，我们手头已有的是相似度数据。我们采用一种常见做法，用相似度的倒数来计算距离：

$$d(n_1, n_2) = \frac{1}{sim(n_1, n_2)} \quad (5)$$

为了建立协同过滤算法与***Nadaraya-Watson*** 估计量之间的联系，我们先回顾一下 Nadaraya-Watson估计量的定义：

$$\hat{m} = \frac{\sum_{i=1}^{n} K_{h_x}(x-x_i)y_i}{\sum_{i=1}^{n} K_{h_x}(x-x_i)} \quad (6)$$

如果用 sim(x,) 替换核函数，就可以得到协同过滤算法。最直接的替换方式，是使用基于隐式反馈计算出的欧几里得距离或 Jaccard距离相似度。

根据上述推理，如果将原始用户-物品评分矩阵网格转换到社交网络图坐标系，并使用用户-用户／物品-物品对的新坐标，我们就成功地将原始协同过滤算法转化为 Nadaraya-Watson估计问题。

在我们的降维框架中，还存在一个问题：如何为 Nadaraya-Watson 核选择最优邻域大小。用协同过滤算法的术语来说，这相当于如何选择在相似度计算步骤中使用的用户数／物品数。

一维域中 Nadaraya-Watson估计量核的最优邻域大小计算如下：

$$h_* = \left(\frac{1}{n}\right)^{1/5} \left(\frac{\sigma^2 \int K^2(x)dx \int dx/f(x)}{(\int x^2 K^2(x)dx)^2 \int \left(r''(x) + 2r'(x)\frac{f'(x)}{f(x)}\right)^2 dx} \right)^{1/5} \quad (7)$$

在公式（7）中，h_*是 Nadaraya-Watson估计量的最优带宽；K 是核函数；x 表示用户或物品；f() 是对 x 进行采样所依据的概率分布函数；r 是按如下方式定义的函数：

$$r(x) = \sum_{i=1}^{n} K_{h_x}(x - x_i)y_i \quad (8)$$

该算法需要估计概率密度函数 f()，该函数可通过核密度估计计算得出。

现在，我们将一维最优带宽的结果推广到二维域。二维域是社交网络图的维度，也是我们为协同过滤算法设定的新坐标系。在二维情况下，我们有两个最优带宽b_u，需要计算：

$$b_t = \left(\frac{\sigma^2 R(K)^2 I_{uu}^{3/4} I_f}{\mu_2(K)^2 I_{tt}^{3/4} \left(I_{tt}^{1/2} I_{uu}^{1/2} + I_{tu}\right) n} \right)^{1/6} \quad (9)$$

$$b_u = (I_{tt}/I_{uu})^{1/4} b_t \quad (10)$$

公式（9）和（10）中的变量定义如下：

$$I_{tt} = \int_A \omega(t,u) r^{(2,0)}(t,u)^2 d(t,u) \quad (11)$$

$$I_{uu} = \int_A \omega(t,u) r^{(0,2)}(t,u)^2 d(t,u) \quad (12)$$

$$I_{uu} = \int_A \omega(t,u) r^{(2,0)}(t,u) r^{(0,2)}(t,u) d(t,u) \quad (13)$$

$$I_f = \int_A \omega(t,u) f(t,u)^{-1} d(t,u) \quad (14)$$

$$R(K) = \int_{-1}^{1} K(x)^2 dx \quad (15)$$

变量 r 表示二维域中点数据的底层函数。二维域中的 Nadaraya-Watson 估计正式定义如下：

$$r(t, u; b_t, b_u) = b_t^{-1} b_u^{-1} \sum_{i=1}^{n} \int_{A_i} K\left(\frac{t-v}{b_t}\right) K\left(\frac{u-w}{b_u}\right) d(v,w) Y_i \quad (16)$$

该变量是特意引入的函数，用于将泛函限制在区域 A 上。 和 是 r 的二阶偏导数。我们假设社交网络顶点受到如下扰动：

$$Y_i = r(t_i, u_i) + \varepsilon_i \quad (17)$$

函数 f 定义为数据噪声的概率密度函数。

在完成最优带宽／邻域选择的数学推导后，我们需要将理论付诸实践。换句话说，我们需要找到方法来近似公式（9）和（10）中的未知值。为了近似 r 的偏导数，我们

使用普通最小二乘法计算底层函数 r 及其导数。为了估计密度函数 f，我们使用核密度估计：

$$f(\mathbf{x},\mathbf{H}) = \frac{1}{n}\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{x} - \mathbf{X_i}) \quad (18)$$

我们应用 参考规则，按如下方式计算**H** 的最优值：

$$\boldsymbol{H}_{REF} = \left(\frac{1}{n}\right)^{1/3} \widehat{\Sigma} \quad (19)$$

，其中是经验协方差矩阵：

$$\widehat{\Sigma} = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})(X_i - \bar{X})^T, \text{ with } \bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

请注意，我们将协同过滤算法简化为核平滑问题的方法，可以推广到多项式阶数大于 0 的核平滑，例如局部加权线性回归。同样，最优邻域大小选择问题可以采用代入法或交叉验证法解决，如 [9]和 [10]中所述。

我们将这种取代协同过滤算法的新型非参数方法命名为 Kernel CF。其正式流程实际上是一种在线学习算法（我们仅说明基于用户的 Kernel CF 流程。基于物品的 Kernel CF 的执行方式类似。）：

1. 使用经典的基于用户的协同过滤计算每位用户的邻域。2. 将现有的用户-物品评分矩阵网格转换到由 Force Atlas 2 算法定义的新坐标系中。3. 对于新坐标系中的每位用户，根据经典协同过滤算法生成的候选列表，使用 Nadaraya-Watson 核计算每位用户的最优邻域；最优带宽由公式（9）和（10）计算得出。4. 使用第 3 步得到的新最优邻域向用户推荐物品。

我们的算法实际上是一种为现有协同过滤寻找最优候选池的方法。最初，这项任务难以实现，因为研究人员找不到合适的坐标系，无法应用插入法等非参数统计方法。

6 结论

本文将经典协同过滤与非参数统计相结合，并给出了计算最优邻域的公式。该算法分为两个步骤：1. 使用经典协同过滤方法计算候选池。2. 使用本文所讨论的非参数统计工具重新筛选候选池。

这是人工智能发展史上首次通过连接三个不同科学领域来解决实际人工智能问题，即推荐系统、信息可视化和非参数统计。我们希望更多研究人员在阅读本文后受到启发，探索多学科交叉科学。

在未来的工作中，我们希望研究其他推荐算法，例如矩阵分解、因子分解机和深度学习，以探究上述所有技术背后是否存在统一的理论。

7 文献

[1]Z.D. Zhao、M.S. Shang，2010年。基于 Hadoop 的用户协同过滤推荐算法。第三届知识发现与数据挖掘国际会议[2]B.Sarwar、G.Karypis 等，2001年。基于物品的协同过滤推荐算法。《万维网国际会议论文集》 10th[3]X.He、L.Liao 等，2017年。神经协同过滤。《万维网国际会议论文集》 26th [4] H.Zhang、F.Shen 等，2016年。离散协同过滤。SIGIR。[5]H.T.Cheng、L.Koc 等，2016年。面向推荐系统的 Wide & Deep Learning。《深度学习推荐系统研讨会论文集》 1st [6]H.Guo、R.Tang 等，2017年。DeepFM：一种基于因子分解机的点击率预测神经网络。IJCAI。[7]H. Wang、Z.H. Wang、W.S. Zhang，2018年。推荐系统中马太效应与稀疏性问题的定量分析。ICCCBDA。[8] M.Jacomy、T.Venturini 等，2014年。ForceAtlas2：一种为 Gephi 软件设计的连续图布局算法，便于网络可视化。PLoS One。[9]M.Brockmann、T.Gasser 等，1993年。核回归估计量的局部自适应带宽选择。美国统计学会期刊。[10]H.Ivanka，2012年。Matlab 中的核平滑：核平滑理论与实践。