

PMUS-HOSGD 张量分解方法及其在标签推荐中的应用

杨 林^{a,b}, 顾军华^{a,b}, 官 磊^{a,b}, 张宇娟^{a,b}, 彭玉青^{a,b}

(河北工业大学 a. 计算机科学与软件学院; b. 河北省大数据重点实验室, 天津 300400)

摘 要: 目前的标签推荐系统使用张量来存储“用户-资源-标签”三维数据, 以挖掘三者之间潜在的语义关联。为更好地解决三维数据的稀疏性问题, 避免张量填充造成的数据失真, 提出基于标签惩罚机制的张量构建方法 PMUS 和基于随机梯度下降的张量分解方法 HOSGD。利用标签惩罚机制和用户评分构建张量, 使用随机梯度下降法对张量的展开矩阵进行分解。在此基础上, 结合 PMUS 和 HOSGD 提出 PMUS-HOSGD 算法对数据进行处理, 根据结果为用户进行个性化标签推荐。在数据集 MovieLens 上的实验结果表明, 与 CubeALS、HOSVD 和 CF 算法相比, 该算法能够有效提高标签推荐的准确率。

关键词: 标签推荐; 数据稀疏性; 张量构建; 张量分解; 惩罚机制; 随机梯度下降

中文引用格式: 杨 林, 顾军华, 官 磊, 等. PMUS-HOSGD 张量分解方法及其在标签推荐中的应用[J]. 计算机工程, 2018, 44(11): 300-305, 312.

英文引用格式: YANG Lin, GU Junhua, GUAN Lei, et al. PMUS-HOSGD tensor decomposition method and its application in tag recommendation[J]. Computer Engineering, 2018, 44(11): 300-305, 312.

PMUS-HOSGD Tensor Decomposition Method and Its Application in Tag Recommendation

YANG Lin^{a,b}, GU Junhua^{a,b}, GUAN Lei^{a,b}, ZHANG Yujuan^{a,b}, PENG Yuqing^{a,b}

(a. School of Computer Science and Software; b. Big Data Key Laboratory of Hebei Province, Hebei University of Technology, Tianjin 300400, China)

[Abstract] The current label recommended system uses tensors to store “user-resource-label” three-dimensional data, mining the potential semantic association among the three. In order to solve the sparseness problem of three-dimensional data and avoid the data distortion caused by the tensor filling process, this paper proposes a tensor construction method based on tag punishment mechanism PMUS and a tensor decomposing method based on stochastic gradient descent HOSGD. Tensor is constructed by combining tag punishment mechanism and user score, the expansion matrix of tensor is decomposed by the method of stochastic gradient descent. On this basis, combined with PMUS and HOSGD, the PMUS-HOSGD algorithm is proposed to deal with the data, and the personalized tag recommendation is made for the user according to the results. The experimental results on the data set MovieLens show that, compared with CubeALS, HOSVD and CF algorithms, this algorithm can effectively improve the accuracy of tag recommendation.

[Key words] tag recommendation; data sparseness; tensor construction; tensor decomposition; penalty mechanism; stochastic gradient descent

DOI: 10.19678/j.issn.1000-3428.0048759

0 概述

随着信息技术和互联网的发展, 互联网使用者从信息匮乏的时代步入了信息过载的时代, 用户的个性化需求也越来越大。用户的个性化特征注重的是用户的参与, 用户可以对资源(视频、歌曲、图片等)

赋予自定义的标签, 逐渐形成 Folksonomy 的大众分类法^[1], 该分类法不仅可以获取并分析用户的兴趣特征, 而且在一定程度上丰富了资源的描述信息。随着网络资源规模爆发式的增长^[2], 用于标注的标签越来越多, 用户面对大量的数据如何选出自己需要的标签越来越困难。解决信息过载的有效方法是

基金项目: 河北省科技计划项目(17210305D, 15210345); 天津市科技计划项目(16ZXHLSF00230)。

作者简介: 杨 林(1991—), 男, 硕士研究生, 研究方向为数据挖掘、推荐系统; 顾军华, 教授、博士; 官 磊、张宇娟, 硕士研究生; 彭玉青, 教授。

收稿日期: 2017-09-21 **修回日期:** 2017-11-08 **E-mail:** jhgu_hebut@163.com

个性化推荐^[3], 由此基于社会标注系统的标签推荐服务应运而生。例如, 给书籍和视频提供短评的豆瓣网、论文书签网站 CiteULike 和视频推荐网站 MovieLens 等都是利用社会标注系统对资源进行标注, 然后通过标签推荐系统将用户感兴趣的标签推荐给用户。推荐系统简化标注过程, 方便用户, 提高了标签的质量和标注的准确性。

标签推荐系统的核心是构建“用户-资源-标签”三维数据, 挖掘数据之间的潜在关系, 从而准确的为用户推荐标签。目前, 针对标签推荐系统已经展开了大量的研究。文献[4-5]将三维关系拆成“用户-资源”、“用户-标签”和“资源-标签”3个二维矩阵, 使用协同过滤算法进行处理。文献[6]受 Google 的 PageRank 算法^[7]启发提出 FolkRank 算法, 同样将三维关系拆分成3个二维关系。但是这些方法破坏了高维空间数据原本的特征结构, 丢失了三者之间的协同关系。为解决这个问题, 挖掘“用户-资源-标签”之间潜在的语义关联, 文献[8]提出使用三维张量存储“用户-资源-标签”数据。

在构建初始张量阶段, 目前使用最多的方法是用“0/1”构建张量, 这种方法构建容易, 可读性强, 但是不能体现出标签在资源中的差异。文献[9]发现由于热门标签通常有较大的权重, 导致推荐结果偏向于热门标签, 反而降低推荐的准确率。文献[10-12]提出使用 TF-IDF 来设置惩罚项, 用以区分标签对资源的重要程度, 可以减少热门标签对结果的影响。然而上述方法都忽略了用户对资源的偏好程度。

对构建完成的张量进行分解, 可以挖掘张量包含的潜在信息。文献[13]将奇异值分解方法推广到三维张量, 提出高阶奇异值分解 (High Order Singular Value Decomposition, HOSVD) 方法对张量进行分解, 该方法保留了三者的关联信息。文献[14]基于矩阵奇异值分解能有效地平滑数据矩阵中的数据特点, 在使用 HOSVD 进行分解的过程中, 结合用户朋友关系修正张量分解结果, 建立张量分解模型。文献[15]提出新的推荐算法 PITF (Pairwise Interaction Tensor Factorization), 该算法在张量分解的过程中加入3个二维关系之间的潜在联系, 推荐质量得到提高。针对目前推荐系统存在的稀疏性问题, 文献[16]在 CubeSVD^[17]的基础上进行改进, 使用 ALS 算法进行矩阵分解, 提出 CubeALS 推荐算法, 该算法有效提高了稀疏数据标签推荐的准确性。文献[18]提出一种基于上下文学习和张量分解的个性化推荐算法, 将用户和项目的上下文信息加入2个张量中, 有效改善数据稀疏性。文献[19]提出一种改进的基于张量分解的推荐算法, 引入基于标签综合共现的谱聚类方法, 使用 HOSVD-HOOI 算法对初始张量进行分解, 进一步优

化推荐效果。目前的张量分解方法多数以 SVD 为基础进行改进。使用 SVD 方法对张量的每个维度矩阵进行分解, 虽然在一定程度上提高了推荐的准确性, 但由于构建的初始张量极其稀疏, 需要在分解前对初始张量的展开矩阵进行填充, 这样存在2个问题: 1) 填充过程增加数据量, 同时增加算法复杂度; 2) 简单的数据填充易造成数据失真, 从而影响推荐结果的准确度。

针对上述问题, 本文结合 PMUS (Penalty Mechanism-User Score) 张量构建方法与 HOSGD (High Order Stochastic Gradient Descent) 张量分解方法, 提出 PMUS-HOSGD 算法对“用户-资源-标签”三维数据进行处理, 为用户推荐个性化标签。本文主要工作如下: 1) 在张量构建阶段, 提出惩罚机制与用户评分相结合的张量构建方法 PMUS 来计算标签的权重; 2) 在张量分解阶段, 提出基于随机梯度下降 (Stochastic Gradient Descent, SGD) 的高阶张量分解方法 HOSGD。

1 相关工作

1.1 张量及其矩阵展开

张量由多维数据组成, 是一个 N 维的向量空间, 一维张量是一个向量 (Vector), 二维张量是一个矩阵 (Matrix), 三维或者更高维的张量则是高维张量 (Tensor)。标签推荐系统使用三维张量存储数据, 3个维度分别代表用户、资源、标签。

使用 a_{ijk} 表示三维张量的值, 其大小代表用户 i 给资源 j 标注标签 k 的概率。例如, 用户 1 给资源 1 标注了标签 2, 则对应张量中的值为 $a_{112} = 1$, 其余的标注 0。

矩阵展开是将一个张量的元素重新排列 (即对张量的不同维度进行重新排列), 得到一个矩阵的过程。三维张量 $\mathbf{A} \in \mathbf{R}^{I_1 \times I_2 \times I_3}$ 在第 n 维度上的展开矩阵表示为 $\mathbf{X}_{(n)} \in \mathbf{R}^{I_n \times (I_1 \times I_2 \times \dots \times I_{n-1} \times I_{n+1} \times \dots \times I_N)}$ 。

1.2 张量分解

基本的张量分解算法 HOSVD 需要对张量 \mathbf{A} 每个维度 ($n = 1, 2, 3$) 的展开矩阵进行 SVD 分解, 计算公式是:

$$\mathbf{A}_n = \mathbf{U}_n \cdot \mathbf{S}_n \cdot \mathbf{V}_n^T \quad (1)$$

通过上述公式对 \mathbf{A}_n 进行奇异值分解, 分别得到3个维度的展开矩阵的 \mathbf{U} 矩阵和奇异值矩阵 \mathbf{S} 。

张量分解中 SVD 的作用是矩阵的低秩逼近, 即保留奇异值矩阵中所占比例较高的奇异值, 达到去噪目的, 可以将这个比值设为 75%, 并得到 $\hat{\mathbf{U}}_1$ 、 $\hat{\mathbf{U}}_2$ 、 $\hat{\mathbf{U}}_3$ 。

张量与矩阵的模积定义为一个张量 $\mathbf{X} \in \mathbf{R}^{I_1 \times I_2 \times \dots \times I_N}$ 和一个矩阵 $\mathbf{U} \in \mathbf{R}^{J \times I_n}$ 的 n -mode 乘积 ($\mathbf{X} \times_n \mathbf{U}$) $\in \mathbf{R}^{I_1 \times I_2 \times \dots \times I_{n-1} \times J \times I_{n+1} \times \dots \times I_N}$, 其元素定义为:

$$(\mathbf{X} \times \mathbf{U})_{i_1 i_2 \dots i_{n-1} j_{n+1} \dots j_N} = \sum_{i_n=1}^{I_N} x_{i_1 i_2 \dots i_n} \mathbf{u}_{j_n} \quad (2)$$

如果 $J < I_n$, 那么张量和矩阵的乘积可以看作降维的过程, 它把一个高维度的张量映射到一个低维度的张量空间。例如, 一个矩阵 $\mathbf{A} \in \mathbb{R}^{4 \times 7}$ 和一个张量 $\mathbf{G} \in \mathbb{R}^{7 \times 5 \times 8}$ 的乘积得到张量 $\mathbf{Y} \in \mathbb{R}^{4 \times 5 \times 8}$, 张量 \mathbf{G} 的第一个维度由 7 变成了 4。

核心张量存储着张量各个维度之间的信息, 因此需要利用 SVD 分解得到的 $\hat{\mathbf{U}}_1$ 、 $\hat{\mathbf{U}}_2$ 和 $\hat{\mathbf{U}}_3$ 计算张量 \mathbf{A} 的核心张量 $\hat{\mathbf{S}}$ 。

$$\hat{\mathbf{S}} = \mathbf{A} \times \hat{\mathbf{U}}_1^T \times \hat{\mathbf{U}}_2^T \times \hat{\mathbf{U}}_3^T \quad (3)$$

根据张量和矩阵模积的计算方法, 构建完成核心张量 $\hat{\mathbf{S}}$ 后, 就可以得到初始张量的近似张量 $\hat{\mathbf{A}}$, 计算公式表示为:

$$\hat{\mathbf{A}} = \hat{\mathbf{S}} \times \hat{\mathbf{U}}_1 \times \hat{\mathbf{U}}_2 \times \hat{\mathbf{U}}_3 \quad (4)$$

其中, $\hat{\mathbf{S}}$ 是核心张量, $\hat{\mathbf{U}}_1$ 、 $\hat{\mathbf{U}}_2$ 、 $\hat{\mathbf{U}}_3$ 是左奇异矩阵降维后的矩阵。

2 PMUS-HOSGD 张量分解方法

传统的张量分解算法 HOSVD 可以挖掘“用户-资源-标签”三维数据之间的关系, 但在实际应用中, 用户仅对个别资源进行标签标注, 这会导致张量中的数据极度稀疏。目前常用的 HOSVD 算法在张量分解的过程前需要对稀疏矩阵进行填充, 这样会造成数据的失真。因此, 本文结合 PMUS 张量构建方法与 HOSGD 张量分解方法, 提出 PMUS-HOSGD 算法对“用户-资源-标签”三维数据进行处理, 为用户推荐个性化标签。

2.1 PMUS 初始张量构建

在“用户-资源-标签”三维张量中, “0/1”构建方法简单易行, 但是标签之间没有区分度, 因此可以使用 PMUS 的思想计算标签的权重; 同时, 用户对资源的评分可以体现出用户对资源的偏好程度, 用标签的权重乘以评分可以很好地区分出标签之间以及用户和资源之间的重要度, 最终构建整个张量。

PMUS 的主要思想是: 如果一个标签在一个资源中出现多次, 并且在其他资源中很少出现, 则可以认为该标签具有很好的区分度, 同时如果一个用户给资源的评分越高, 用户对这个资源的偏爱程度越大, 则给这个视频标注的标签的概率就越大。

使用 PMUS 计算标签权值的过程如下:

1) 计算标签 t 对于资源 i 的重要度 $import(t, i)$, 针对每个资源都有一个标签权重向量, 公式如下:

$$import(t, i) = num(t, i) \times \ln\left(\frac{N}{n_i} + 0.01\right) \quad (5)$$

其中, $num(t, i)$ 表示全部用户给资源 i 标注的标签集合中标签 t 出现的次数, $\ln\left(\frac{N}{n_i} + 0.01\right)$ 表示标签的惩

罚项, N 表示资源总数, n_i 表示在所有资源中被标签 t 标注的资源数量, 常数项 0.01 是为了防止计算结果返回 0。

2) 根据重要度 ($import$) 计算标签 t 在用户 u 对资源 i 标注的标签集中占的比例权重 $weight(u, t, i)$, 公式如下:

$$weight(u, t, i) = import(t, i) / sum_weight(u, i) \quad (6)$$

其中, $sum_weight(u, i)$ 是用户 u 给资源 i 标注的所有标签的重要度的总和。

3) $weight(u, t, i)$ 乘以用户 u 给资源 i 的评分就是张量中用户 u 在资源 i 中使用标签 t 的权值 $Value(u, t, i)$, 公式如下:

$$Value(u, t, i) = weight(u, t, i) \times score(u, i) \quad (7)$$

2.2 基于随机梯度下降的张量分解算法 HOSGD

传统的 SVD 算法对二维矩阵进行分解, 可以求出对应的特征矩阵, 随着用户数量、资源数量和标签数量的急剧增长, SVD 分解带来的误差和复杂度也在不断增加, 正是由于这些问题, Simon Funk 发表了一个只考虑已有评分的矩阵分解方法, 称为 Funk-SVD, 也就是被文献[20]称为隐语义模型的矩阵分解方法, 该方法使用梯度下降法 (Gradient Descent, GD) 最小化训练集中观察值的 RMSE (Root Mean Squared Error), 在二维矩阵分解中取得了较好的推荐效果。

标准的梯度下降法在更新变量前要对所有的样本计算误差并汇总, 导致算法收敛速度较慢, 因此本文使用随机梯度下降法, SGD 是在梯度下降法的基础上, 在迭代过程中使用部分样本计算梯度, 因此其比标准梯度下降法有更高的收敛速度。

借鉴 SGD 在二维矩阵分解领域中的应用, 本文提出 HOSGD 张量分解算法。HOSGD 算法在张量进行分解的过程中, 为提高准确性, 使用 SGD 算法对展开矩阵分解, 降低了传统分解方法带来的计算复杂度及误差。

在用户给资源标注标签的过程中, 形成了若干 {用户, 资源, 标签} 数据, 使用 PMUS 方法构建利用式(5)、式(6)和式(7)计算 $Value(u, t, i)$, 得到三维张量 $\mathbf{A} \in \mathbb{R}^{x \times y \times z}$, HOSGD 算法对张量的展开矩阵进行 SGD 分解, 得到 3 个特征矩阵, 进而计算出初始张量的核心张量, 然后可以得到初始张量的近似张量。算法描述如下:

算法 1 PMUS-HOSGD 张量分解

输入 用户、资源、标签数据三元组 (u, i, t)

输出 初始张量的近似张量

1. 使用 PMUS 方法按照式(5)、式(6)、式(7)计算标签的权值
2. 使用张量 \mathbf{A} 存储式(7)计算得到的 $Value(u, t, i)$
3. 将张量 \mathbf{A} 按照 3 个维度展开得到 \mathbf{A}_1 、 \mathbf{A}_2 、 \mathbf{A}_3
4. for $i = 1$ to 3

5. 按照算法 2 对 A_i 进行处理, 得到 P_1, P_2, P_3
6. end for

7. 按照式(3), 计算核心张量 $\hat{S} \in R^{k_1 \times k_2 \times k_3}$
8. 按照式(4), 计算张量 A 的近似张量 $\hat{A} \in R^{X \times Y \times Z}$

在算法 1 中, 对于三维张量 $A \in R^{X \times Y \times Z}$, 其中, 用户、资源和标签的数量分别为 X, Y 和 Z , 初始张量 A 展开得到的 3 个矩阵 A_1, A_2, A_3 的规模分别为 $X \times YZ, Y \times XZ$ 和 $Z \times XY$, 按照算法 2, 分别对 A_1, A_2, A_3 进行分解, 得到每个维度的特征矩阵 P_1, P_2, P_3 , 其中 $P_1 \in R^{X \times k_1}, P_2 \in R^{Y \times k_2}, P_3 \in R^{Z \times k_3}$, 3 个特征矩阵的特征数 k_1, k_2, k_3 是根据数据规模设定的, 一般 $k_i < \min(X, Y, Z)$ 。

在算法 1 的第 5 步, 对于张量展开矩阵 $A_i \in R^{m \times n}$, 要找到一个低秩矩阵 \hat{A}_i 逼近 A_i , 并使损失函数最小:

$$L = \sum_{(U, I) \in K} (A_i - \hat{A}_i)^2 \quad (8)$$

其中, \hat{A}_i 可以通过矩阵分解方法计算, $\hat{A}_i = PQ^T, P \in R^{m \times k_i}, Q \in R^{n \times k_i}, k_i$ 表示特征数, 损失函数可以写成如下形式:

$$L = \sum_{(U, I) \in A} (A_{i_{UI}} - P_U \times Q_I)^2 + \lambda (\|P_U\|^2 + \|Q_I\|^2) \quad (9)$$

其中, 左边第 1 项是误差项, 用原始矩阵中有值的项减去 P 和 Q 对应行和列相乘得到的值, 左边第 2 项是正则化项, 防止过拟合。对 L 最小化就得到了 P 和 Q 。

通过式(8)分别对 P_U 和 Q_I 求其梯度:

$$\frac{\partial L}{\partial P_U} = 2 \sum_{I \in A(U)} [(A_{UI} - P_U \times Q_I) \times (-Q_I) + \lambda P_U] \quad (10)$$

$$\frac{\partial L}{\partial Q_I} = 2 \sum_{U \in A(I)} [(A_{UI} - P_U \times Q_I) \times (-P_U) + \lambda Q_I] \quad (11)$$

其中, λ 表示正则化参数, $A(U)$ 表示矩阵 A 第 U 行中不为 0 或者空的列, $A(I)$ 表示矩阵 A 第 I 列中不为 0 或者空的行, 通过矩阵 A 第 U 行, 可以得到矩阵 P 第 U 行的梯度, 也就是 P_U 需要更新的值:

$$P_U = P_U - \alpha \left(\frac{\partial L}{\partial P_U} \right) \quad (12)$$

$$Q_I = Q_I - \alpha \left(\frac{\partial L}{\partial Q_I} \right) \quad (13)$$

其中, α 表示步长, 对矩阵 P 中所有的 P_U 进行更新, 或者对 Q 矩阵中所有的 Q_I 进行更新, 就完成了迭代, 在每次迭代的过程中, 实现了 P 和 Q 矩阵的一次更新, 损失函数 L 的值减小。算法描述如下:

算法 2 基于 SGD 的张量展开矩阵分解

- 输入 张量的展开矩阵 A ,
输出 A_i 的特征矩阵 P

1. 先初始化 P_U^0 和 Q_I^0 , 可以使用均值填充或者符合正态分布的随机数

2. 根据式(10)和式(12), 以及 P_U^0 和 Q_I^0 更新矩阵 P 中所有的 P_U , 得到经过第一次迭代的 P_U^1

3. 根据式(11)和式(13), 以及 P_U^1 和 Q_I^0 更新矩阵 Q 中所有的 Q_I , 得到经过第一次迭代的 Q_I^1

4. 步骤 2 和步骤 3 是一次迭代的过程, 多次执行步骤 2 和步骤 3, 不断更新 P_U 和 Q_I 的值, 直到完成迭代次数 t 或者误差小于阈值, 得到 P 和 Q 的最优解

在算法 2 中, λ 和 α 参数需要在实验中多次调优得到。算法的核心在于每次更新 P_U 和 Q_I 时只使用原始矩阵中有值的部分, 得到 P 和 Q 的最优解, P 即为对应的特征矩阵。

通过算法 1 得到的近似张量 \hat{A} 里面包含了推荐结果, 初始张量值为“0”的地方, 也就是用户没有使用过的标签, 在近似张量中可以找到一个近似的值, 这个值就是用户使用该标签的概率, 基于这种概率, 可以为用户 u 针对资源 i 生成一个推荐列表, 将 $\{$ 用户 u , 资源 i $\}$ 中 top_ N 的标签列表推荐给用户。

PMUS-HOSGD 张量分解方法的时间复杂度主要是在对每个维度展开矩阵的分解基础上进行计算的。算法 2 矩阵分解的时间复杂度是 $O(t \times X \times k_1 \times n_1')$, 据此可得算法 1 的时间复杂度是 $O(t \times (X \times k_1 \times n_1' + Y \times k_2 \times n_2' + Z \times k_3 \times n_3'))$, 其中, t 为迭代次数, k 为特征数, n' 是矩阵中平均每行非空数据的数目。

3 实验结果与分析

本文使用相同的数据集和评价标准对比张量构建方法和张量分解方法, 采用常用的评价指标验证算法的有效性。

3.1 数据集

本文使用 MovieLens 数据集, 包含用户对视频的评分, 以及用户给视频标注的标签数据。

使用 MovieLens 数据集构建的三维张量极其稀疏, 因此对初始数据进行预处理, 预处理后的数据中每个用户都对 15 个或 15 个以上视频打过标签, 每个视频都由 15 个或 15 个以上用户打过标签。处理后的数据中用户、视频、标签的数量分别是 184、122、378, 有 20 149 条“用户、资源、标签”数据。

3.2 评价标准

本文使用准确率 (Precision)、召回率 (Recall) 和 F 值^[21] 作为算法的评价标准。在推荐系统中, 准确率表示在推荐列表中得到的推荐结果与测试集中实际情况相同的物品数与所有的推荐物品数的比值, 召回率指的是推荐列表中准确的结果占测试样本的比例。在实验过程中, 将预处理后的数据集分成 2 部分: 训练集和测试集, 其中, 训练集占 75%, 测试集占 25%。

准确率和召回率的计算公式如下:

$$Precision = \frac{|test \cap top_N|}{N} \quad (14)$$

$$Recall = \frac{|test \cap top_N|}{|test|} \quad (15)$$

其中, $test$ 表示测试集, top_N 表示推荐的结果, N 表示推荐的数目, 准确率和召回率的值越高推荐效果越好。

F 值作为常用评价指标, 能更好地反映推荐结果的效果, F 的值越高推荐效果越好。

$$F = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (16)$$

3.3 实验结果

为验证 PMUS 方法构建张量能提高推荐结果的准确率, 实验结合使用 HOSVD 分解算法, 分别对比“0/1”、TF-IDF 和 PMUS 张量构建方法的准确率、召回率和 F 值。实验结果如图 1~图 3 所示, 每个图中各有 3 条曲线, 分别代表“0/1”、TF-IDF 和 PMUS 张量构建方法结合 HOSVD 分解方法的实验结果, 每条曲线有 8 个节点, 横轴代表 top_N 的值, 纵轴分别代表准确率、召回率和 F 值。

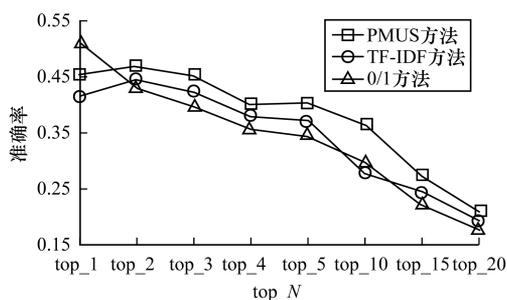


图 1 3种构建方法的准确率比较

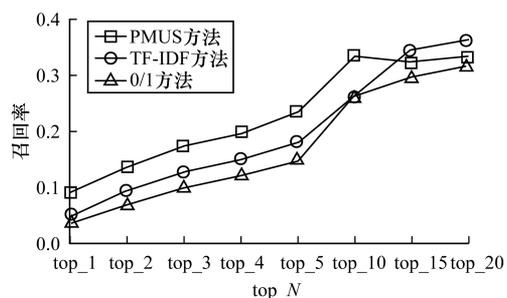


图 2 3种构建方法的召回率比较

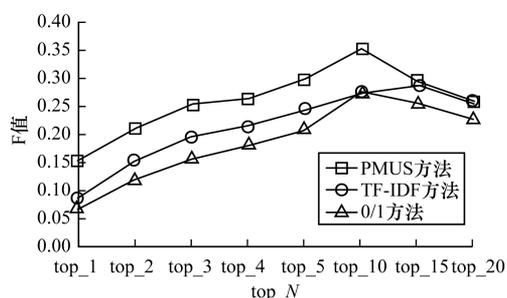


图 3 3种构建方法的 F 值比较

由图 1~图 3 中 3 种不同构建方法的对比可知, 在 top_1 时, 使用“0/1”构建的张量进行分解推荐的准确率比 PMUS 和 TF-IDF 方法要好, 但是随着 N 的增长, PMUS 方法构建的张量的准确率要高于其他 2 种算法, 比 TF-IDF 平均高 0.03。在推荐数量小于 10 时, PMUS 方法构建张量的召回率和 F 值要明显高于其他 2 种方法, 说明使用 PMUS 方法构建张量, 张量权值在加入标签对视频的权重以及用户对视频的评分后, 使用用户对视频标注的标签权值更加真实。在实际推荐系统中, 给用户推荐的少量标签不止 1 个, 因此使用 PMUS 方法构建张量得到的推荐结果要优于普通的“0/1”和 TF-IDF 构建方法。

为进一步验证 HOSGD 推荐算法的性能, 本文结合使用 PMUS 方法构建张量, 对 HOSGD 与 HOSVD、协同过滤 (Collaborative Filtering, CF) 和 CubeALS 算法进行了对比。基于 CF 的标签推荐算法是目前应用比较广泛的个性化推荐算法; HOSVD 是一种经典的张量分解算法, 被大量的应用于三维数据推荐领域, 而且取得了良好的实验结果; CubeALS 推荐算法与其他优秀的算法对比, 推荐效果有显著提高。

在用 HOSGD 分解展开矩阵的过程中, 涉及到步长 α 、正则化系数 λ 、特征因子数目 k 、迭代次数和阈值 5 个参数。 α 过大可能会导致迭代不收敛, 从而发散, 因此 α 分别取 0.01、0.05、0.1、0.2、0.3、0.6 进行对比; k 数目过多会导致收敛速度慢, 程序时间复杂度高, 因此特征因子 k 数目取 10 到 20 进行对比。通过多次实验, 发现在迭代 50 次左右误差结果接近 0.08, 趋于稳定。结合实验结果, 张量分解中取步长 α 为 0.2, 正则化系数 λ 为 0.0003, 特征因子 k 数目为 17, 迭代 50 次, 阈值为 0.08。实验结果如图 4~图 6 所示。由图 4~图 6 可知, 4 种算法的准确率呈下降趋势, HOSGD 的准确率在 top_N 小于 5 的情况下均高于其他算法, 尤其在 top_1 至 top_4 阶段 HOSGD 的准确率平均比 CubeALS 提升 0.07。在 top_1 至 top_5 阶段 HOSGD 的召回率和 F 值也明显高于次优的 CubeALS 算法。在实际推荐系统中, 给用户提供 1-5 个标签, HOSGD 算法符合实际要求。实验结果表明, 使用随机梯度下降的张量分解算法 HOSGD 能够充分利用 SGD 方法的优势, 有效处理稀疏张量, 减少误差。

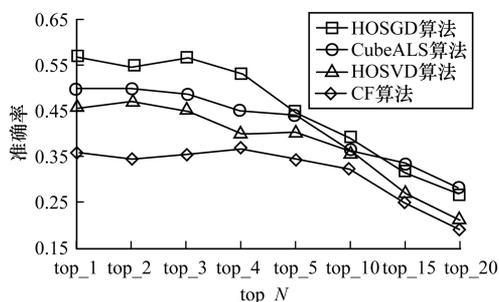


图 4 4种推荐算法的准确率比较

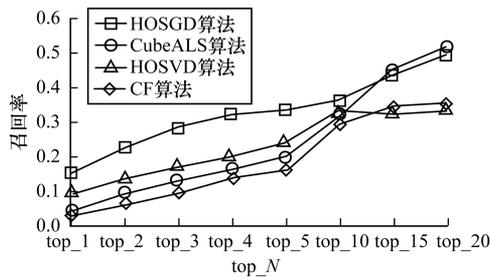


图 5 4 种推荐算法的召回率比较

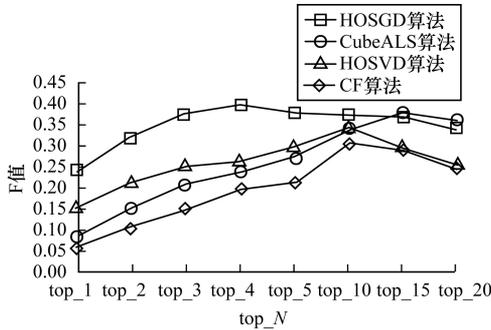


图 6 4 种推荐算法的 F 值比较

综上所述,使用 PMUS 构建张量并结合使用基于随机梯度下降法的 HOSGD 进行张量分解的 PMUS-HOSGD 算法,可以有效提高标签推荐的准确率。

4 结束语

在个性化标签推荐领域,使用张量存储“用户-资源-标签”数据是一种很好的数据表示形式,但由于三维数据的稀疏性,传统的张量构建方法和张量分解方法的推荐准确率较低。因此,本文利用 PMUS 构建张量,并结合基于随机梯度下降法的 HOSGD 对张量进行分解。实验结果表明,与 HOSVD、CF 和 CubeALS 算法相比,PMUS-HOSGD 算法具有更好的推荐效果。下一步将重点研究在大数据量的情况下如何提高推荐速度,并使用分布式平台运行该算法。

参考文献

[1] 刘向红,宋 文,姚 朋.基于标签的 Folksonomy 机制研究——以 CiteUlike 为例[J]. 图书馆理论与实践, 2010,32(5):29-33.

[2] KIM J,LEE D,CHUNG K Y. Item recommendation based on context-aware model for personalized u-healthcare service[J]. Multimedia Tools and Applications, 2014, 71(2):855-872.

[3] 靳延安. 社会标签推荐技术与方法研究[D]. 武汉: 华中科技大学,2011.

[4] TSO-SUTTER K H L, MARINHO L B, SCHMIDT-THIEME L. Tag-aware recommender systems by fusion of collaborative filtering algorithms[C]//Proceedings of

2008 ACM Symposium on Applied Computing. New York, USA: ACM Press, 2008:1995-1999.

[5] MARINHO L, HOTH O A, SCHMIDT-THIEME L, et al. Tag recommendations in social bookmarking systems[J]. AI Communications, 2008, 21(4):231-247.

[6] HOTH O A, JÄSCHKE R, SCHMITZ C, et al. Information retrieval in folksonomies: search and ranking [C]// Proceedings of the 3rd European Conference on the Semantic Web: Research and Applications. Berlin, Germany: Springer, 2006:411-426.

[7] PAGE L, BRIN S, MOTWANI R, et al. The pagerank citation ranking: bringing order to the Web, in world wide Web internet and Web information systems [EB/OL]. [2017-08-20]. https://www.researchgate.net/publication/221996591_The_PageRank_Citation_Ranking_Bringing_Order_to_the_Web.

[8] SYMEONIDIS P, NANOPOULOS A, MANOLOPOULOS Y. Tag recommendations based on tensor dimensionality reduction[C]//Proceedings of 2008 ACM Conference on Recommender Systems. New York, USA: ACM Press, 2008:43-50.

[9] VALLET D. Content-based recommendation in social tagging systems [C]//Proceedings of the 4th ACM Conference on Recommender Systems. New York, USA: ACM Press, 2010:237-240.

[10] KUNCORO B A, ISWANTO B H. TF-IDF method in ranking keywords of Instagram users' image captions[C]// Proceedings of 2015 International Conference on Information Technology Systems and Innovation. Washington D. C., USA: IEEE Press, 2015:1-5.

[11] HUANG X, WU Q. Micro-blog commercial word extraction based on improved TF-IDF algorithm[C]// Proceedings of IEEE Region 10 Annual International Conference. Washington D. C., USA: IEEE Press, 2014:1-5.

[12] 王 成,朱志刚,张玉侠,等. 基于用户的协同过滤算法的推荐效率和个性化改进[J]. 小型微型计算机系统, 2016,37(3):428-432.

[13] LATHAUWER L D, MOOR B D, VANDEWALLE J. A multilinear singular value decomposition [J]. SIAM Journal on Matrix Analysis and Applications, 2006, 21(4):1253-1278.

[14] 丁小焕,彭甫谔,王 琼,等. 融合朋友关系和标签信息的张量分解推荐算法[J]. 计算机应用, 2015, 35(7):1979-1983.

[15] RENDLE S, SCHMIDT-THIEME L. Pairwise interaction tensor factorization for personalized tag recommendation [C]//Proceedings of the 3rd ACM International Conference on Web Search and Data Mining. New York, USA: ACM Press, 2010:81-90.

[16] 李 改,潘 嵘,李 磊. CubeALS: 新的三维协同过滤推荐算法[J]. 计算机科学与探索, 2012, 6(2):156-164.

5 结束语

本文提出了一种基于多服务器的 DASH 客户端优化算法,借助于 SDN,实现了对服务器可用带宽的准确测量。通过将多个视频分片组成一个视频块,并对这个视频块中的码率进行统一化处理,一方面减少了客户端视频码率的频繁切换,另一方面提高了总体视频质量。同时,本文采用 Q 学习算法,通过自我学习训练 Q 值矩阵,为每一个视频块选择合适的码率,大幅提高了码率选择的效率。下一步将考虑客户端和服务器之间的路径规划问题,使客户端能够自动切换到网络状况较好的传输路径上。

参考文献

- [1] Cisco. The Zettabyte era: trends and analysis [EB/OL]. [2017-09-10]. <https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-hyperconnectivity-wp.html>.
- [2] STOCKHAMMER T. Dynamic adaptive streaming over HTTP--: standards and design principles [C]// Proceedings of the 2nd Annual ACM Conference on Multimedia Systems. New York, USA: ACM Press, 2011:133-144.
- [3] JIANG J, SEKAR V, ZHANG H. Improving fairness, efficiency, and stability in HTTP-based adaptive video streaming with festive [C]// Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies. New York, USA: ACM Press, 2014:97-108.
- [4] PU W, ZOU Z, CHEN C W. Dynamic adaptive streaming over HTTP from multiple content distribution servers [C]// Proceedings of Global Telecommunications Conference. Washington D. C., USA: IEEE Press, 2011: 1-5.
- [5] ZHANG S, LI B, LI B. Presto: towards fair and efficient HTTP adaptive streaming from multiple servers [C]// Proceedings of IEEE International Conference on Communications. Washington D. C., USA: IEEE Press, 2015:6849-6854.
- [6] McKEOWN N, ANDERSON T, BALAKRISHNAN H, et al. OpenFlow: enabling innovation in campus networks [J]. ACM SIGCOMM Computer Communication Review, 2008, 38 (2): 69-74.
- [7] 曹荣珍. 基于 OpenFlow 的视频用户 QoE 优化研究 [D]. 北京: 北京交通大学, 2016.
- [8] BOUTEN N, CLAEYS M, VAN POECKE B, et al. Dynamic server selection strategy for multi-server HTTP adaptive streaming services [C]// Proceedings of the 12th International Conference on Network and Service Management. Hong Kong, China: [s. n.], 2016:82-90.
- [9] LIU L, ZHOU C, ZHANG X, et al. Probabilistic chunk scheduling approach in parallel multiple-server DASH [C]// Proceedings of Visual Communications and Image Processing Conference. Washington D. C., USA: IEEE Press, 2014:5-8.
- [10] 左青云, 陈 鸣, 赵广松, 等. 基于 OpenFlow 的 SDN 技术研究 [J]. 软件学报, 2013, 24 (5): 1078-1097.
- [11] WATKINS C J, DAYAN P. Q-learning [J]. Machine Learning, 1992, 8 (3/4): 279-292.
- [12] CLAEYS M, LATRÉ S, FAMAËY J, et al. Design of a Q-learning-based client quality selection algorithm for HTTP adaptive video streaming [C]// Proceedings of Adaptive and Learning Agents Workshop. Saint Paul, USA: [s. n.], 2013:30-37.
- [13] KAEHLING L P, LITTMAN M L, MOORE A W. Reinforcement learning: a survey [J]. Journal of artificial intelligence research, 1996, 4:237-285.
- [14] MOK R K, CHAN E W, CHANG R K. Measuring the quality of experience of HTTP video streaming [C]// Proceedings of IFIP/IEEE International Symposium on Integrated Network Management. Washington D. C., USA: IEEE Press, 2011:485-492.
- [15] NEMETHOVA O, RIES M, ZAVODSKY M, et al. PSNR-based estimation of subjective time-variant video quality for mobiles [EB/OL]. [2017-09-10]. <https://pdfs.semanticscholar.org/a496/411a16b59c265c53aa07de11a9c32ec90f65.pdf>.
- [17] SUN J T, ZENG H J, LIU H, et al. CubeSVD: a novel approach to personalized Web search [C]// Proceedings of the 14th International Conference on World Wide Web. New York, USA: ACM Press, 2005:382-390.
- [18] 刘振娇, 王新华. 基于上下文学习和张量分解的个性化推荐 [J]. 山东师范大学学报 (自然科学版), 2017, 32 (1): 36-42.
- [19] 陈梅梅, 薛康杰. 基于改进张量分解模型的个性化推荐算法研究 [J]. 数据分析与知识发现, 2017 (3): 38-45.
- [20] KOREN Y, BELL R, VOLINSKY C. Matrix factorization techniques for recommender systems [J]. Computer, 2009, 42 (8): 30-37.
- [21] RENDLE S, MARINHO L B, NANOPOULOS A, et al. Learning optimal ranking with tensor factorization for tag recommendation [C]// Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA: ACM Press, 2009:727-736.

编辑 金胡考

(上接第 305 页)

编辑 司淼淼