基于张量方法的数据约简算法研究

Research on Data Reduction Algorithm Based on Tensor Method

金青海　晏祖根

Qinghai Jin　Zugen Yan

江西工程学院　中国·江西 新余 338000

Jiangxi Institute of Engineering, Xinyu, Jiangxi, 338000, China

摘　要：互联网时代收集了大量的用户使用网络的信息记录，对用户的网络使用行为进行分析是当前人工智能研究的一个热点。现有的推荐算法存在对用户的个性化推荐准确度不高和处理多路数据比较困难的不足，我们提出处理推荐系统典型的三路数据（用户、项目、评分）的统一架构，将这类数据用一个三阶张量表示，运用高阶奇异值分解技术对其进行隐含语义挖掘和降维，提出的算法具有较大的实用价值。

Abstract: In the Internet age, a large number of information records of users' network usage have been collected, and the analysis of users' network usage behavior is a hot spot in current artificial intelligence research. The existing recommendation algorithms have some shortcomings, such as low accuracy of personalized recommendation to users and difficulty in processing multi-channel data. We propose a unified framework for processing typical three-channel data (users, items, ratings) of recommendation system, which is expressed by a third-order tensor, and the implicit semantic mining and dimensionality reduction are carried out by using high-order singular value decomposition technology. The proposed algorithm has great practical value.

关键词：张量；降维；多路数据

Keywords: tensor; dimension reduction; multiplex data

DOI: 10.12346/csai.v1i2.7118

1 引言

大数据时代产生了大量的高维数据，比如图像识别、信息检索、生物大数据分析等领域获得的数据都是高维的。为了更好地对这些数据进行分析，经常需要对大型数据进行维数约简（降维）。经典的基于流形学习（Manifold Learning）的降维方法主要有主成分分析（Principal Component Analysis，PCA）[1]（又称为 K-L 变换）、局部线性嵌入（Locally Linear Embedding， LLE）[2]、等度规映射（ISOMAP）[3] 等，这些算法都是典型的一维数据约简方法，它们已经被广泛应用于模式识别等领域。传统的向量化算法降维的办法是将高维向量空间中的数据投影到低维向量空间，这些方法存在一些弊端：①会损失掉原始数据内部的一些重要信息，比如图像数据的空间结构信息；②向量化后的数据一般而言维数比较高，这给机器带来较大的计算负担。以图像数据为例，基于张量思想的算法将灰度图像视作其行向量空间 $\mathbb{R}^{d_1}$ 和列向量空间 $\mathbb{R}^{d_2}$ 的张量积，即一幅灰度图像 $I \in \mathbb{R}^{d_1} \otimes \mathbb{R}^{d_2}$（一个图像集合可视作嵌入在张量空间 $\mathbb{R}^{d_1} \otimes \mathbb{R}^{d_2}$ 中的子流形 [4]），它的维数是行向量空间的维数 d_1 和列向量空间的维数 d_2，一般远小于图像向量化后的维数 $d_1 d_2$，同时保存了图像数据的空间结构，使得计算的效率高于传统的线性（向量化）降维方法。

许多实际数据在曲线坐标系中可以被自然地表示成张量形式，张量表示的数据具有与坐标系的选择无关的性质，我们在多重线性代数、微分几何、流形等数学理论的基础之上，通过研究典型的几种高维数据的特征，建立了具有张量特征的数据的约简表示模型。

现有的推荐系统算法无法很好地处理三路数据（比如三元组（用户、项目、评分）数据），它们采取的办法是将一

【作者简介】金青海（1987-），男，中国江西九江人，硕士，从事科学与工程计算研究。

个三路数据拆分成 3 个二维（成对的）数据：（用户、项目）、（用户、评分）、（项目、评分），这会损失原始三路数据蕴含的部分内在的语义信息，普通的推荐系统，比如协同筛选算法（Collaborative Filtering，CF）[5] 只能应用于二维数据的处理。为挖掘三路数据隐藏的内在关联，通过对推荐系统典型的三路数据进行分析，将这些数据表示成一个三阶张量，运用高阶奇异值分解（High Order Singular Value Decomposition，HOSVD）技术对它们进行 3-mode 分析，提出了处理三路数据的一个有效方法。

2 流形学习

这里所说的流形（Manifold）是指拓扑流形，其定义如下：

定义 1：一个 d 维的拓扑流形 M 是指一个具有可数基的 Hausdorff 拓扑空间，它的每一点 $P\in M$，都存在 P 的一个邻域 U 与 $\mathbb{R}^d$ 中的一个开子集同胚。

通常的曲线、曲面分别称为一维流形和二维流形。

定义 2：黎曼流形空间 (M,D) 中的一条光滑曲线 $\gamma:I=(a,b)\to M$，$\gamma'\equiv\dfrac{d\gamma}{dt}$ 是沿 γ 的切向量场，若 γ 的切向量 γ' 沿 γ 是平行的，即 $\dfrac{D}{dt}(\dfrac{d\gamma}{dt})\equiv 0$，则 γ 称为黎曼流形空间 (M,D) 的测地线。

黎曼流形空间任意两点之间存在多条测地线，其中最短的测地线长度被定义为黎曼空间两点间的距离。测地线是欧式空间两点之间直线段概念在黎曼空间中的推广。

设 $Y\in\mathbb{R}^d$ 是一个 d 维的数据集，$f:Y\to\mathbb{R}^D$ 是一个光滑嵌入，$D\gg d$，流形学习的目标是基于给定的 $\mathbb{R}^D$ 中的观测数据集 $\{x_i\}$ 去恢复 Y 和 f，即通过特定的嵌入映射得到高维空间的低维流形，达到数据压缩的目的。f 的实现形式有一般有两种，第一种是黎曼几何意义下的等距嵌入，第二种是保角嵌入，它保角度不保长度，保角嵌入包括所有的等距嵌入和许多其他的映射（比如立体投影中的 Mercator 投影）。假定 y、z 是 Y 中的任意两点，按照等距嵌入的定义，f 保留了路径长度，Y 中 y 和 z 之间的最短路径与沿着 $f(Y)$ 的 $f(y)$ 和 $f(z)$ 之间的最短路径是等长的，即测地线意义下 Y 和 $f(Y)$ 是等距的。

近年来，出现了大量的基于流形学习理论的降维算法，它们已广泛应用于生物数据挖掘、计算机视觉等领域。

3 张量基础

一个 k 阶张量是基于 k 个向量空间的实值多线性函数：

$$T:\ \mathbb{R}^{n_1}\times\cdots\times\mathbb{R}^{n_k}\to\mathbb{R}$$

k 称为 T 的阶。一个多线性函数分别是每一个变量的线性函数，$\mathbb{R}^{n_i}\ (i=1,2,\cdots,k)$ 上所有的 k 阶张量的集合记作 $\mathcal{T}^k$，它们的运算满足通常的加法和数量乘法：

$$(T+T')(\boldsymbol{a}_1,\cdots,\boldsymbol{a}_k)\ =\ T(\boldsymbol{a}_1,\cdots,\boldsymbol{a}_k)\ +\ T'(\boldsymbol{a}_1,\cdots,\boldsymbol{a}_k)$$

$$(aT)(\boldsymbol{a}_1,\cdots,\boldsymbol{a}_k)\ =\ a(T(\boldsymbol{a}_1,\cdots,\boldsymbol{a}_k))$$

其中，向量 $\boldsymbol{a}_i\in\mathbb{R}^{n_i}$。

给定两个张量 $\mathcal{T}_1\in\mathcal{T}^k$，$\mathcal{T}_2\in\mathcal{T}^l$，定义一个映射：

$$\mathcal{T}_1\otimes\mathcal{T}_2:\ \mathbb{R}^{n_1}\times\cdots\times\mathbb{R}^{n_{k+l}}\to\mathbb{R}$$

其中，$\mathcal{T}_1\otimes\mathcal{T}_2(\boldsymbol{a}_1,\cdots,\boldsymbol{a}_{k+l})\ =\ \mathcal{T}_1(\boldsymbol{a}_1,\cdots,\boldsymbol{a}_k)\mathcal{T}_2(\boldsymbol{a}_{k+1},\cdots,\boldsymbol{a}_{k+l})$。

由 $\mathcal{T}_1$ 和 $\mathcal{T}_2$ 的多重线性易得 $\mathcal{T}_1\otimes\mathcal{T}_2$ 分别线性地依赖于各个参数 $\boldsymbol{a}_i$，故 $\mathcal{T}_1\otimes\mathcal{T}_2$ 是一个（$k+l$）阶张量，称为 $\mathcal{T}_1$ 和 $\mathcal{T}_2$ 的张量积。

对于一阶张量，它们是 $\mathbb{R}^{n_1}$ 的余向量，即 $\mathcal{T}^1=\mathcal{R}^{n_1}$，$\mathcal{R}^{n_1}$ 是 $\mathbb{R}^{n_1}$ 的对偶空间。二阶张量空间是两个一阶张量空间的积，即 $\mathcal{T}^2\ =\ \mathcal{R}^{n_1}\otimes\mathcal{R}^{n_2}$，$\mathcal{R}^{n_1}\otimes\mathcal{R}^{n_2}$ 中每一个二阶张量唯一的对应于一个 $n_1\times n_2$ 矩阵。

定义 3：两个张量 $\mathcal{A}$、$\mathcal{B}\in\mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ 的数量积（内积）定义为：

$$<\mathcal{A},\mathcal{B}>\ =\ \sum_{i_1}\sum_{i_2}\cdots\sum_{i_N}a_{i_1i_2\cdots i_N}b_{i_1i_2\cdots i_N}\qquad(1)$$

定义 4：一个张量 的 Frobenius 范数定义为：

$$\|\mathcal{A}\|\ =\ \sqrt{<\mathcal{A},\mathcal{A}>}\qquad(2)$$

定义 5：设有 N 阶张量 $\mathcal{A}\in\mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$，则 $\mathcal{A}$ 的 mode- k 矩阵（k 阶矩阵展开）$A_{(k)}\in\mathbb{R}^{I_k\times(I_{k+1}I_{k+2}\cdots I_NI_1\cdots I_{k-1})}$，其元素为 $a_{i_1i_2\cdots i_N}$。

例 1　对于三阶张量 $\mathcal{A}\in\mathbb{R}^{3\times2\times3}$，其中 $I_1=3$，$I_2=2$，$I_3=3$，我们有 $A_{(1)}\in\mathbb{R}^{I_1\times(I_2I_3)}$，$A_{(2)}\in\mathbb{R}^{I_2\times(I_3I_1)}$，$A_{(3)}\in\mathbb{R}^{I_3\times(I_1I_2)}$，且

$$A_{(1)}=\begin{pmatrix}a_{111}&a_{112}&a_{113}&a_{121}&a_{122}&a_{123}\\a_{211}&a_{212}&a_{213}&a_{221}&a_{222}&a_{223}\\a_{311}&a_{312}&a_{313}&a_{321}&a_{322}&a_{323}\end{pmatrix}_{3\times6},$$

$$A_{(2)}=\begin{pmatrix}a_{111}&a_{211}&a_{311}&a_{112}&a_{212}&a_{312}&a_{113}&a_{213}&a_{313}\\a_{121}&a_{221}&a_{321}&a_{122}&a_{222}&a_{322}&a_{123}&a_{223}&a_{323}\end{pmatrix}_{2\times9},$$

$$A_{(3)}=\begin{pmatrix}a_{111}&a_{121}&a_{211}&a_{221}&a_{311}&a_{321}\\a_{112}&a_{122}&a_{212}&a_{222}&a_{312}&a_{322}\\a_{113}&a_{123}&a_{213}&a_{223}&a_{313}&a_{323}\end{pmatrix}_{3\times6}$$

如上定义的 mode- k 矩阵（$k=1,\cdots,N$）的列向量称为张量 $\mathcal{A}$ 的 mode- k 向量。

定义 6：张量 $\mathcal{A}\in\mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ 和矩阵 $M\in\mathbb{R}^{J_n\times I_n}$ 的 n -mode 积用 $\mathcal{A}\times_n M$ 表示，它是一个 $I_1\times I_2\times\cdots\times I_{n-1}\times J_n\times I_{n+1}\times\cdots\times I_N$ 张量，且

$$(\mathcal{A}\times_n M)_{i_1\cdots i_{n-1}j_ni_{n+1}\cdots i_N}=\sum_{i_n}a_{i_1i_2\cdots i_N}m_{j_ni_n}\qquad(3)$$

张量和矩阵的 n -mode 积是两个矩阵乘积概念的推广，依据张量的 k 阶矩阵展开，有

$$B_{(n)} = MA_{(n)} \quad (4)$$

其中 $B_{(n)}$ 是张量 $\mathcal{B} = \mathcal{A} \times_n M$ 的 n 阶矩阵展开。

n -mode 积有下面的性质：

性质 1　给定张量 $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$，矩阵 $U \in \mathbb{R}^{J \times I_n}$，$V \in \mathbb{R}^{K \times I_m}$（$n \neq m$），则 $(\mathcal{A} \times_n U) \times_m V = (\mathcal{A} \times_m V) \times_n U$。

性质 2　给定张量 $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$，矩阵 $U \in \mathbb{R}^{J \times I_n}$，$V \in \mathbb{R}^{K \times J}$，则 $(\mathcal{A} \times_n U) \times_n V = \mathcal{A} \times_n (VU)$。

4 张量视角下的数据约简表示模型

4.1 张量场

张量场是几何学中很普通的概念，它应用于微分几何、流形理论等领域之中，是向量场概念的推广，具有与坐标系选择无关的性质（见图 1）。

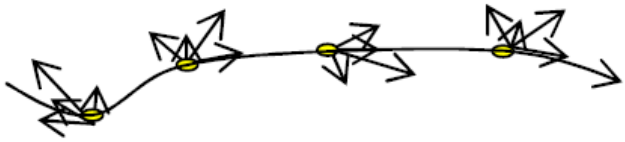

图 1 张量从一点到另一点的变化

在张量空间中建立一个坐标系，张量在坐标系中从一点到另一点变化（即张量变化过程中在相应点的秩保持不变）。对坐标系 (x^i)，$P \in \mathbb{R}^n$，有 $V_{i_1 i_2 \cdots i_N}(P) = V_{i_1 i_2 \cdots i_N}(x^1, x^2, \cdots, x^m)$，张量描述的对象的自然形式在任意坐标系中被保留。在不同的坐标系，一个张量的坐标可能不相同，但表示的是同一个张量，张量的不变性使得在张量场框架下处理张量数据成为可能。张量在新坐标系 $(x^{i'})$ 中的坐标与其在旧坐标系 (x^i) 中的坐标遵循下面的变换规则：

$$V_{i'_1 i'_2 \cdots i'_N}(P) = \frac{\partial x^{i_1}}{\partial x^{i'_1}}(P) \cdots \frac{\partial x^{i_k}}{\partial x^{i'_k}}(P) \cdot V_{i_1 i_2 \cdots i_k}(P) \quad (5)$$

定义 7：令 F_1、F_2 分别表示两个张量场，变换 $\tau: F_1 \to F_2$ 定义为张量场变换。

4.2 几种典型机器学习数据的张量表示模型

许多应用领域需要处理多通道数据，这些数据可以用张量的形式建立模型。

4.2.1 文论文档的张量表示模型

我们用一个简单的例子加以说明。考虑一个简单的文档：“tensor method”，用一个三阶张量 $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ 表示这个文档并用 26 个英文字母对它进行索引。我们约定标点、空格等 26 个英文字母以外的其他所有字符统一用“_”表示，将这个文档中的字符串作如下拆分：“ten”，“ens”，“nso”，“sor”，“or_”，“_me”，“met”，“eth”，“tho”，“hod”。分别通过“_”和从“a”到“z”的 26 个字母构建张量空间的三个坐标轴，它们顺次对应每个坐标轴的 0，1，2，…，26，这是一个 $27 \times 27 \times 27$ 张量。例如，“ten”的对应位置为（20，5，14），“or_”的对应位置为（15，18，0）。下一步采用 TFIDF[10] 权重计算方法给每个张量的对应位置加上权重并将它们作为张量的元素的值，如图 2 所示。

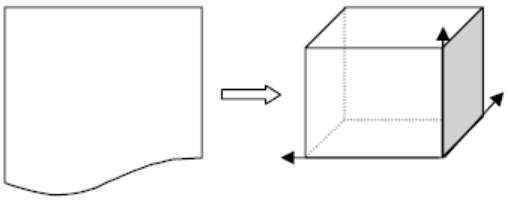

图 2 用字符级别的三阶张量表示一个文档

4.2.2 图像集合的张量表示模型

一个 $I \times J$ 的人脸图像集可视作具有潜在关联的二阶张量，张量数据对象不同的模表示该数据对象不同的“视角”（比如人脸图像不同的表情，光照条件等等）。同时，张量计算可用于对数据对象的操作。图 3 说明了如何将一个灰度图像集合表示成一个张量。

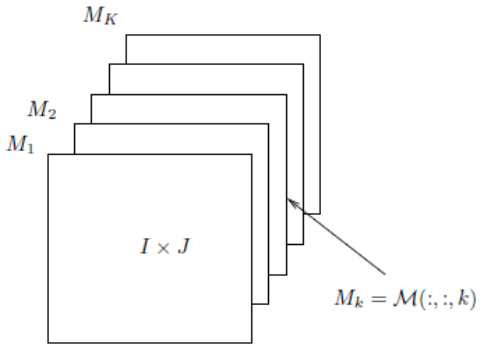

图 3 用一个 $I \times J \times K$ 张量 $\mathcal{M}$ 表示 K 个矩阵 $M_k \in \mathbb{R}^{I \times J}$ 集合

4.2.3 RGB 图像的二阶张量（矩阵）表示模型

我们用矩阵的形式表示 RGB 图像，矩阵表示模型可以直接操作颜色信息。

设 n 表示一个 $n_1 \times n_2$ 的 RGB 图像像素的数目，即 $n = n_1 \times n_2$。RGB 图像像素 3 个颜色通道的值 $(R,G,B)^T \in \mathbb{R}^3$，这里将 RGB 图像的像素定义为矩阵表示模型的基本单位，设 u_i 表示第 i 个基本单位（$1 \le i \le n$），则一幅 RGB 图像可表示为 $I = (u_1, u_2, \cdots, u_n)$，$u_i \in \mathbb{R}^m$ 表示 RGB 图像第 i 个像素的特征信息，m 是基本单位的维度。将 RGB 图像的第 i 个像素的 3 个颜色通道的值定义成表示模型的基本成分，令 $c_j^{(i)}$ 表示第 i 个基本单位的第 j 个基本成分（$1 \le j \le m$），则 $u_i = (c_1^{(i)}, c_2^{(i)}, \cdots, c_m^{(i)})^T$，所以 RGB 图像的矩阵表示模型为：

$$I = (u_1, u_2, \cdots, u_n) = \begin{pmatrix} c_1^{(1)} & c_1^{(2)} & \cdots & c_1^{(n)} \\ c_2^{(1)} & c_2^{(2)} & \cdots & c_2^{(n)} \\ \vdots & \vdots & & \vdots \\ c_m^{(1)} & c_m^{(2)} & \cdots & c_m^{(n)} \end{pmatrix}_{m \times n} \quad (6)$$

上述模型的基本成分可以是 RGB 图像像素的任何性质的值。特别地，如果基本单位的维数减为 1，矩阵表示模型将退化成一个向量。

4.3 约简模型

考虑一个数据集 $X=\{X_i\}$，通过研究 X 的特征得到 X 的张量信息 TF，我们用张量场的概念建立模型。为得到 X 的一个约简表示形式，将坐标系 α 中的 TF 变换到坐标系 β 中新的形式 TF'，这里建立模型 $<X, \varphi, TF, TF' \tau>$，其中，$X$ 表示输入数据集 $X=\{X_1, X_2, \cdots, X_n\}$；$TF$ 表示从数据集 X 提取出的张量场中的张量；TF' 表示经约简后的张量；φ 表示由数据集 X 提取出张量场 TF 的变换 $\varphi: X \to TF$；τ 表示数据压缩算法，τ 实际上是通过一个变换 $\tau: TF \to TF'$ 来实现的。

5 截断 HOSVD 算法（THOSVD）

5.1 算法提出的背景

推荐系统经常面对三路数据，一般包括三个类别：项目（item），用户（user），评分（rating），项目为需要推荐的东西，比如产品，电影，URLs 或者信息片段；用户为对项目进行评分的人或被推荐系统推荐项目的人；评分表达了用户对项目的情感态度，评分可以是二分类的（比如 Yes 或 No），也可以取整数（比如 1 星级到 5 星级）或者某个区间内连续的实数值。另外，还有一些隐式的反馈，仅记录用户与某个项目是否进行了交互。

推荐系统收集了大量的用户数据，将这些数据用一个三元组（u，i，r）（u，i，r 分别代表 user，item，rating）集合表示。通过这些数据构造一个张量，用 $\mathcal{A}$ 表示原始输入数据，重构张量 $\widetilde{\mathcal{A}}$ 表示输出数据，揭示了 users，items，ratings 之间的内在关联，$\widetilde{\mathcal{A}}$ 的元素可表示成一个四元组（u，i，r，p），p 表示用户 u 给项目 i 评 r 分的可能性（权重）。因而，可根据对二元组（u，i）评 r 分的权重的大小来判断是否应对用户 u 推荐项目 i。

5.2 多线性 SVD

定理 1：高阶奇异值分解（HOSVD）是矩阵 SVD 的推广：每一个 $I_1 \times I_2 \times \cdots \times I_N$ 张量 都可以写成 n -mode 积的形式：

$$\mathcal{A}=\mathcal{S} \times_1 U^{(1)} \times_2 U^{(2)} \times \cdots \times_N U^{(N)} \tag{7}$$

$U^{(n)}$ 为 $A_{(n)}$（张量 $\mathcal{A}$ 的 n -mode 矩阵展开）的标准正交的左奇异向量，$\mathcal{S}$ 称为核张量，$\mathcal{S}$ 具有全正交性，即对任意 n，α，β，$\alpha \neq \beta$，子张量 $\mathcal{S}_{i_n=\alpha}$ 和 $\mathcal{S}_{i_n=\beta}$ 都是正交的，$\mathcal{S}$ 起着控制 $U^{(n)}$ 之间相互关系的作用。同时，$\mathcal{A}$ 的 n -mode 奇异值 $\sigma_i^{(n)}=\left\|\mathcal{S}_{i_n=i}\right\|$ 以非递增的顺序排列：$\sigma_1^{(n)} \geq \sigma_2^{(n)} \geq \cdots \geq \sigma_{I_n}^{(n)} \geq 0$。

特别地，一个三阶张量 $\mathcal{A}$ 的 HOSVD 为：

$$\mathcal{A}=\mathcal{S} \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)} \tag{8}$$

三阶张量的奇异值分解见图 4。

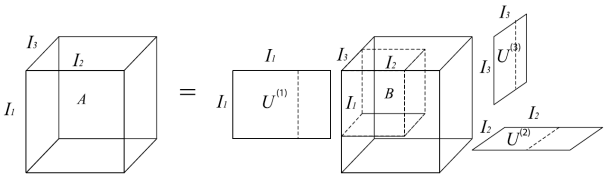

图 4 三阶张量的奇异值分解

5.3 截断 HOSVD 算法

在高阶奇异值分解原理的基础上，提出处理三路数据的 THOSVD 算法。

算法 1 THOSVD。

①由原始三路数据构造张量 $\mathcal{A}$，设用户、项目、评分的数量分别为 N_1、N_2、N_3，则 $\mathcal{A} \in \mathbb{R}^{N_1 \times N_2 \times N_3}$，每一个张量元素表征了给二元组（u，i）评 r 分的可能性（权重）大小；

②分别计算张量 $\mathcal{A}$ 的 3-mode 矩阵展开 $A_{(1)}$，$A_{(2)}$，$A_{(3)}$，$A_{(1)}$ 是通过固定项目和评分指标，改变用户指标而得到。$A_{(2)}$，$A_{(3)}$ 通过类似的方式计算，$A_{(1)} \in \mathbb{R}^{N_1 \times N_2 N_3}$，$A_{(2)} \in \mathbb{R}^{N_2 \times N_3 N_1}$，$A_{(3)} \in \mathbb{R}^{N_3 \times N_1 N_2}$；

③分别对 $A_{(1)}$，$A_{(2)}$，$A_{(3)}$ 作奇异值分解，将它们的左奇异矩阵分别记为 $U^{(1)}$，$U^{(2)}$，$U^{(3)}$；

④分别从 $U^{(1)}$，$U^{(2)}$，$U^{(3)}$ 中取出它们的前 n_1，n_2，n_3 列构造子矩阵 $U_{n_1}^{(1)}$，$U_{n_2}^{(2)}$，$U_{n_3}^{(3)}$，其中 $n_1 \in [1, N_1]$，$n_2 \in [1, N_2]$，$n_3 \in [1, N_3]$；

⑤构造核张量 $\mathcal{S}=\mathcal{A} \times_1 U_{n_1}^{(1)^T} \times_2 U_{n_2}^{(2)^T} \times_3 U_{n_3}^{(3)^T}$；

⑥重构原始张量 $\widetilde{\mathcal{A}}=\mathcal{S} \times_1 U_{n_1}^{(1)} \times_2 U_{n_2}^{(2)} \times_3 U_{n_3}^{(3)}$。

将压缩比定义为原始（未经压缩）张量 $\mathcal{A}$ 中总的元素个数和近似张量 $\widetilde{\mathcal{A}}$ 中总的元素个数之比。在 THOSVD 算法中，原始张量 $\mathcal{A} \in \mathbb{R}^{N_1 \times N_2 \times N_3}$，有 $N_1 \times N_2 \times N_3$ 个元素；核张量 $\mathcal{S} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$，有 $n_1 \times n_2 \times n_3$ 个元素；三个子矩阵 $U_{n_1}^{(1)} \in \mathbb{R}^{N_1 \times n_1}$，$U_{n_2}^{(2)} \in \mathbb{R}^{N_2 \times n_2}$，$U_{n_3}^{(3)} \in \mathbb{R}^{N_3 \times n_3}$ 分别有 $N_1 \times n_1$，$N_2 \times n_2$，$N_3 \times n_3$ 个元素。此时压缩比（Compression Ratio，CR）：

$$CR=\frac{N_1 N_2 N_3}{n_1 n_2 n_3+N_1 n_1+N_2 n_2+N_3 n_3} \tag{9}$$

其中 n_1，n_2，n_3 分别为矩阵 $U^{(1)}$，$U^{(2)}$，$U^{(3)}$ 中保留的“最重要”的列数。

重构数据关于原始数据的 RMSE（Root Mean Square Error）通过下式得到：

$$RMSE=\sqrt{\frac{1}{N_3}\left\|\mathcal{A}-\widetilde{\mathcal{A}}\right\|^2} \tag{10}$$

RMSE 是衡量算法性能的重要指标，RMSE 越小，精确度越高。

下面给出一个算法求解的例子。

通过 7 个用户对 5 个项目给出的 4 个评分构建三阶张量 $\mathcal{A}$（$7 \times 5 \times 4$）。为便于理解，我们设定权重的初始值具有二值性，即一个用户对一个项目进行了评分，那么这个评分

能“百分之百”代表他（她）个人的意志，不存在含糊，权重初值为 1（表示该用户对该项目的评分是确信无疑的）或 0（表示该用户对该项目未进行该评分），如表 1 所示。

表 1 原始三阶张量数据

序号	用户	项目	评分	权重
1	U1	I1	R1	1
2	U1	I2	R4	1
3	U1	I5	R2	1
4	U2	I3	R1	1
5	U3	I1	R3	1
6	U3	I4	R3	1
7	U4	I4	R2	1
8	U5	I1	R1	1
9	U5	I2	R1	1
10	U6	I5	R1	1
11	U7	I1	R1	1
12	U7	I5	R4	1

执行 THOSVD 算法后，得到重构张量 $\widetilde{\mathcal{A}}$，如表 2 所示，我们得到 19 组关于用户、项目、评分之间的新的关联。

表 2 重构三阶张量的新增数据

序号	用户	项目	评分	权重
1	U1	I2	R1	0.5000
2	U2	I2	R4	0.9967
3	U2	I3	R4	0.9967
4	U2	I4	R1	0.0015
5	U2	I5	R3	0.0060
6	U3	I1	R1	0.0406
7	U3	I2	R1	0.0406
8	U3	I3	R1	0.2225
9	U3	I3	R2	0.0015
10	U4	I1	R1	0.0037
11	U5	I1	R3	1.0000
12	U6	I1	R3	0.5000
13	U6	I2	R3	0.5000
14	U6	I2	R4	0.5033
15	U6	I3	R4	0.5033
16	U6	I4	R1	0.2221
17	U7	I3	R1	0.0225
18	U7	I3	R2	0.0221
19	U7	I4	R4	0.3594

从原始张量 $\mathcal{A}$ 无法得知 U1 对 I2 评分为 R1 的情况，但从表 2 第 1 行数据可知 $\widetilde{\mathcal{A}}$ 的元素（U1，I2，R1）为 0.5000。除此之外，U1 再无给其他项目评过分，据此可初步判断 U1 对 I2 较为关注。同样从表 2 第 2、3、4、5 行可知 U2 对 I2、I3、I4、I5 的评分分别为 R4、R4、R1、R3 的权重为 0.9967、0.9967、0.0015、0.0060，U2 对 I2，I3 评分为 R4 的权重相当且都比较大，U2 对 I2，I3 几乎具有相同的偏好程度，U2 对 I4，I5 评分分别为 R1，R3 的权重都比较小，几乎可以忽略不计，其余行的数据可类似地分析。事实上，可以根据实际问题为权重设定一个阈值，将小于阈值的权重舍去，因为其不能反映用户对项目的评分状况，这样可以减少噪声，提高输出数据的有效性。

分别利用（9）式和（10）式计算 THOSVD 算法的压缩比为 1.5385，均方根误差为 1.1452，所以本算法能够实现高阶张量数据的压缩且能保证一定的准确度。THOSVD 算法能够挖掘高阶数据对象：用户，项目，评分之间的内在关联，这在提升推荐系统效率方面具有重要意义。

6 结语

论文为张量视角下的典型的机器学习数据构建了一般意义下的约简表示模型，张量是描述多路数据良好的工具，基于张量方法的数据挖掘算法有较大的应用潜力。

我们针对推荐系统典型的三路数据给出了一般的 THOSVD 算法，旨在挖掘高维数据潜在 的语义信息并实现数据降维。后期的工作我们将研究设计一个自动化的算法，这个算法可以自适应地调节核张量的维数，达到既满足特定的精度要求又能实现维数约简的目的。现实的推荐系统数据结构较为复杂，论文的算法对于处理三阶以上的张量数据亦有借鉴意义。

参考文献

[1] Turk M, Pentland A. Eigenfaces for recognition[J]. J Cogn Neurosci,1991,3(1):71-86.

[2] Roweis S T, Saul L K. Nonlinear Dimensionality Reduction by Locally Linear Embedding[J]. Science, 2000,290(5500):2323-2326.

[3] Silva V D, Tenenbaum J B. Global versus local methods in nonlinear dimensionality reduction[C]//Advances in neural information processing systems,2003.

[4] Tenenbaum J B, Silva V D, Langford J C. A Global Geometric Framework for Nonlinear Dimensionality Reduction[J]. Scien ce,2000,290(5500):2319-2323.

[5] Vasilescu M A O, Terzopoulos D. Multilinear subspace analysis of image ensembles[C]//Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on. IEEE, 2003.