

基于张量分解的个性化标签推荐算法

李 贵 王 爽 李征宇 韩子扬 孙 平 孙焕良

(沈阳建筑大学信息控制工程学院 沈阳 110168)

摘 要 基于互联网的社会标签推荐系统为广大用户提供了一个信息共享平台,让用户以“标签”的形式为其浏览的物品标注信息。标签既描述了物品语义,又反映了用户偏好。标签系统的最大优势在于可以发挥群体的智能,获得用户对物品比较准确的关键词描述,而准确的标签信息是提升个性化推荐系统性能的重要资源。然而,现存的标签推荐系统面临的问题是:由于兴趣的不同,不同的用户对于同一物品可能会打不同的标签,或者是同一标签对于不同用户可能会蕴含不同的语义。因此如何有效获取用户、物品、标签三者之间潜在的语义关联成为标签推荐系统需要解决的主要问题。为此引入三维张量模型,利用三维张量的 3 个维度来分别描述社会标签推荐系统中 3 种类型的实体:用户、物品、标签。在基于历史标签数据(标签元数据)构建初始张量的基础上,应用高阶奇异值分解(HOSVD)方法降低张量维度,同时实现 3 种类型实体之间潜在的语义关联分析,从而进一步提高标签推荐系统的准确性。实验结果表明,该方法较当前两种典型的标签推荐算法(FolkRank 和 PR)在准确率和召回率性能指标上有明显提升。

关键词 社会标签,标签推荐,张量分解,高阶奇异值分解(HOSVD)

中图法分类号 TP301.6 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.2.056

Personalized Tag Recommendation Algorithm Based on Tensor Decomposition

LI Gui WANG Shuang LI Zheng-yu HAN Zi-yang SUN Ping SUN Huan-liang

(Faculty of Information & Control Engineering, Shenyang Jianzhu University, Shenyang 110168, China)

Abstract Internet-based social tag recommendation system provides a information sharing platform for the majority of users, allowing the users to annotate information for the items they have browsed in the form of “tag”. It not only describes the item’s semantics but also reflects the user’s preferences. The advantage of tag recommendation system is that the system can play a swarm intelligence to obtain the accurate keywords description of the item, and accurate tag information is an important resource to improve the performance of personalized recommendation system. However, due to the different interests of different users, the existing tag recommendation system is facing a problem that different users may tag different tags for the same items, and another problem is that the same tag for different users may contain different semantics. So how to effectively get the potential semantic association among the users, items and tags has become a main problem which needs to be solved. Therefore, we introduced a tensor model and used the third-order tensor to describe the three types of entities of social tag recommendation system: users, items and tags. On the basic of constructing initial tensor based on the history tagging data(tagging metadata), we applied the higher order singular value decomposition (HOSVD) method to reduce the dimension of tensor, at the same time to realize the analysis of potential semantic association between three types of entities. We performed experimental comparison of the method against two tag recommendations algorithms (FolkRank and PR) with two real data sets (Last, fm and Movielens). Experimental results show significant improvements of the method in terms of recall and precision.

Keywords Social tag, Tag recommendation, Tensor decomposition, Higher order singular value decomposition

1 引言

当前的 Last.fm、Movielens 和 YouTube 等大型网站都是应用社会标签推荐系统来实现物品的信息分类以及用户之间的信息分享,逐步实现标签归类并构建相应的用户群。然而

目前互联网上存在这样一个问题,即一些用户觉得键入标签非常乏味,这就造成了由于用户不愿意提供足够数量的标签而导致的标签系统的数据稀疏问题。

目前标签推荐系统还存在着词汇差异和语义模糊等问题。对于标签推荐系统而言,最基本的要求就是随着时间的

到稿日期:2014-03-06 返修日期:2014-05-11 本文受国家自然科学基金(61070024),辽宁省自然科学基金(2014020068)资助。

李 贵(1964—),男,博士,教授,主要研究方向为 Web 数据挖掘与信息集成、分布对象技术、软件工程, E-mail: Ligui21c@sina.com; 王 爽(1990—),女,硕士生,主要研究方向为 Web 数据挖掘和推荐系统; 李征宇(1980—),硕士,讲师,主要研究方向为 Web 数据挖掘和推荐系统; 韩子扬(1979—),硕士,讲师,主要研究方向为 Web 数据挖掘和推荐系统; 孙 平(1980—),硕士,副教授,主要研究方向为 Web 数据挖掘和推荐系统; 孙焕良(1969—),博士,教授,主要研究方向为数据仓库与数据挖掘技术、空间数据库、社会网络与社会计算等。

推移,新用户的不断增加,系统仍能快速地对用户进行准确的标签推荐。因此我们需要找出用户认为最恰当、最全面地诠释物品信息的一类标签,以便于用户更高效地查询、分享以及整合物品信息。

在一个图像标签推荐系统中,有这样一个实例,系统中的两个用户分别看到了以下两张图片,并在打标签时均接受了系统推荐的“苹果”标签,但是当其他用户在键入“苹果”标签进行图片检索时,这两张性质完全不同的图片却同时出现在图片推荐列表中,由于推荐系统不够完善而导致的语义模糊问题给用户带来了很大的不便和困扰。



图1 标签均为“苹果”的两张图片

基于以上原因,最近的研究都致力于挖掘用户在特定物品上所打标签(标签元数据)来完善标签推荐算法。传统的推荐系统一般使用协同过滤推荐算法(CF)^[2,6,7,11],该算法仅应用用户与物品之间的二维数据{用户,物品},因此传统的推荐系统捕获不到多种对象之间潜在的语义关联信息。为了缓解这一问题,文献[3]提出了一种通用算法,该算法将标签合并到标准的CF算法中构成了三维关联数据,并将三维关联数据{用户,物品,标签}转换成三组二维关联数据{用户,物品}、{用户,标签}、{物品,标签},这有助于挖掘3种类型实体之间潜在的语义关联。

本文将标签元数据进行三维分析,根据挖掘3种类型实体(用户、物品、标签)之间潜在的语义关联来提高标签推荐系统的准确性。首先我们需要解决以下两方面问题:

(1)用户、物品、标签3种类型实体之间的关系十分复杂,相同类型的实体之间存在相互关系({用户-用户}、{物品-物品}、{标签-标签}),不同类型的实体之间也存在相互关系({用户-物品}、{用户-标签}、{物品-标签}),因此个性化的标签推荐需要研究3种类型实体{用户-物品-标签}之间的三维关系。

(2)标签推荐系统中标签元数据的数据稀疏问题。

为此本文提出一个三维张量模型来存储三维的标签元数据,利用三维张量的3个维度分别描述用户、物品、标签这3种实体;并在三维张量上应用张量分解方法高阶奇异值分解HOSVD^[13]挖掘用户、物品、标签之间潜在的语义关联,并应用基于Kernel-SVD^[1,9]的平滑法解决数据稀疏问题。

2 相关工作

目前许多推荐系统都是应用CF推荐算法借助相似用户的喜好向用户推荐物品^[2]。CF算法应用的是用户与物品之间的二维关系。但由于社会标签推荐系统中用户、物品、标签之间是三维关系,因此不能直接应用CF算法。为了应用CF算法,文献[10]将三维关系降阶到二维关系,提出一种二维模型来描述用户、物品、标签之间的关联。

还有一类标签推荐算法则是基于与搜索引擎中应用的超链接相类似的概念结构,例如,协同标签推荐算法也被称为奖

惩算法(PR)^[18],算法中每个用户都对应一个权威分数,表示每个用户过去打标签的频率,且可以由一个与HITS^[12]算法相似的迭代算法计算得出。PR算法“奖励”推荐标签之间的高度相关性,“惩罚”推荐标签之间的概念重叠性,以达到物品在多个维度上较高的信息覆盖率。

另外一个典型的标签推荐算法是FolkRank^[8],该算法受PageRank^[14]算法启发,当许多网页都链接到同一网页时,随着网页的增多被链接网页的重要性逐步提升,前提是链接到该网页的“许多网页”本身的重要性也很高。该算法应用了社会标签推荐系统中的网页搜索与网页排名的核心思想,当一个重要用户在一个物品上打上一个重要标签时,这一物品就变成了重要物品,同理,标签和用户遵循的思想和物品一样。因此可以把三者之间的关系看成一个三部图,位于三部图顶点的3个节点(分别代表用户集、物品集、标签集)通过相互传递权重来加强自身的权重。

以上提到的所有算法都是将用户、物品、标签之间的三维关系转化为用户、物品、标签两两之间的相互关系,即二维关系{用户,物品}、{用户,标签}、{物品,标签},但是都丢失了三者之间的协同关系{用户、物品、标签}。相比之下,本文提出的方法引用了一个能同时描述三个维度的三维张量模型,3个维度分别代表用户、物品、标签,利用三维张量来存储标签元数据,并应用高阶奇异值分解方法(HOSVD)来挖掘用户、物品、标签三者之间潜在的语义关联。

HOSVD方法^[13]是以多元线性代数为基础的奇异值泛化解方法,用于解决多维数据的降维问题,并已成功应用于计算机视觉等领域,例如图像合成、视频排序以及数据集成等。另外,在个性化的网页搜索领域中,随着网页搜索的竞争越发激烈,互联网对于个性化的网页搜索要求也越来越高。文献[15]提出CubeSVD理念,并借助HOSVD方法抽取MSN搜索引擎中的数据进行实验验证,实验结果表明该方法成功地提升了网页搜索能力。随后文献[4]提出应用HOSVD方法来实现物品推荐,但它并没有将此方法应用到标签推荐系统。本文则重点研究如何应用HOSVD方法来解决标签推荐系统中由历史标签数据构建的三维张量的降维问题。

3 张量分解

本节简单介绍张量和高阶奇异值分解方法,符号表示定义如下:

张量:(A, B, \dots); 矩阵:(A, B, \dots);

向量:(a, b, \dots); 标量:(a, b, \dots)。

3.1 张量的矩阵展开与乘积

张量也称多维矩阵,张量的维也称张量的模,向量和矩阵可以分别看作是一阶张量和二阶张量。一个 N 阶张量 A 被定义为 $A \in \mathbb{R}^{I_1 \times \dots \times I_N}$,张量中的元素表示为 a_{i_1, \dots, i_N} ,由于本文侧重于研究标签推荐系统中的三大实体用户、物品、标签,因此,本文仅针对三维张量 $A \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ 进行分析。

举一个张量 $A \in \mathbb{R}^{3 \times 2 \times 3}$ 的实例,张量 A 中的元素值如下:

$$a_{111} = a_{221} = a_{321} = a_{312} = a_{222} = a_{313} = -a_{311} = 1; a_{121} = a_{212} = a_{113} = 0; a_{211} = a_{213} = a_{112} = a_{122} = a_{323} = -a_{322} = 2; a_{123} = a_{223} = 4。$$

上述实例中,张量 $A \in \mathbb{R}^{3 \times 2 \times 3}$ 的构成图如图2所示。

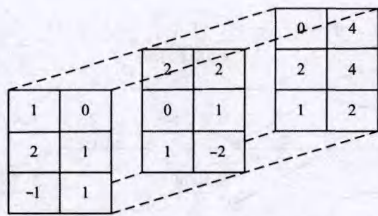


图2 实例中张量 $A \in R^{3 \times 2 \times 3}$ 的构成图

3.1.1 张量的矩阵展开

为了在三维张量 A 上应用高阶奇异值分解技术,并简化张量与矩阵的乘积,首先需要将张量进行矩阵展开,也就是将张量按照不同的维度(n -模)重新排列成一个矩阵,我们将这一过程称作张量的 n -模矩阵展开。一个张量 A 的 n -模矩阵展开是指将 A 的 n -模元素排列成一个矩阵的列向量从而得到一个矩阵,记做 A_n 。由于本文只研究三维张量,因此将张量 A 的 1-模、2-模、3-模矩阵展开式 A_1, A_2, A_3 分别定义如下^[13]:

$$A_1 \in R^{I_1 \times I_2 I_3}; a_{i_1 i_2 i_3} = a_{i_1}^{(1)}, v = i_2 + (i_3 - 1)I_2 \quad (1)$$

$$A_2 \in R^{I_2 \times I_1 I_3}; a_{i_1 i_2 i_3} = a_{i_2}^{(2)}, v = i_3 + (i_1 - 1)I_1 \quad (2)$$

$$A_3 \in R^{I_1 I_2 \times I_3}; a_{i_1 i_2 i_3} = a_{i_3}^{(3)}, v = i_1 + (i_2 - 1)I_2 \quad (3)$$

根据式(1)~式(3)即可得出 $A_i (i \in [1, 3])$ 与张量 A 中元素的对应关系,从而得到矩阵展开式 A_i 。三维张量的矩阵展开图如图3所示。

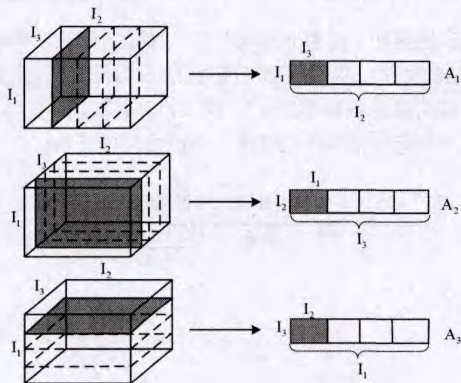


图3 三维张量的矩阵展开图

根据上述定义即可得出图2中张量 $A \in R^{3 \times 2 \times 3}$ 在 1-模、2-模、3-模下的矩阵展开式 A_1, A_2, A_3 , 如下所示:

$$A_1 = \begin{bmatrix} 1 & 0 & 2 & 2 & 0 & 4 \\ 2 & 1 & 0 & 1 & 2 & 4 \\ -1 & 1 & 1 & -2 & 1 & 2 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 1 & 2 & 0 & 2 & 0 & 2 & -1 & 1 & 1 \\ 0 & 2 & 4 & 1 & 1 & 4 & 1 & -2 & 2 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 1 & 2 & 0 & 0 & 2 & 4 \\ 2 & 0 & 2 & 1 & 1 & 4 \\ -1 & 1 & 1 & 1 & -2 & 2 \end{bmatrix}$$

3.1.2 矩阵与张量的 n -模乘积

设矩阵 $U \in R^{J_n \times I_n}$, 张量 $A \in R^{I_1 \times \dots \times I_N}$, 将矩阵 U 与张量 A 的 n -模乘积定义为 $A \times_n U$;

$$(A \times_n U)_{i_1 i_2 \dots i_{n-1} i_{n+1} \dots i_N} = \sum_{i_n=1}^{I_n} a_{i_1 i_2 \dots i_{n-1} i_n i_{n+1} \dots i_N} u_{i_n} \quad (4)$$

例如, 矩阵 $U \in R^{I_1 \times I_1}$, 张量 $A \in R^{I_1 \times I_2 \times I_3}$ 的 1-模乘积为

$$B = A \times_1 U \Leftrightarrow \sum_{i_1=1}^{I_1} a_{i_1 i_2 i_3} u_{i_1} = B(i_2, i_3)$$

其中, $B \in R^{I_2 \times I_3}$ 。

由于 A_n 的列向量是张量 A 的 n -模, 因此矩阵 U 与张量 A 的 n -模乘积可以理解为先进行普通的矩阵相乘 $B_n = UA_n$, 再将 B_n 重新排列成张量 B 。

3.2 奇异值分解

奇异值分解 (singular value decomposition, SVD) 是线性代数中一种重要的矩阵分解方法, 例如一个 $I_1 \times I_2$ 阶矩阵 F 可以表示成 3 个矩阵的乘积, 如式(5)所示。

$$F_{I_1 \times I_2} = U_{I_1 \times I_1} \cdot S_{I_1 \times I_2} \cdot V_{I_2 \times I_2}^T \quad (5)$$

其中, U 表示矩阵 $F_{I_1 \times I_2}$ 的左奇异矩阵, V^T 表示矩阵 $F_{I_1 \times I_2}$ 的右奇异矩阵的转置矩阵, S 表示由矩阵 $F_{I_1 \times I_2}$ 奇异值组成的对角矩阵, 矩阵奇异值分解的可视图如图4所示。

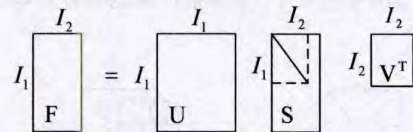


图4 矩阵 F 的奇异值分解可视图

通过将矩阵 F 进行奇异值分解, 得到由矩阵 F 的奇异值组成的对角矩阵 S , 并且该对角矩阵 S 满足以下两个特点:

(1) 对角矩阵 $S = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_{\min\{I_1, I_2\}})$

(2) 满足递减性 $(\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{\min\{I_1, I_2\}} \geq 0)$

再通过对矩阵 F 的奇异值进行删减(保留前 c 个较大的奇异值, 且满足 $\text{largest } c < \min\{I_1, I_2\}$), 其中 c 值的大小取决于保留原矩阵信息百分比的值)来求解矩阵 F 的近似矩阵, 进而反映矩阵 F 的主要趋势, 这一方法被应用于隐语义分析领域中用于挖掘潜在的语义关联。

根据矩阵与张量 n -模乘积的定义, 并把矩阵看作是二维张量, 可以将矩阵的奇异值分解 SVD 表示成二维张量 S 与矩阵 $U^{(1)}$ 的 1-模以及矩阵 $U^{(2)}$ 的 2-模乘积的形式, 如下所示:

$$F = S \times_1 U^{(1)} \times_2 U^{(2)} \quad (6)$$

其中 $U^{(1)}, U^{(2)}$ 分别满足以下两式:

$$F_1 = U^{(1)} \cdot S \cdot V_1^T \quad (7)$$

$$F_2 = U^{(2)} \cdot S \cdot V_2^T \quad (8)$$

F_1, F_2 分别为矩阵 $F_{I_1 \times I_2}$ 及其转置矩阵 $F_{I_2 \times I_1}$, $U^{(1)}, U^{(2)}$ 均为正交矩阵。其中 S 是由矩阵 F 的奇异值 σ 组成的对角矩阵。

3.3 高阶奇异值分解

HOSVD(Higher Order Singular Value Decomposition) 是奇异值分解(SVD)在张量中的推广, 张量的 HOSVD 是指将张量分解成一个与其大小相同的核心张量与多个矩阵的乘积形式。由于标签推荐系统的需要, 本文只需将其应用于三维张量, 因此将三维张量 A 的 HOSVD 定义如下^[13]:

$$A = S \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)} \quad (9)$$

核心张量 $S \in R^{I_1 \times I_2 \times I_3}$ 是一个与张量 A 维数相同的张量, 具有正交的所有性质, 协调各正交矩阵 $U^{(1)}, U^{(2)}, U^{(3)}$ 之间的相互作用, 其中 $U^{(1)}, U^{(2)}, U^{(3)}$ 分别满足以下 3 式:

$$A_1 = U^{(1)} \cdot S_1 \cdot V_1^T \quad (10)$$

$$A_2 = U^{(2)} \cdot S_2 \cdot V_2^T \quad (11)$$

$$A_3 = U^{(3)} \cdot S_3 \cdot V_3^T \quad (12)$$

由于用户、物品、标签各自及其相互之间都存在相关性，因此由标签元数据 $\{u, i, t\}$ 构成的三维张量 A 具有低秩特性。一般情况下，张量数据中含有噪声，因此三维张量 A 并不是低秩的，我们可以通过应用HOSVD方法求得一个低秩的近似张量 \hat{A} 来近似表示张量 A ，数学表达式如下：

$$A = S \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)}$$

$$\hat{S} = A \times_1 U_{c_1}^{(1)T} \times_2 U_{c_2}^{(2)T} \times_3 U_{c_3}^{(3)T} \quad (13)$$

$$\hat{A} \approx \hat{S} \times_1 U_{c_1}^{(1)} \times_2 U_{c_2}^{(2)} \times_3 U_{c_3}^{(3)} \quad (14)$$

其中， $U_{c_i}^{(i)}$ 为 $U^{(i)}$ ($i \in [1, 3]$)的近似矩阵，张量 \hat{S} 为核心张量 S 的近似张量，张量 \hat{A} 为张量 A 的近似张量。通常情况下，可以认为噪声远小于张量的核心数据，因此可以利用式(14)来计算张量 A 的近似张量 \hat{A} 。三维张量 A 的高阶奇异值分解如图5所示。

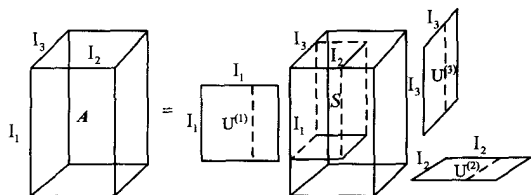


图5 三维张量 A 的高阶奇异值分解

4 基于张量分解的标签推荐算法

本文提出的方法以张量分解为基础，以下小节将通过实例来概括方法的整体思路及方法步骤。

4.1 算法轮廓

社会标签推荐系统将用户给物品打标签的所有标签元数据用三元组集合 $\{u, i, t\}$ 的形式表示，并以张量为模型来呈现这些三元组数据 $\{u, i, t\}$ 。张量分解方法就是在三维张量上应用HOSVD方法，这既能降低张量的维度，又能实现3种类型实体之间潜在的语义分析。概括的说，就是输入由标签元数据 $\{u, i, t\}$ 构造的初始张量 A ，通过应用HOSVD方法，输出重建的近似张量 \hat{A} 。

张量 \hat{A} 预测用户、物品、标签三者之间的协同关联。 \hat{A} 中的元素由一个四元组 $\{u, i, t, p\}$ 表示。其中 p 表示用户 u 给物品 i 打标签 t 的概率(也称关联权重)。因此，标签 t 是否被推荐给用户 u 由三元组 $\{u, i, t\}$ 之间的关联权重 p 决定。

为了解释如何实现张量降维以及挖掘3种类型实体之间潜在的语义关联，我们通过实例来进行说明，如图6所示。有4个用户分别给4个不同的物品打了标签，图6中，用户与物品之间的箭头代表用户给相应的物品打过标签，物品和标签之间的箭头表示该物品被用户打过相应的标签，箭头上注释的数字表示用户、物品、标签三者之间的关系链。该实例包含6条关系链，数字1表示用户 U_1 给物品 I_1 打过标签 T_1 （“电视机”），数字2表示用户 U_2 给物品 I_2 打过标签 T_2 （“数码相机”），数字3、4分别表示用户 U_2, U_3 给物品 I_3 打过标签 T_3 （“机械相机”），数字5、6分别表示用户 U_3, U_4 给物品 I_4 打过标签 T_4 （“摄像机”）。

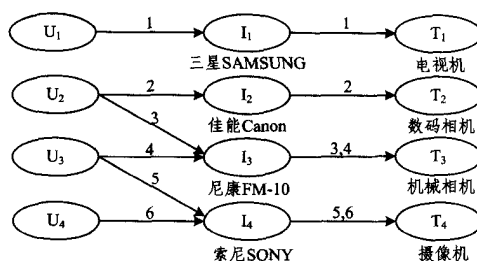


图6 标签元数据的输入模拟图

从图6可以看出，用户 U_2, U_3, U_4 对摄影器材感兴趣，不同于用户 U_2, U_3, U_4 ，用户 U_1 是对电视机感兴趣。下面通过标签元数据来构造一个三维张量 $A \in \mathbb{R}^{4 \times 4 \times 4}$ ，并且把三元组中的用户、物品、标签同时出现的概率(也称关联权重)作为张量 A 的元素，所有的关联权重都被初始化为1，如表1所列。

表1 通过实例中的标签元数据构建的初始张量 A

	用户	物品	标签	关联权重
1	U_1	I_1	T_1	1
2	U_2	I_2	T_2	1
3	U_2	I_3	T_3	1
4	U_3	I_3	T_3	1
5	U_3	I_4	T_4	1
6	U_4	I_4	T_4	1

应用张量分解算法后得到重建的近似张量 \hat{A} ，如表2和图7所示。表2描述了近似张量 \hat{A} 中元素(关联权重)的值，图7形象地描述了近似张量 \hat{A} 中用户、物品、标签之间的关系(关联权重省略)。应用张量分解算法后明显看到，系统的输出新生成两条关联数据链 $\{U_3, I_2, T_2\}$ 和 $\{U_4, I_3, T_3\}$ ，如表2中第7、8两行加粗字体以及图7中虚线位置所示。

表2 通过实例中的标签元数据重建的近似张量 \hat{A}

	用户	物品	标签	关联权重
1	U_1	I_1	T_1	1
2	U_2	I_2	T_2	0.89
3	U_2	I_3	T_3	1.08
4	U_3	I_3	T_3	0.85
5	U_3	I_4	T_4	1.09
6	U_4	I_4	T_4	0.90
7	U_3	I_2	T_2	0.19
8	U_4	I_3	T_3	0.20

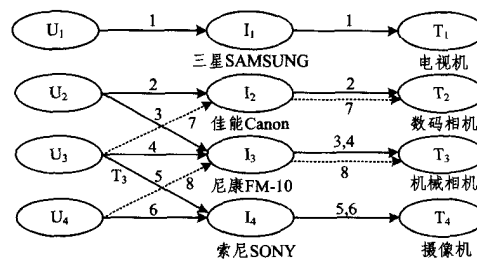


图7 基于张量分解算法的输出模拟图

根据新生成的关联数据，当用户 U_3 给物品 I_2 打标签时，系统会自动给用户 U_3 推荐标签 T_2 ；当用户 U_4 给物品 I_3 打标签时，系统会自动给用户 U_4 推荐标签 T_3 ，由于在原来的张量 A 中并没有用户 U_3 和物品 I_2 之间的关联数据，也没有用户 U_4 和物品 I_3 之间的关联数据，这将直接影响推荐系统的推荐准确性。然而表2中的数据明确显示，近似张量 \hat{A} 中三元组 $\{U_3, I_2, T_2\}$ 的关联权重为0.19，三元组 $\{U_4, I_3, T_3\}$ 的关

联权重为 0.2, 这表明系统应用张量分解算法后成功地挖掘了用户、物品、标签之间的潜在关联, 并预测用户 U_3 在物品 I_2 上打标签 T_2 的关联权重为 0.19, 用户 U_4 在物品 I_3 上打标签 T_3 的关联权重为 0.2, 而且数据中并没有用户 U_3 对物品 I_2 以及用户 U_4 对物品 I_3 打过其他标签的记录。因此, 系统直接向用户 U_3 推荐标签 T_2 给物品 I_2 , 向用户 U_4 推荐标签 T_3 给物品 I_3 。

由此看出推荐的结果很合理, 因为用户 U_3 、 U_4 都是对摄影方面的器材感兴趣。结果证明张量分解算法能够成功地挖掘多种类型实体(用户、物品、标签)之间潜在的语义关联, 从而提高标签推荐系统的性能。

4.2 张量分解算法

张量分解算法首先需要基于用户、物品、标签的标签元数据三元组 $\{u, i, t\}$ 来构造一个初始张量 A , 它承载着社会标签推荐系统中用户、物品、标签三者之间所有的关联数据; 其次应用 HOSVD 方法将张量 A 进行 n -模矩阵展开, 生成 3 个新的矩阵 A_1 、 A_2 、 A_3 , 然后在矩阵 A_1 、 A_2 、 A_3 上分别进行二维矩阵奇异值分解(SVD)计算; 最后构建新的核心张量并计算近似张量 \hat{A} 。方法计算步骤总结如下:

第 1 步 基于标签元数据三元组 $\{u, i, t\}$ 构造初始张量 A 。

第 2 步 应用 HOSVD 方法将张量 A 在 3 个模式(1-模、2-模、3-模)下按照式(1)一式(3)分别进行 n -模矩阵展开, 生成 3 个矩阵展开式 A_1 、 A_2 、 A_3 。

第 3 步 在矩阵展开式 A_1 、 A_2 、 A_3 上按照式(5)分别进行二维矩阵 SVD 计算, 得到 3 个对应的左奇异矩阵 $U^{(1)}$ 、 $U^{(2)}$ 、 $U^{(3)}$ 。

第 4 步 根据低秩近似计算过滤掉由较小的奇异值引起的噪声数据, 通过得到的左奇异矩阵 $U^{(1)}$ 、 $U^{(2)}$ 、 $U^{(3)}$, 求解其降秩后的维度参数 c_1 、 c_2 、 c_3 (求解方法见 4.2.4 节), 再根据维度参数求解各左奇异矩阵的近似矩阵 $U_{c_1}^{(1)}$ 、 $U_{c_2}^{(2)}$ 、 $U_{c_3}^{(3)}$ 。

第 5 步 根据求解的近似矩阵 $U_{c_1}^{(1)}$ 、 $U_{c_2}^{(2)}$ 、 $U_{c_3}^{(3)}$, 按照式(13)构造核心张量 S 的近似张量 \hat{S} 。

第 6 步 根据求得的近似张量 \hat{S} , 按照式(14)计算张量 A 的近似张量 \hat{A} 。

第 7 步 根据重建的近似张量 \hat{A} 中元素的关联权重, 向目标用户推荐标签。

前 6 步是建立模型的过程, 均需离线实现; 第 7 步则需在线才能实现。以下小节将详细描述算法每一步的细节。

4.2.1 构造初始张量 A

基于标签元数据三元组 $\{u, i, t\}$ 构造初始张量 $A \in R^{I_u \times I_i \times I_t}$, 其中 I_u 、 I_i 、 I_t 分别表示标签元数据中用户、物品、标签的数量。张量中的元素代表用户 u 给物品 i 打标签 t 的概率(三元组 $\{u, i, t\}$ 之间的关联权重)。实例中由标签元数据构造的初始张量 $A \in R^{4 \times 4 \times 4}$, 如图 8 所示。

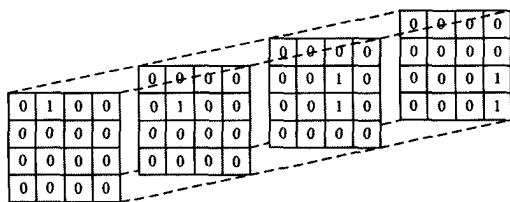


图 8 实例中的初始张量 A

4.2.2 张量 A 的矩阵展开

根据式(1)一式(3)将图 8 中构建的初始张量 A 展开成 3 个矩阵 A_1 、 A_2 、 A_3 。矩阵展开式如下所示:

$A_1 \in R^{I_u \times I_i \times I_t}$ (1-模); $A_2 \in R^{I_i \times I_u \times I_t}$ (2-模); $A_3 \in R^{I_u \times I_i \times I_t}$ (3-模)。

$$A_1 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

4.2.3 在 3 个模式上分别应用 SVD

在 3 个矩阵展开式 A_1 、 A_2 、 A_3 上按照式(10)一式(12)分别进行二维矩阵 SVD 计算, 得到左奇异矩阵 $U^{(1)}$ 、 $U^{(2)}$ 、 $U^{(3)}$ 及其对应的奇异值对角矩阵 S_1 、 S_2 、 S_3 。

$$A_1 = U^{(1)} \cdot S_1 \cdot V_1^T, A_2 = U^{(2)} \cdot S_2 \cdot V_2^T$$

$$A_3 = U^{(3)} \cdot S_3 \cdot V_3^T$$

由于核心张量 \hat{S} 的维度最终是由左奇异矩阵的维度参数 c_i ($i \in [1, 3]$) 决定的, 因此首先需要通过初始的对角矩阵 S_1 、 S_2 、 S_3 做适当调整, 过滤掉原始矩阵中稀疏的噪声数据(默认为保留原始矩阵信息的 80%)来确定 3 个模式(1-模、2-模、3-模)下左奇异矩阵的维度参数 c_1 、 c_2 、 c_3 。图 9—图 11 表示将矩阵展开式 A_1 、 A_2 、 A_3 分别进行二维矩阵 SVD 计算得到的左奇异矩阵及其对应的奇异值。

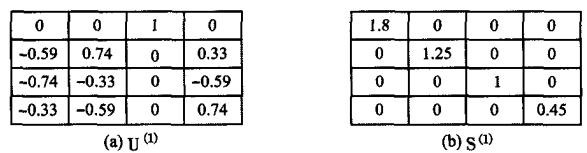


图 9

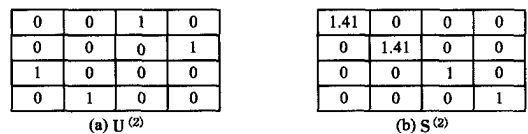


图 10

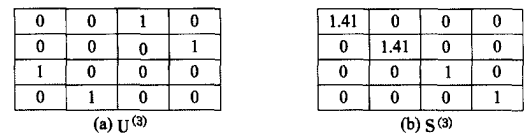


图 11

4.2.4 低秩近似计算

在矩阵降秩过程中, 低秩近似计算通常用于过滤由较小的奇异值引起的噪声。因此, 可以在二维矩阵 SVD 计算过程中将对角矩阵 $S^{(1)}$ 、 $S^{(2)}$ 、 $S^{(3)}$ 中较小的奇异值以及与其对应的左奇异矩阵中的数据全部删除, 保留秩为 c_i ($i \in [1, 3]$) 的奇异值以及与其对应的左奇异矩阵, 其中 $U_{c_i}^{(i)}$ 表示秩为 c_i 的

左奇异矩阵 $U^{(i)} (i \in [1, 3])$ 的降秩矩阵。由此得到的矩阵被定义为 c -秩近似矩阵 $U_c^{(i)} (i \in [1, 3])$ ，其中 c_i 的值取决于保留初始对角矩阵 $S^{(1)}$ 、 $S^{(2)}$ 、 $S^{(3)}$ 数据的百分比。

实验结果表明，保留初始对角矩阵 $S^{(1)}$ 、 $S^{(2)}$ 、 $S^{(3)}$ 数据的 90% 能够得到最好的近似值。通过将矩阵 S 中的奇异值求和并根据具体实例决定保留对角矩阵 $S^{(i)} (i \in [1, 3])$ 中多少比例的奇异值从而得到理想的百分比。以实例中的对角矩阵 $S^{(1)}$ 为例，计算其奇异值之和为 4.5 (即 $1.8 + 1.25 + 1 + 0.45$)，因此我们保留初始对角矩阵 $S^{(1)}$ 90% (即 $4.05/4.5$) 的数据，将 c_1 设定为 3，同理将 c_2 、 c_3 设定为 4。该方法既简单又高效，曾被应用于很多相关张量分解的工作^[1,7] 当中，因此本文也采用了该方法。图 12 为矩阵 $U^{(i)} (i \in [1, 3])$ 进行低秩近似计算后的转置矩阵。

0	-0.59	-0.74	-0.33
0	0.74	-0.33	-0.59
1	0	0	0

($U_{c_1}^{(1)}$)^T

0	0	1	0
0	0	0	1
1	0	0	0
0	1	0	0

($U_{c_2}^{(2)}$)^T

0	0	1	0
0	0	0	1
1	0	0	0
0	1	0	0

($U_{c_3}^{(3)}$)^T

图 12 $U^{(i)} (i \in [1, 3])$ 降秩后的转置矩阵

4.2.5 近似核心张量 \hat{S} 的构造

由于核心张量控制用户、物品、标签之间的相互作用，因此我们需要根据降秩后的矩阵 $U_{c_1}^{(1)}$ 、 $U_{c_2}^{(2)}$ 、 $U_{c_3}^{(3)}$ ，并依据式 (13) 来构造近似核心张量 \hat{S} 。

$$\hat{S} = A \times_1 U_{c_1}^{(1)T} \times_2 U_{c_2}^{(2)T} \times_3 U_{c_3}^{(3)T}$$

其中， A 为实例中的初始张量 (见图 9)， $U_{c_1}^{(1)T}$ 、 $U_{c_2}^{(2)T}$ 、 $U_{c_3}^{(3)T}$ 分别为维度参数为 c_1 、 c_2 、 c_3 的左奇异矩阵 $U^{(1)}$ 、 $U^{(2)}$ 、 $U^{(3)}$ 的转置矩阵。从式 (13) 可以看出， c_1 、 c_2 、 c_3 的选值将最终决定核心张量 \hat{S} 的维度系数。图 13 为重建的核心张量 \hat{S} 。

图 13 重建的核心张量 \hat{S}

4.2.6 计算近似张量 \hat{A}

最后，近似张量 \hat{A} 是由重建的核心张量 \hat{S} 以及 3 个降秩的左奇异矩阵 $U_{c_1}^{(1)}$ 、 $U_{c_2}^{(2)}$ 、 $U_{c_3}^{(3)}$ ，并根据式 (14) 计算求得的。

$$\hat{A} = \hat{S} \times_1 U_{c_1}^{(1)} \times_2 U_{c_2}^{(2)} \times_3 U_{c_3}^{(3)}$$

其中， \hat{S} 为重建的核心张量， $U_{c_1}^{(1)}$ 、 $U_{c_2}^{(2)}$ 、 $U_{c_3}^{(3)}$ 分别为维度参数为 c_1 、 c_2 、 c_3 的左奇异矩阵 $U^{(1)}$ 、 $U^{(2)}$ 、 $U^{(3)}$ 的降秩矩阵。针对本实例，得到的近似张量 \hat{A} 如图 14 所示。

图 14 实例中的近似张量 \hat{A}

4.2.7 标签推荐列表的生成

重建的近似张量 \hat{A} 反映用户、物品、标签之间的关联，张量 \hat{A} 的元素均用一个四元组 $\{u, i, t, p\}$ 表示，其中 p 表示用户 u 、物品 i 以及标签 t 之间的关联权重。

在此基础上，标签 t 是否会被推荐给用户 u 打在物品 i 上 是根据标签 t 与二元组 $\{u, i\}$ 之间的关联权重决定的。如果要向用户 u 推荐关于物品 i 的 N 个标签，我们只需选择权重最高的前 N 个关联标签即可。然而在图 14 中我们可以看到张量 \hat{A} 中的元素四元组为 $\{U_3, I_2, T_2, 0.19\}$ 和 $\{U_4, I_3, T_3, 0.2\}$ ，这表示用户 U_3 、物品 I_2 、标签 T_2 之间的关联权重为 0.19，用户 U_4 、物品 I_3 、标签 T_3 之间的关联权重为 0.2，也就是用户 U_3 给物品 I_2 打标签 T_2 的概率为 0.19，用户 U_4 给物品 I_3 打标签 T_3 的概率为 0.2。鉴于该实例的标签元数据，对于用户 U_3 并没有其他标注在物品 I_2 上的相关标签记录，用户 U_4 在物品 I_3 上也没有相关的标签记录。因此，基于物品 I_2 我们向用户 U_3 推荐标签 T_2 ，基于物品 I_3 我们向用户 U_4 推荐标签 T_3 ，类似的方法也可以应用到用户推荐和物品推荐。

4.2.8 实现张量分解的实验环境

通过应用张量分解算法来实现张量降维和多种类型实体之间潜在的语义分析，使得标签推荐更为准确需要两个环节：离线环节和在线环节。离线环节解决的是后台 HOSVD 的计算，在线环节解决的是从重建的近似张量 \hat{A} 中检索出新的关联权重从而进行实时推荐。在实际的应用系统中，后者能够通过实时推荐影响用户的实际行为。

对于离线环节，在实际的应用系统中 (标签元数据中包含大量的用户、物品、标签数据)，其主要的特征就是数据稀疏，意味着张量的大部分元素都是 0。因此，HOSVD 的计算问题归结为以下两点：

- (1) 计算在 3 个模式下的矩阵展开式 A_1 、 A_2 、 A_3 的奇异值和左奇异矩阵 (这些矩阵规模较大且数据稀疏)。
- (2) 计算初始张量与进行低秩近似计算后得到的 3 个降秩矩阵的转置矩阵 $U_{c_1}^{(1)T}$ 、 $U_{c_2}^{(2)T}$ 、 $U_{c_3}^{(3)T}$ 的乘积来求取新的核心张量 \hat{S} 。

5 实验结果

本节将本文提出的张量分解算法 (定义为 Tensor Reduction) 与两个经典的标签推荐算法 FolkRank^[8] 和协同标签推荐算法^[5] (定义为 PR) 进行实验比较，得到各算法准确率和召回率的实验数据。

5.1 数据集

为了评估算法性能，我们从两个不同的社会标签推荐系统中选择两个真实的数据集 MovieLens 和 Last.fm，并应用两个数据集的核心子集进行实验验证。

MovieLens: 数据集中三元组 $\{u, i, t\}$ 分别代表用户、电影以及公开的标签，数据集中用户、物品、标签的数量分别为 1104、28599 和 87096。

Last.fm: 数据集中三元组 $\{u, i, t\}$ 分别代表用户、艺术家以及公开的标签，数据集中用户、物品、标签的数量分别为 4561、1872 和 2895。

从数据集 MovieLens 和 Last.fm 中提取的两个核心子集

中用户、物品、标签的数量如表 3 所列。

表 3 数据集 Movielens 和 Last. fm 的核心子集大小

数据集	用户数	物品数	标签数
Movielens	110	257	600
Last. fm	114	248	559

5.2 实验设置

传统的经典评估指标^[17]中,准确率描述的是最终的推荐列表中有多少比例的标签推荐是发生过的,召回率描述的是有多少比例的标签推荐包含在最终的推荐列表中。本文在此基础上将传统的评估指标做一些调整,将每个测试用户所打标签分为两组,其中 1 组和 2 组的测试集大小默认为各占测试用户所打标签数量的 50%。

1 组:测试用户以前打过的标签(定义为 $T_1(u)$);

2 组:预测测试用户未来要打的标签(定义为 $T_2(u)$)。

令 $R(u)$ 是根据用户在训练集上的行为给用户做出的 top-N 推荐列表, $T(u)$ 为系统向用户推荐标签后用户实际所打标签。评估指标准确率和召回率的定义如下:

$$precision = R(u) \cap T_2(u) / N$$

$$recall = R(u) \cap T(u) / T_2(u) \cap T(u)$$

5.3 实验结果

本节以准确率和召回率为评估指标将张量分解算法与 FolkRank 和 PR 算法进行实验比较,每个算法的鲁棒性都是以尽可能小地降低准确率为代价来获取更高的召回率。

基于数据集 Movielens 和 Last. fm ($N \in [1, 5]$), 将 3 种算法的准确率和召回率绘制成曲线,如图 15、图 16 所示。图中各有 3 条曲线,每条曲线上有 5 个节点,每个节点代表一个标签,节点位置从左至右代表该标签在 top-N 列表中的排名,依次为 1 到 5。从图中可以看出,每种算法的准确率都会随着 N 的上升而下降;相反,随着 N 值的增大,3 种算法的召回率都有所提升。根据推荐列表中的标签排名,张量分解算法的准确率和召回率均高于算法 FolkRank 和 PR,实验结果表明张量分解算法更具鲁棒性。

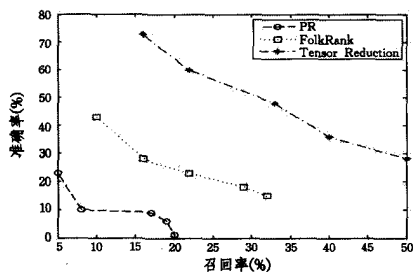


图 15 基于 Movielens 数据集的实验数据比较

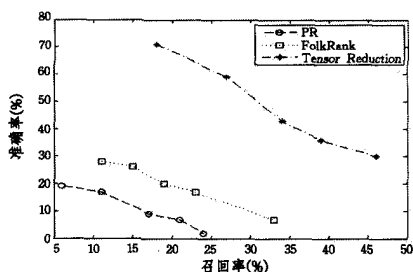


图 16 基于 Last. fm 数据集的实验数据比较

从图 15、图 16 中明显看出算法的数据轨迹极其相似。实验结果表明,在两个不同数据集中张量分解算法较 FolkRank 和 PR 算法提供的标签推荐精度更高,更具鲁棒性。

结束语 协同标签推荐算法向用户推荐标签是基于收集其他用户对于同一物品打了哪些标签,旨在挖掘哪些标签能够更好地描述该物品。本文则引入一个三维张量模型来描述存在于社会标签推荐系统中的 3 种类型实体用户、物品、标签。利用三维张量来存储系统中 3 种类型实体的元数据,并应用 HOSVD 方法将三维张量降维来挖掘 3 种类型实体之间潜在的语义关联。将张量分解算法同两个经典的标签推荐算法 FolkRank 和 PR 进行了实验比较,应用两个真实的数据集 Movielens 和 Last. fm 作为实验平台,结果表明基于张量分解的标签推荐算法在准确率和召回率方面都有大幅度提升。

参考文献

- [1] Cristianini N, Shawe-Taylor J. Kernel Methods for Pattern Analysis[M]. Cambridge University Press, 2004
- [2] Breese J, Heckerman D, Kadie C. Empirical analysis of predictive algorithms for collaborative filtering[C]//Proc. Conf. on Uncertainty in Artificial Intelligence. 1998:43-52
- [3] Tso-Sutter K, Marinho B, Schmidt-Thieme L. Tag-Aware Recommender Systems by Fusion of Collaborative Filtering Algorithms[C]//Proc. ACM Symp. Applied Computing (SAC). 2008
- [4] Xu Y, Zhang L, Liu W. Cubic analysis of social bookmarking for personalized recommendation[C]//Frontiers of WWW Research and Development - APWeb. 2006:733-738
- [5] Xu Z, Fu Y, Mao J, et al. Towards the semantic Web; Collaborative tag suggestions[C]//Proceedings of Collaborative Web Tagging Workshop at the 15 WWW Conference. 2006
- [6] Herlocker J, Konstan J, Riedl J. An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms [J]. Information Retrieval, 2002, 5(4): 287-310
- [7] Sun J, Tao D, Faloutsos C. Beyond Streams and Graphs: Dynamic Tensor Analysis[C]//Proc. ACM Conf. Knowledge Discovery and Data Mining (KDD). 2006:374-383
- [8] Hotho A, Jaschke R, Schmitz C, et al. Information retrieval in folksonomies; Search and ranking[C]//The Semantic Web; Research and Applications. 2006:411-426
- [9] Chin T, Schindler K, Suter D. Incremental Kernel SVD for Face Recognition with Image Sets[C]//Proc. Int'l Conf. Automatic Face and Gesture Recognition (FGR). 2006:461-466
- [10] Jaschke R, Marinho L, Hotho A, et al. Tag recommendations in folksonomies[C]//Knowledge Discovery in Databases (PKDD). 2007:506-514
- [11] Karypis G. Evaluation of item-based top-n recommendation algorithms[C]//Proc. ACM CIKM Conf. . 2001:247-254
- [12] Kleinberg J. Authoritative sources in a hyper linked environment [J]. Journal of the ACM, 1999, 46(5): 604-632
- [13] Lathauwer L D, Moor B D, Vandewalle J. A multilinear singular value decomposition[J]. SIAM Journal of Matrix Analysis and Applications, 2000, 21(4): 1253-1278
- [14] Page L, Brin S, Motwani R, et al. The pagerank citation ranking; bringing order to the Web[R]. Technical Report, 1998
- [15] Sun J, Shen D, Zeng H, et al. Cubesvd: a novel approach to personalized web search[C]//World Wide Web Conference. 2005: 382-390
- [16] Wang H, Ahuja N. A tensor approximation approach to dimensionality reduction [J]. International Journal of Computer Vision. , 2008, 76: 217-229
- [17] 项亮, 陈义, 王益. 推荐系统实践[M]. 河北: 人民邮电出版社, 2012: 39-43