

基于矩阵分解优化的排序学习特征构造方法

杨 潇¹ 崔超然² 王帅强^{3,4}

(山东财经大学管理科学与工程学院 济南 250014)¹

(山东财经大学计算机科学与技术学院 济南 250014)²

(齐鲁工业大学金融学院 济南 250014)³ (曼彻斯特大学曼彻斯特商学院 曼彻斯特 M13 9SS)⁴

摘 要 在排序学习中引入特征选择可以提高学习的效率和准确率。出于对选择速度的考虑,当前的研究主要从特征选择的角度出发,根据特征对排序的作用和特征之间的相似性选择对排序区分度最大的特征集合。由于特征大都是人工归纳的,因此特征和特征之间难免存在重叠和冗余。为了减少特征之间的冗余,从特征生成的角度出发,对现有特征进行矩阵分解,从而生成新的特征集。考虑到使用奇异值分解(Singular Value Decomposition SVD)等方法进行矩阵分解时不能综合考虑排序结果对特征的影响,基于特征矩阵对排序的效果、特征矩阵与原矩阵之间的差距来构造优化算法,提出了一种基于矩阵分解的排序学习优化方法,并根据该优化方法设计了排序学习特征选择算法MFRank。实验中使用映射随机梯度下降法近似求得优化问题的最优值,在公开测试集MQ2008上的结果显示,所提MFRank方法获得了与当前最优的特征选择方法即RankBoost和RankSVM-Struct等排序算法相当的结果。

关键词 特征生成,排序学习,矩阵分解,优化

中图分类号 TP391 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2017.12.046

Feature Construction Method for Learning to Rank Based on Optimization of Matrix Factorization

YANG Xiao¹ CUI Chao-ran² WANG Shuai-qiang^{3,4}

(School of Management Science and Engineering, Shandong University of Finance and Economics, Jinan 250014, China)¹

(School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan 250014, China)²

(Department of Finance Management, Qilu University of Technology, Jinan 250014, China)³

(Alliance Manchester Business School, University of Manchester, Manchester M13 9SS, UK)⁴

Abstract Feature selection can improve ranking efficiency and accuracy. Current study mainly prefers selecting the most distinguishing feature set rather than feature construction, where the selection is mostly according to the significance of features and the similarity between features. Since the features are mostly induced manually, there are inevitably overlap and redundancy between them. In order to reduce the redundancy, matrix decomposition is used to generate new features set. An optimization algorithm was designed according to the effect of the feature matrix decomposed, and the gap between the decomposed feature matrix and the original matrix. Then a matrix decomposition based optimization for learning to rank, which named by MFRank, was proposed to take into account, the ranking result acquired by the features, which cannot be handled by matrix decomposition method such as singular value decomposition (SVD), etc. A stochastic projective sub-gradient algorithm was used in experiments to obtain the approximate optimal values for the optimization problems, and experimental result on MQ2008, which is an open test set, shows that the proposed MFRank algorithm can obtain comparative result as RankBoost, RankSVM-Struct which are the state-of-the-art algorithms.

Keywords Feature construction, Learning to rank, Matrix factorization, Optimization

1 前言

排序学习方法目前已经比较成熟,如 AdaBoost, ListNet, RankSVM 等都获得了很好的效果,已经被应用于推荐^[1-2]、信息检索^[3]等领域。但目前用于排序的特征有上百种,并且

这些特征之间并不是相互独立的,例如在 LETOR^[4] 提供的 136 个特征中,词频的和(sum of term frequency)与词频的均值(mean of term frequency), $tf * idf$ 的和(sum of $tf * idf$), $tf * idf$ 的均值(mean of $tf * idf$)与 $tf * idf$ 的方差(variance of $tf * idf$)等特征之间都有很强的关联关系,但又不能相互替代。

到稿日期:2016-10-08 返修日期:2017-02-19 本文受国家自然科学基金项目:基于机器学习融合精确性和多样性的电子商务协同过滤推荐方法研究(71402083),山东省高等学校科技计划项目:基于语义角色主题模型的细粒度情感分析研究(J15LN56)资助。

杨 潇(1981—),女,博士,讲师,CCF 会员,主要研究方向为自然语言处理, E-mail: yangxiao@sdufe.edu.cn; 崔超然(1985—),男,博士,教授,主要研究方向为图像检索; 王帅强(1981—),男,博士,讲师,主要研究方向为推荐算法、排序学习、社交网络。

随着排序函数中使用的特征数量的持续增长,选择有效的特征集成为了瓶颈问题^[5]。

在排序问题中,作为分类对象的排序关系是一种有序的集合,与普通的分类问题中类别之间是无序的扁平结构不同。在分类问题中,对所有的实例的分类是同等重要的,因此在分类中精确率和召回率都很重要;而在排序问题中则主要关注前 n 项的排序结果^[6],精确率比召回率更重要^[7]。普通分类问题中的特征选择方法不适用于排序学习的特征选择问题。另外,排序问题中采用的评价标准 MAP 和 NDCG 与分类中所用的评价标准也不同。

特征处理的方法主要包括特征选择 (Feature Selection) 和特征构造 (Feature Construction) 两大类。特征选择根据某个评价函数,从原始数据特征空间中去除冗余的或者不相关的特征,以达到降低特征空间维数的目的。特征构造又称为特征提取,是一种特征的变换方法,是采用空间映射或者空间变换等方式将数据集从高维特征空间转换成低维空间表示的过程。

现有的排序学习对特征处理的研究大都采用特征选择方法。例如:Geng^[8]提出了一种特征选择的贪心算法来选择重要性分数最大和相似性分数最小的特征子集,其中重要性指标由使用该特征得到的排序结果的 MAP 和 NDCG 表示,而相似性指标由排序结果列表之间的 Kendall-tau 度量表示。将最大化重要性指标和最小化相似性指标的问题转化为线性优化问题,使用构建无向图的贪心算法来解决该问题。

Dang 等人^[9]通过最优优先搜索算法将特征增量式地分成子集,然后通过最大化某些排序标准的方式将子集内的特征组合成单个特征,并使用坐标下降法进行优化。Li^[10]提出了一种排序-组合的方法,首先将特征分成单个的子集,然后按照结果进行排序,再递归地合成相邻的子集以生成新的特征集。Silva^[11]提出了一种基于遗传算法来进行特征选择的方法。Lai^[12]以皮尔逊相关系数为度量,将重要性和相似性统一到联合优化问题中,并使用 Nesterov's 算法来解决该优化问题;在文献中作者证明该问题是凸的,有良好的理论基础,并且该方法由于不限制重要性和相似性指标,因此具有良好的可扩展性。Pan^[3]则比较了不同的特征选择方法。上述方法都是基于特征选择的思想,并未涉及对特征进行重新构造。

排序学习中使用特征构造方法的研究主要是基于矩阵分解 (Matrix Factorization, MF) 的方法,将矩阵分解成两个或三个矩阵的形式。例如,花贵春等^[13]使用主成分分析法 (PCA) 和 MAP 等 4 种技术对特征进行重组;林原等^[14]将奇异值分解 (Singular Vector Decomposition SVD) 应用于未标注的数据集来提高 RankBoost 排序算法的性能。在这些方法中,使用 SVD 与 PCA 提取的主特征向量有着相似的性质^[15]。

SVD 和 PCA 等矩阵分解方法实现简单、预测准确度高、扩展性强,广泛应用于特征选择、基于语言模型的文档表示和推荐系统^[13,16]中;但其忽略了排序学习中非常重要的偏序关系^[2],也不能根据排序结果的影响对分解到的矩阵进行修正。因此,排序学习的特征处理更偏向于特征选择方法,很少有从特征重构的角度来构建更有效的排序函数^[13]。

考虑到特征之间的冗余和交叉,使用特征构造方法更能有效地进行特征的处理。针对当前特征构造方法忽视排序学习中偏序关系的问题,选择优化方法对矩阵进行分解。将排序学习中的偏序关系和排序结果纳入矩阵分解的优化问题中,使用优化方法选择对排序效果最优的特征分解矩阵作为特征构造的结果。

2 基于矩阵分解的优化方法

2.1 排序学习

首先介绍排序学习中用到的符号。 $S = \{(q_k, \hat{X}^{(q_k)}, Y^{(q_k)})\}_{k=1}^n$ 表示训练集,其中 q_k 表示查询; $\hat{X}^{(q_k)} = \{\hat{X}_i^{(q_k)}\}_{i=1}^{n(q_k)}$ 表示查询 q_k 对应的检索项, $\hat{X}_i^{(q_k)} \in \mathbb{R}^d$ 为查询 q_k 的第 i 个特征向量,且 $\hat{X}_i^{(q_k)}$ 的各特征投影到 $[0, 1]$ 之间; $Y^{(q_k)} = \{Y_i^{(q_k)}\}_{i=1}^{n(q_k)}$ 为人工标注的 $\hat{X}^{(q_k)}$ 对应的相关性标签, $n(q_k)$ 为查询 q_k 所对应的检索项的数目, $Y_i^{(q_k)}$ 为查询 q_k 的第 i 个元素对应的相关性标签; $T = \{(q_k, \hat{X}^{(q_k)})\}_{k=n+1}^{n+u}$ 表示测试集。

为了表述方便,不失一般性,将 $\hat{X}^{(q_k)}$ 表示为 $\mathbf{X}$,将 $Y^{(q_k)}$ 表示为 $\mathbf{y}$ 。用小写粗体字母表示向量,例如 $\mathbf{x}$;用大写粗体字母表示矩阵,例如 $\mathbf{X}$;用大写粗体字母加下标表示矩阵的列元素,例如 $\mathbf{X}_i$ 表示矩阵 $\mathbf{X}$ 的第 i 列元素;对于任意向量 $\mathbf{x}$ 和矩阵 $\mathbf{X}$, $\|\mathbf{x}\|_2 = \sqrt{\sum_i x_i^2}$ 为 $\mathbf{x}$ 的 l_2 范数。

2.2 矩阵分解

矩阵分解 (MF) 可以将排序特征矩阵近似分解为两个高质量的低维特征矩阵表示,类似于推荐系统,称分解到的特征为潜在语义特征。令排序学习查询 q_k 对应的文档-特征矩阵为 $\mathbf{X}^{(q_k)} = (\mathbf{x}_{n \times m}^{(q_k)}) \in [0, 1]^{n \times m}$,假设可以将其近似分解为 $\mathbf{X} \approx \mathbf{V}^T \mathbf{U}$,其中 $\mathbf{U} \in \mathbb{R}^{l \times m}$ 是一个 $l \times m$ 的矩阵,表示 m 个文档在 l 维潜在语义特征上的向量集合; $\mathbf{V} \in \mathbb{R}^{l \times n}$ 是一个 $l \times n$ 的矩阵,表示 l 个潜在语义特征在源特征上的集合,其中 $l < \min(m, n)$ 。

为了确保求得的潜在语义特征为唯一解,即潜在语义特征无二义,在矩阵分解时,将潜在语义特征矩阵 $\mathbf{V}$ 的列向量限制为两两正交的单位向量,即 $\mathbf{V}^T \mathbf{V}$ 为对角矩阵。

$$\frac{\mathbf{V}^T \mathbf{V}}{|\mathbf{V}^T \mathbf{V}|} = \mathbf{I} \quad (1)$$

为了处理的方便性,文中未将式(1)作为优化函数的约束条件,而是在每次优化迭代时将 $\mathbf{V}$ 的列向量映射为单位向量,称为映射梯度下降。

2.3 优化函数

为了求得最优的潜在语义特征,令分解得到的文档特征矩阵 $\mathbf{U}$ 进行排序获得的损失函数 $f(\mathbf{w}, (\mathbf{U}, \mathbf{y}))$ 最小,且分解后的矩阵乘积 $\mathbf{V}^T \mathbf{U}$ 与原特征矩阵的差最小。可以构造如下多目标优化函数:

$$\min_{\mathbf{w}, \mathbf{U}, \mathbf{V}} f(\mathbf{w}, (\mathbf{U}, \mathbf{y})), \min_{\mathbf{U}, \mathbf{V}} \|\mathbf{V}^T \mathbf{U} - \mathbf{X}\|^2 \quad (2)$$

由于 $f(\mathbf{w}, (\mathbf{U}, \mathbf{y}))$ 与 $\|\mathbf{V}^T \mathbf{U} - \mathbf{X}\|^2$ 之间无约束关系,使用加权法^[17]求解上述多目标优化问题,得到:

$$\min_{\mathbf{w}, \mathbf{U}, \mathbf{V}} f(\mathbf{w}, (\mathbf{U}, \mathbf{y})) + \alpha \cdot \|\mathbf{V}^T \mathbf{U} - \mathbf{X}\|^2 \quad (3)$$

其中, α 为权重因子, 表示 $f(w, (U, y))$ 和 $\|V^T U - X\|_2^2$ 的重要性。不失一般性, 假设 $f(w, (U, y))$ 的权重为 1。目前有许多方法可以确定权重因子的值, 例如可以由设计者根据其重要性进行人工赋值, 本文假设在该值已知的情况下考虑优化问题。

为了减轻模型的过拟合问题, 首先为 w 和 U 添加正则项, 然后将惩罚因子除以 2 以使计算更加方便。此时式(3)可以表示为:

$$\min_{w, U, V} f(w, (U, y)) + \alpha \cdot \|V^T U - X\|_2^2 + \frac{\lambda_1}{2} \|w\|_2^2 + \frac{\lambda_2}{2} \|U\|_2^2 \quad (4)$$

其中, $\lambda_1, \lambda_2 > 0$, 为正则化项的惩罚因子。 λ_1, λ_2 越大, 模型越简单, 模型的泛化能力越强; λ_1, λ_2 越小, 对问题的描述能力越强。为了在经验风险最小化和结构风险最小化之间取得折中, 本文通过交叉验证方法估算最优的 λ_1 和 λ_2 。

定义拉格朗日函数 $L(w, U, V, X, y, \alpha, \lambda_1, \lambda_2)$ 为:

$$L(w, U, V, X, y, \alpha, \lambda_1, \lambda_2) = \min_{w, U, V} f(w, (U, y)) + \alpha \cdot \|V^T U - X\|_2^2 + \frac{\lambda_1}{2} \|w\|_2^2 + \frac{\lambda_2}{2} \|U\|_2^2 \quad (5)$$

2.4 双铰链损失函数

类似于文献[12], 本文可以使用双铰链损失函数作为损失函数 $f(w, (U, y))$:

$$f(w, (U, y)) = \frac{1}{p} \sum_{(U_i^{(q_k)}, y_i^{(q_k)}, U_j^{(q_k)}, y_j^{(q_k)}) \in P} (1 - y_{i,j}^{(q_k)} \langle w, U_i^{(q_k)} - U_j^{(q_k)} \rangle)^+ \quad (6)$$

其中, p 为 P 中排序对的个数, $(x)^+ = \max(0, x)$ 。

在式(6)中, 当且仅当 $(U_i^{(q_k)}, y_i^{(q_k)}), (U_j^{(q_k)}, y_j^{(q_k)})$ 属于同一个查询 q_k , 并且满足条件 $y_i^{(q_k)} \neq y_j^{(q_k)}$ 时, $((U_i^{(q_k)}, y_i^{(q_k)}), (U_j^{(q_k)}, y_j^{(q_k)})) \in P$ 成立。且

$$y_{i,j}^{(q_k)} = \begin{cases} 1 & y_i^{(q_k)} > y_j^{(q_k)} \\ -1 & y_i^{(q_k)} \leq y_j^{(q_k)} \end{cases}$$

因为下文中的所有 $U_i^{(q_k)}$ 和 $y_i^{(q_k)}$ 都是针对查询 q_k 的, 为了表述简单, 下文将 $U_i^{(q_k)}$ 简写为 U_i , $y_i^{(q_k)}$ 简写为 y_i 。又由于损失函数 $f(w, (U, y))$ 中使用的双铰链函数涉及到矩阵 U 的列向量, 为统一, 将损失函数分解为向量形式, 因此式(5)可写为:

$$L_1(w, U, V, X, y, \alpha, \lambda_1, \lambda_2) = \frac{1}{p} \sum_{(U_i, y_i), (U_j, y_j) \in P} (1 - y_{i,j} \langle w, U_i - U_j \rangle)^+ + \alpha \cdot \sum_{k=1}^n \sum_{i=1}^m \|V_k^T \cdot U_i - X_{k,i}\|_2^2 + \frac{\lambda_1}{2} \|w\|_2^2 + \frac{\lambda_2}{2} \|U\|_2^2 \quad (7)$$

2.5 优化过程

鉴于偏序对的数量太大, 在大量的偏序对上更新变量的花费过高, 类似于文献[18], 本文使用随机子梯度方法寻找该函数的最小值, 并将变量 V 的列向量映射到单位向量上, 称为映射随机梯度下降法 (Projected Stochastic Gradient Descent)。

对于一个训练实例 $((U_i, y_i), (U_j, y_j)) \in P$, 其拉格朗日函数为:

$$L(w, U, V, X, y, \alpha, \lambda_1, \lambda_2) = (1 - y_{i,j} \langle w, U_i - U_j \rangle)^+ + \alpha \cdot \sum_{k=1}^n \|V_k^T \cdot U_i - X_{k,i}\|_2^2 + \frac{\lambda_1}{2} \|w\|_2^2 + \frac{\lambda_2}{2} \|U\|_2^2 \quad (8)$$

根据 KKT 条件, $L(w, U, V, X, y, \alpha, \lambda_1, \lambda_2)$ 对于 U_i, U_j 和 w 的梯度应为 0, 因此有: (1) 如果 $\langle w, U_i - U_j \rangle > 1$, 或者 $\langle w, U_j - U_i \rangle > 1$, 则

$$\begin{cases} \nabla_{U_i} L = \alpha \cdot \sum_{k=1}^n (V_k^T \cdot U_i - X_{k,i}) \cdot V_k + \lambda_2 U_i = 0 \\ \nabla_{U_j} L = \alpha \cdot \sum_{k=1}^n (V_k^T \cdot U_j - X_{k,j}) \cdot V_k + \lambda_2 U_j = 0 \end{cases} \quad (9)$$

(2) 如果 $\langle w, U_i - U_j \rangle \leq 1$, 则

$$\begin{cases} \nabla_{U_i} L = -w y_{ij} + \alpha \cdot \sum_{k=1}^n (V_k^T \cdot U_i - X_{k,i}) \cdot V_k + \lambda_2 U_i = 0 \\ \nabla_{U_j} L = w y_{ij} + \alpha \cdot \sum_{k=1}^n (V_k^T \cdot U_j - X_{k,j}) \cdot V_k + \lambda_2 U_j = 0 \end{cases} \quad (10)$$

在这两种情况下, 其他变量的梯度相同, 均为:

$$\nabla_w L = (-y_{ij}) \cdot (U_i - U_j) + \lambda_1 w = 0 \quad (11)$$

$$\nabla_{V_k} L = \alpha \cdot (V_k^T \cdot U_i - X_{k,i}) U_i + \alpha \cdot (V_k^T \cdot U_j - X_{k,j}) U_j = 0 \quad (12)$$

由于上面的方程组无形式解, 因此可以使用子梯度下降方法来近似求解。第 $t+1$ 次迭代的变量值可由第 t 次迭代的变量值计算得到。

如果 $\langle w, U_i - U_j \rangle > 1$, 或者 $\langle w, U_j - U_i \rangle > 1$, 则

$$U_{i,t+1} = U_{i,t} + \eta_t \cdot w_t \cdot y_{ij} - \eta_t \cdot \alpha \cdot \sum_{k=1}^n (V_{k,t}^T \cdot U_{i,t} - X_{k,i}) \cdot V_{k,t} - \eta_t \cdot \lambda_2 \cdot U_{i,t} \quad (13)$$

$$U_{j,t+1} = U_{j,t} - \eta_t \cdot w_t \cdot y_{ij} - \eta_t \cdot \alpha \cdot \sum_{k=1}^n (V_{k,t}^T \cdot U_{j,t} - X_{k,j}) \cdot V_{k,t} - \eta_t \cdot \lambda_2 \cdot U_{j,t} \quad (14)$$

否则

$$U_{i,t+1} = U_{i,t} - \eta_t \cdot \alpha \cdot \sum_{k=1}^n (V_{k,t}^T \cdot U_{i,t} - X_{k,i}) \cdot V_{k,t} - \eta_t \cdot \lambda_2 \cdot U_{i,t} \quad (15)$$

$$U_{j,t+1} = U_{j,t} - \eta_t \cdot \alpha \cdot \sum_{k=1}^n (V_{k,t}^T \cdot U_{j,t} - X_{k,j}) \cdot V_{k,t} - \eta_t \cdot \lambda_2 \cdot U_{j,t} \quad (16)$$

对于其他变量而言,

$$w_{t+1} = (1 - \eta_t \lambda_1) w_t + \eta_t \cdot y_{ij} \cdot (U_{i,t} - U_{j,t}) \quad (17)$$

$$V_{g,t+1} = V_{g,t} - \eta_t \cdot \alpha \cdot (V_{g,t}^T \cdot U_{i,t} - X_{g,i}) U_{i,t} - \eta_t \cdot \alpha \cdot (V_{g,t}^T \cdot U_j - X_{g,j}) U_{j,t} - \eta_t \cdot \sum_{h=1, h \neq g}^n \theta_{gh,t} \cdot V_{h,t} - \eta_t \cdot \sum_{h=1, h \neq g}^n \theta_{hg,t} \cdot V_{h,t} \quad (18)$$

$$V_{h,t+1} = V_{h,t} - \eta_t \cdot \alpha \cdot (V_{h,t}^T \cdot U_{i,t} - X_{h,i}) U_{i,t} - \eta_t \cdot \alpha \cdot (V_{h,t}^T \cdot U_{j,t} - X_{h,j}) U_{j,t} - \eta_t \cdot \sum_{g=1, g \neq h}^n \theta_{gh,t} \cdot V_{g,t} - \eta_t \cdot \sum_{g=1, g \neq h}^n \theta_{hg,t} \cdot V_{g,t} \quad (19)$$

由于上述优化方法使用矩阵分解(MF)的方式进行特征选择优化, 以用于排序学习, 因此将其命名为 MFRank。

3 实验

为了验证本文所提出的矩阵分解特征选择优化方法的有

效性,采用 LETOR^[20]中的几种排序学习的最新研究结果作为基准数据。实验中采用 LETOR 4.0 数据集的 MQ2008 作为训练和测试数据。为方便进行交叉验证,每个数据集预分解为 5 个文件夹,每个文件夹都分别包含训练集/验证集/测试集。实验中选用目前前沿的排序学习算法 ListNet^[20], RankSVM-Struct^[21], RankBoost^[22] 和 AdaRank_NDCG^[23] 作为实验的基准数据,其中 RankSVM-Struct 和 RankBoost 为成对排序学习算法,而 ListNet 和 AdaRank_NDCG 为序列排序学习算法。

3.1 实验数据集

本文实验采用微软亚洲研究院发布的 LETOR 数据集,其作为排序学习领域标准测试集合而广泛地被研究者采用,实验中采用的是最新版本:LETOR4.0,发布于 2009 年 7 月。LETOR4.0 采用 Gov2 的文档集合,有 25205179 个网页,采用 TREC 2007 和 TREC 2008 中 Million QueryTrack 的查询,查询数目分别为 1692 和 784,无论是文档数目还是查询数目,LETOR4.0 都是排序学习领域乃至信息检索领域中规模最大的数据集之一。每个查询、文档对的相关度标注分为 3 级:非常相关、相关和不相关(分别为 2,1,0);该数据集的抽取基于内容、链接等 46 个特征。

3.2 测试标准

实验中采用的排序函数评价指标为排序问题中广泛采用的 MAP 和 NDCG@ n 两种评测函数。

MAP 是信息检索领域最常用的评测函数之一,是在查询集合上的平均准确率。准确率 $j(P@j)^{[7]}$ 表示检索文档中前 j 个文档相关的比例,可由 $P@j = N_{pos}(j)/j$ 计算得到,其中 $N_{pos}(j)$ 表示前 j 个文档中相关文档的个数。给定查询 q_i ,则 q_i 的平均准确率定义为所有 $P@j(j=1,2,\dots,n)$ 的平均值,即:

$$\text{Avg}Pj = \sum_{j=1}^M P@j \times pos(j) / N_{pos}$$

其中, j 为文档位置; M 为检索的文档个数; $pos(j)$ 为指示函数,若位置 j 的文档是相关的,则 $pos(j)=1$,否则 $pos(j)=0$ 。

评测函数 MAP 综合考虑了查询的准确率和召回率。

给定查询 q_i ,定义位置 m 上的 DCG 分数为:

$$\text{DCG}@m = \sum_{j=1}^m \frac{2^{r(j)} - 1}{\log(1+j)}$$

其中, $r(j)$ 为第 j 个文档的分数。

位置 m 上的 NDCG 分数定义为 $\text{NDCG}@m = 1/Z_m \text{DCG}@m$,其中 Z_m 为 $\text{DCG}@m$ 的最大值。这样, $\text{NDCG}@m$ 的值介于 $[0,1]$ 之间。后文将 $\text{NDCG}@m$ 简写为 $N@m$ 。

$\text{NDCG}@m$ 综合考虑了查询与文档之间的相关度和文档在排序中所处的位置对排序效果的影响。

3.3 实验结果

本小节给出了 MFRank 在 MQ2008 数据集上的 5 个文件夹中对数据进行排序而获得的 NDCG 和准确率、MAP 等度量值在位置 1—10 上的详细结果,并给出了 5 个文件夹上的平均值。其中,NDCG 的详细结果如表 1 所列,准确率 Precision 和 MAP 如表 2 所列。

表 1 MFRank 在 MQ2008 数据集上获得的 NDCG@ p

	Folder1	Folder2	Folder3	Folder4	Folder5	Average
N@1	0.3098	0.2420	0.3376	0.3694	0.3673	0.3252
N@2	0.3638	0.2935	0.3540	0.4188	0.4156	0.3691
N@3	0.3983	0.3213	0.3746	0.4481	0.4410	0.3967
N@4	0.4279	0.3304	0.4016	0.4692	0.4647	0.4188
N@5	0.4474	0.3434	0.4262	0.4974	0.4926	0.4414
N@6	0.4622	0.3684	0.4441	0.5150	0.5107	0.4601
N@7	0.4695	0.3828	0.4467	0.5228	0.5159	0.4675
N@8	0.4180	0.3747	0.4297	0.4753	0.4757	0.4347
N@9	0.1988	0.1380	0.2299	0.2769	0.1930	0.2073
N@10	0.2035	0.1420	0.2337	0.2803	0.1985	0.2116
MeanNDCG	0.4502	0.3723	0.4487	0.5125	0.4993	0.4566

表 2 MFRank 在 MQ2008 数据集上获得的 Precision@ p

	Folder1	Folder2	Folder3	Folder4	Folder5	Average
P@1	0.3782	0.3057	0.3885	0.4586	0.4140	0.3890
P@2	0.3782	0.3057	0.3662	0.4427	0.4013	0.3788
P@3	0.3675	0.3015	0.3355	0.4161	0.3737	0.3589
P@4	0.3590	0.2771	0.3248	0.3997	0.3567	0.3435
P@5	0.3423	0.2599	0.3172	0.3822	0.3414	0.3286
P@6	0.3152	0.2548	0.2994	0.3673	0.3195	0.3112
P@7	0.2885	0.2475	0.2784	0.3449	0.2930	0.2905
P@8	0.2700	0.2325	0.2611	0.3272	0.2707	0.2723
P@9	0.2521	0.2201	0.2442	0.3086	0.2519	0.2554
P@10	0.2365	0.2076	0.2293	0.2892	0.2389	0.2403
MAP	0.4401	0.3751	0.4273	0.4968	0.4821	0.4443

除了对所提算法在数据集 MQ2008 上的 5 个文件夹下的 NDCG 和 Precision、MAP 评价指标进行测试外,还将所提算法 MFRank 与目前最先进的几种方法(如 ListNet, AdaRank, RankBoost, RankSVM-Struct)在 MQ2008 数据集上进行了比较,结果如表 3 和表 4 所列。

表 3 各算法在 MQ2008 数据集上获得的 NDCG@ p 比较

	ListNet	AdaRank-NDCG	RankBoost	MFRank	RankSVM-Struct
N@1	0.3754	0.3876	0.3856	0.3252	0.3627
N@2	0.4112	0.3967	0.3993	0.3691	0.3985
N@3	0.4324	0.4044	0.4288	0.3967	0.4286
N@4	0.4568	0.4067	0.4479	0.4188	0.4509
N@5	0.4747	0.4102	0.4666	0.4414	0.4695
N@6	0.4894	0.4156	0.4816	0.4601	0.4851
N@7	0.4978	0.4203	0.4898	0.4675	0.4905
N@8	0.4630	0.4258	0.4568	0.4347	0.4564
N@9	0.2265	0.4319	0.2214	0.2073	0.2239
N@10	0.2303	0.4369	0.2255	0.2116	0.2279
MeanNDCG	0.4914	0.4914	0.4850	0.4566	0.4832

表 4 各算法在 MQ2008 数据集上获得的 Precision@ p 比较

	ListNet	AdaRank-NDCG	RankBoost	MFRank	RankSVM-Struct
P@1	0.4451	0.3826	0.3856	0.3890	0.4273
P@2	0.4120	0.4211	0.3993	0.3788	0.4059
P@3	0.3835	0.4420	0.4288	0.3589	0.3903
P@4	0.3651	0.4653	0.4479	0.3435	0.3696
P@5	0.3426	0.4821	0.4666	0.3286	0.3474
P@6	0.3204	0.4948	0.4816	0.3112	0.3265
P@7	0.3006	0.4993	0.4898	0.2905	0.3021
P@8	0.2791	0.4636	0.4568	0.2723	0.2822
P@9	0.2627	0.2270	0.2214	0.2554	0.2647
P@10	0.2476	0.2307	0.2255	0.2403	0.2491
MAP	0.4775	0.4824	0.4775	0.4443	0.4696

其中,表3为各算法在MQ2008数据集上获得的在位置1-10上的NDCG@ p ;表4为各算法在MQ2008数据集上获得的在位置1-10上的Precision@ p 和MAP值。

从表3和表4的结果可以看出,本文所提方法可获得与目前最先进的排序方法相接近的结果,从实验上验证了所提方法的有效性。

结束语 为了去除特征的交叉和冗余,提出了一种基于矩阵分解的特征选择优化方法MFRank。该方法将排序学习中的偏序关系纳入特征构造中,使用优化方法对矩阵进行分解,并根据排序结果的影响对分解到的矩阵进行修正。实验中将所提方法与目前最先进的算法在MQ2008数据集上进行比较,所提方法的结果与目前发表的最先进的结果相接近,从而验证了其有效性。

参 考 文 献

- [1] HUANG Z H, ZHANG J W, TIAN C Q, et al. Survey on Learning-to-Rank Based Recommendation Algorithms [J]. Journal of Software, 2016, 27(3): 691-713. (in Chinese)
黄震华, 张佳雯, 田春岐, 等. 基于排序学习的推荐算法研究综述[J]. 软件学报, 2016, 27(3): 691-713.
- [2] GUO L, MA J, CHEN Z, et al. Learning to Recommend with Social Contextual Information from Implicit Feedback [J]. Soft Computing, 2015, 19(5): 1351-1362.
- [3] LIU L, LU X, LIAO Y, et al. Improving Retrieval of Plane Geometry Figure with Learning to Rank [J]. Pattern Recognition Letters, 2016, 83(3): 423-429.
- [4] <http://research.microsoft.com/en-us/projects/mslr/feature.aspx>.
- [5] PAN F, CONVERSE T, AHN D, et al. Greedy and Randomized Feature Selection for Web Search Ranking [C]// Proceedings of 11th IEEE International Conference on Computer and Information Technology. Sydney: IEEE Press, 2011: 436-442.
- [6] JÄRVELIN K, KEKÄLÄINEN J. Cumulated Gain-based Evaluation of IR Techniques [J]. ACM Transactions on Information Systems, 2002, 20(4): 422-446.
- [7] YATES R B, NETO B R. Modern Information Retrieval [M]. New York: Addison Wesley, 1999.
- [8] GENG X, LIU T, QIN T. Feature Selection for Ranking [C]// Sigir: International ACM SIGIR Conference on Research & Development in Information Retrieval. New York: ACM, 2007: 407-414.
- [9] DANG V, CROFT W B. Feature Selection for Document Ranking using Best First Search and Coordinate Ascent [C]// Proceedings of SIGIR Workshop Feature Generation and Selection of Information Retrieval, 2010: 1-5.
- [10] LI C, SHAO L, XU C S, et al. Feature selection under learning to rank model for multimedia retrieval [C]// Proceedings of 2nd International Conference of Internet Multimedia Computer Service, 2010: 69-72.
- [11] DA SILVA S R F, RIBEIRO M X, NETO J E S B, et al. Improving the Ranking Quality of Medical Image Retrieval using a Genetic Feature Selection Method [J]. Decision Support System, 2011, 5(4): 810-820.
- [12] LAI H, PAN Y, TANG Y, et al. FSMRank: Feature Selection Algorithm for Learning to Rank [J]. IEEE Transaction on Neural Networks and Learning Systems, 2013, 24(6): 940-952.
- [13] HUA G C, ZHANG M, KUANG D, et al. Feature Analysis Methods for Learning to Rank [J]. Computer Engineering and Applications, 2011, 47(17): 122-127. (in Chinese)
花贵春, 张敏, 邝达, 等. 面向排序学习的特征分析的研究[J]. 计算机工程与应用, 2011, 47(17): 122-127.
- [14] LIN Y. Research of Learning to Rank in Information Retrieval [D]. Dalian: Dalian University of Technology, 2012. (in Chinese)
林原. 信息检索中排序学习方法的研究[D]. 大连: 大连理工大学, 2012.
- [15] WU C G, LIANG Y Y, SUN Y F, et al. On the Equivalence of SVD and PCA [J]. Chinese Journal of Computers, 2004, 27(2): 286-288. (in Chinese)
吴春国, 梁艳芳, 孙延凤, 等. 关于 SVD 与 PCA 等价性的研究[J]. 计算机学报, 2004, 27(2): 286-288.
- [16] ZHANG C. Research on Matrix Factorization Based Collaborative Filtering Recommendation Algorithms [D]. Changchun: Jilin University, 2013. (in Chinese)
张川. 基于矩阵分解的协同过滤推荐算法研究[D]. 长春: 吉林大学, 2013.
- [17] HILLERMEIER C. Nonlinear Multiobjective Optimization [M]. Birkhäuser Verlag, Kluwer Academic Publishers, 2001.
- [18] RENDLE S, FREUDENTHALER C, GANTNER Z, et al. Bayesian Personalized Ranking from Implicit Feedback [C]// Proceedings of the Twenty-fifth Conference on Uncertainty in Artificial Intelligence. Arlington: AUAI Press, 2009: 452-461.
- [19] QIN T, LIU T, XU J, et al. LETOR: A Benchmark Collection for Research on Learning to Rank for Information Retrieval [J]. Information Retrieval Journal, 2010, 13(4): 346-374.
- [20] CAO Z, QIN T, LIU T, et al. Learning to Rank: From Pairwise Approach to Listwise Approach [C]// Proceedings of 24th International Conference of Machine Learning. New York: ACM, 2007: 129-136.
- [21] JOACHIMS T. Training Linear SVMs in Linear Time [C]// Proceedings of 12th ACM SIGKDD International Conference of Knowledge Discovery Data Mining. New York: ACM, 2006: 217-226.
- [22] FREUND Y, IYER R, SCHAPIRE R, et al. An Efficient Boosting Algorithm for Combining Preferences [J]. Journal of Machine Learning Research, 2003, 4(11): 933-969.
- [23] XU J, LI H. AdaRank: A Boosting Algorithm for Information Retrieval [C]// Proceedings of 30th Annual International ACM SIGIR Conference Research & Develop in Information Retrieval. New York: ACM, 2007: 391-398.