基于加速无约束张量隐因子分解模型的 Web 服务 QoS 估计

林铭炜[1,2]，李文强[1,2]，许秀琴[3]，刘健[1,2]

（1. 福建师范大学福建省公共服务大数据挖掘与应用工程技术研究中心，福建 福州 350117；
2. 福建师范大学计算机与网络空间安全学院，福建 福州 350117；
3. 福建师范大学数学与统计学院，福建 福州 350117）

摘 要：针对基于张量非负隐因子分解模型的 Web 服务 QoS 估计方法过于依赖非负初始随机数据以及特意设计的非负训练方法，导致模型的兼容性和扩展性不高的问题，提出了加速无约束张量隐因子分解模型。其主要思想包括三部分：将非负性约束从决策参数转移到输出的隐因子，并通过单元素映射函数连接它们；运用结合动量方法的随机梯度下降算法，有效提高模型的收敛速度与估计精度；给出加速无约束张量隐因子分解模型的详细算法和结果分析。在实际工业应用中的 2 个动态 QoS 数据集上的实证研究表明，与最先进的 QoS 估计模型相比，所提模型具有较高的计算效率和估计精度。

关键词：服务质量；隐因子分解分析；张量非负隐因子分解模型；无约束非负；动量方法

中图分类号：TN92

文献标志码：A

DOI: 10.11959/j.issn.1000−436x.2024064

Accelerated unconstrained latent factorization of tensor model for Web service QoS estimation

LIN Mingwei[1, 2], LI Wenqiang[1, 2], XU Xiuqin[3], LIU Jian[1, 2]

1. Fujian Provincial Engineering Research Center for Public Service Big Data Mining and Application, Fujian Normal University, Fuzhou 350117, China
2. College of Computer and Cyber Security, Fujian Normal University, Fuzhou 350117, China
3. School of Mathematics and Statistics, Fujian Normal University, Fuzhou 350117, China

Abstract: Aiming at the problem that the Web service quality of service (QoS) estimation methods based on the non-negative latent factorization of tensor model (NLFT) depend heavily on non-negative initial random data and specially designed non-negative training schemes, which lead to low compatibility and scalability, an accelerated unconstrained latent factorization of tensor (AULFT) model was proposed. The proposed model consisted of three main parts. The non-negative constraints from decision parameters were transferred to output latent factors and they were connected through the single-element-dependent mapping function. A momentum-incorporated stochastic gradient descent (MSGD) algorithm was used to effectively improve the convergence rate and estimation accuracy of the proposed AULFT model. The detailed algorithm and result analysis of the proposed AULFT model were presented. The empirical study on two dynamic QoS datasets in real industrial applications demonstrates that the proposed AULFT model has higher computational efficiency and estimation accuracy than the state-of-the-art QoS estimation models.

Keywords: quality of service, latent factorization analysis, non-negative latent factorization of tensor model, unconstrained non-negative, momentum method

---

收稿日期：2023−10−11；**修回日期：**2023−11−21

基金项目：国家自然科学基金资助项目（No.62272103）；福建省自然科学基金杰青项目资助项目（No.2022J06020）；福建省“雏鹰计划”青年拔尖人才计划基金资助项目（No.F21E0011202B01）

Foundation Items: The National Natural Science Foundation of China (No.62272103), Distinguished Young Project of Natural Science Foundation of Fujian Province (No.2022J06020), The Young Top Talent of Young Eagle Program of Fujian Province (No.F21E0011202B01)

0　引言

随着新兴互联网技术的迅速发展[1-2]，服务供应商提供了海量的 Web 服务并发布到互联网上供用户调用[3]。许多 Web 服务虽然功能类似，但它们的服务质量、成本、价格却不尽相同[4-5]。如何帮助用户从大量功能相似的 Web 服务中找到满足其需求且性价比最高的服务是 Web 服务推荐中亟待解决的重要问题[6]。

Web服务的非功能特性在为目标用户推荐合适的服务中发挥着重要的作用。非功能特性统称为服务质量（QoS, quality of service），通常以响应时间、吞吐量、调用失败率、容量、鲁棒性、可用性等进行衡量，其作用是确保终端应用程序的可靠性[7]。基于 QoS 数据的 Web 服务推荐，旨在从众多具有相似功能的服务中挑选出满足用户需求的 Web 服务，目前已成为服务计算领域最热门的研究方向之一。通常，QoS 数据可以通过预热测试[8-9]获得。但调用相应的业务服务通常需要大量的费用，并且评估所有候选服务相当耗时。此外，随着 Web 服务数量的爆炸性增长，许多服务的 QoS 数据并未被观测到。在这种情况下，QoS 估计在获取完整的 QoS 数据中起着至关重要的作用[10-11]，并且能够为 Web 服务推荐提供有效的数据支撑。

QoS 估计方法可大致分为两类：静态与动态方法。在静态方法中，由于用户不可能调用互联网上所有的 Web 服务，因此执行 QoS 估计的输入数据形式是稀疏 QoS 矩阵[12]。该类矩阵中的每行代表一个用户，每列表示一项服务，矩阵中的每个元素表示某个用户调用某项服务的记录。QoS 估计的原理是根据已知的观测值来估计缺失的数据[13]。在众多静态的 QoS 估计方法中，基于隐因子分析的方法由于高效率和易实现性而被广泛研究和采用[14-18]。基于隐因子分析的QoS估计方法旨在从给定稀疏QoS矩阵中训练 2 个隐因子矩阵，通过最小化已知值和估计值之间的差异进行低秩近似。尽管各种基于隐因子分析的 QoS 估计方法在模型设计上不同，但它们存在一个共同的局限性：没有考虑 QoS 数据的时间变化特性。

动态 QoS 估计方法打破了这一局限性，其输入数据的形式是高维不完备的 QoS 张量。与稀疏 QoS 矩阵相比，高维不完备的 QoS 张量多了一个时间维度以描述 QoS 数据的时间变化特性。在对高维不完备 QoS 数据的时间动态特性进行建模的方法中，张量非负隐因子分解（NLFT, non-negative latent factorization of tensor）模型因能够有效地实现时变感知的 QoS 估计[19-20]，而受到业界广泛关注。Luo 等[19]提出了张量非负隐因子分解模型以实现高度准确的 QoS 估计，该模型使用线性偏差来刻画 QoS 数据随时间变化的波动性，并且对模型进行非负性约束以描述 QoS 数据的非负性。Chen 等[20]提出基于非负乘法更新算法的有偏非负隐因子分解模型，与广义 Nesterov 的加速梯度法相结合，从而提高了模型的收敛速度。虽然上述张量非负隐因子分解模型对 QoS 数据有较高的估计精度，但由于该类模型过分地依赖非负初始随机数据以及特意设计的非负训练方法，模型的兼容性和扩展性不高[21]。

为了解决张量非负隐因子分解模型存在的局限性，本文提出了一种加速无约束张量隐因子分解（AULFT, accelerated unconstrained latent factorization of tensor）模型。其主要思想包括三部分：1) 将非负性约束从决策参数转移到输出的隐因子，并通过单元素映射函数连接它们；2) 运用结合动量方法的随机梯度下降算法，有效提高模型的收敛速度与估计精度；3) 给出加速无约束张量隐因子分解模型的详细算法和结果分析。

1　相关工作

1.1　基本符号表示

在执行时变感知 QoS 分析和估计时，通常将用户–服务–时间张量当作基本输入数据源。本文所用符号及其含义如表 1 所示。由于 QoS 数据定义在非负实数域上且包含了众多未被观测到的数据，因此目标张量 T 通常是高维不完备的，如图 1 所示，其中，白色格子表示未被观测到的数据，灰色格子表示观测到的数据。首先定义本文的目标张量 T 。

定义 1　高维不完备的用户–服务–时间张量。I 、J 和 K 分别表示 QoS 数据中用户、服务和时间的集合，定义用户–服务–时间的目标张量为 $T^{|I|\times|J|\times|K|}$，张量 T 中的每个元素 t_{ijk} 表示第 i 个用户在 k 时刻调用第 j 个 Web 服务所产生的 QoS 数据，其中，$i\in I$，$j\in J$，$k\in K$。Λ 表示张量 T 中已知元素集合，Γ 表示未知元素集合。当 $|\Lambda|\ll|\Gamma|$ 时，则称该张量是高维不完备的。

表 1 本文所用符号及其含义

符号	含义
$T^{\lvert I\rvert\times\lvert J\rvert\times\lvert K\rvert}$	用户–服务–时间的高维不完备目标张量
$\boldsymbol{U}$、$\boldsymbol{S}$、$\boldsymbol{W}$	用户、服务以及时间节点的隐因子矩阵
$\boldsymbol{Q}_{(U)}$、$\boldsymbol{Q}_{(S)}$、$\boldsymbol{Q}_{(W)}$	隐因子矩阵 $\boldsymbol{U}$、$\boldsymbol{S}$、$\boldsymbol{W}$ 对应的决策参数矩阵
$\boldsymbol{V}_{\boldsymbol{Q}_{(U)}}$、$\boldsymbol{V}_{\boldsymbol{Q}_{(S)}}$、$\boldsymbol{V}_{\boldsymbol{Q}_{(W)}}$	更新 $\boldsymbol{Q}_{(U)}$、$\boldsymbol{Q}_{(S)}$、$\boldsymbol{Q}_{(W)}$ 的速度矩阵
I、J、K	QoS 数据中用户、服务以及时间的集合
$\hat{T}^{\lvert I\rvert\times\lvert J\rvert\times\lvert K\rvert}$	目标张量 T 的近似估计张量
t_{ijk}	T 中的单个元素
$\hat{t}_{ijk}$	$\hat{T}$ 中的单个元素
R	$\hat{T}$ 的秩（即隐因子矩阵的维度）
$\boldsymbol{u}_r$、$\boldsymbol{s}_r$、$\boldsymbol{w}_r$	$\boldsymbol{U}$、$\boldsymbol{S}$、$\boldsymbol{W}$ 中的隐因子向量
u_{ir}、s_{jr}、w_{kr}	$\boldsymbol{U}$、$\boldsymbol{S}$、$\boldsymbol{W}$ 中的单个隐因子
$Q_{(\boldsymbol{U})ir}$、$Q_{(\boldsymbol{S})jr}$、$Q_{(\boldsymbol{W})kr}$	$\boldsymbol{Q}_{(U)}$、$\boldsymbol{Q}_{(S)}$、$\boldsymbol{Q}_{(W)}$ 中的单个元素
Λ	T 中已知元素集合
Γ	T 中未知元素集合

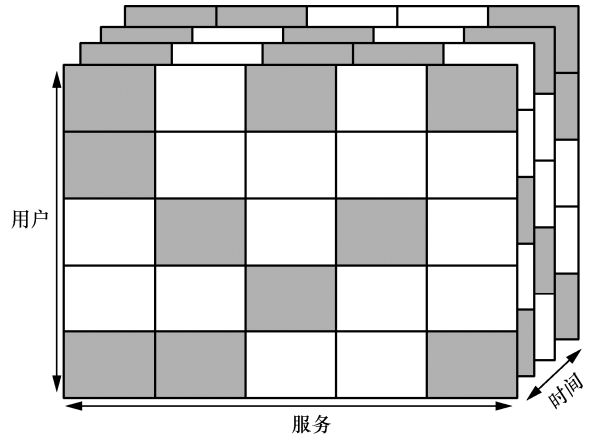


图 1 用户–服务–时间的高维不完备张量

1.2 问题表述

本文采用正则多元张量分解（CPTF, canonical polyadic tensor factorization）法[22-23]分解目标张量 T，以获取 T 中的隐因子。目标张量 T 被 CPTF 分解成 R 个秩一张量 $X_1, X_2, X_3, \cdots, X_R$。其中，$R$ 是近似估计张量 $\hat{T}$ 的秩。

定义 2 秩一张量。$X_r^{\lvert I\rvert\times\lvert J\rvert\times\lvert K\rvert}$ 表示一个秩一张量。每个秩一张量由 3 个隐因子向量 $\boldsymbol{u}_r$、$\boldsymbol{s}_r$、$\boldsymbol{w}_r$ 进行外积得到，即 $X_r=\boldsymbol{u}_r\circ\boldsymbol{s}_r\circ\boldsymbol{w}_r$，其中 $r\in R$。隐因子向量 $\boldsymbol{u}_r$、$\boldsymbol{s}_r$、$\boldsymbol{w}_r$ 的长度分别是 $\lvert I\rvert$、$\lvert J\rvert$、$\lvert K\rvert$，通过展开它们的外积，得到秩一张量 X_r 中单个元素 $x_{ijk}^{(r)}$ 的计算式为

$$x_{ijk}^{(r)}=u_{ir}s_{jr}w_{kr} \tag{1}$$

如图 2 所示，R 个长度分别为 $\lvert I\rvert$、$\lvert J\rvert$、$\lvert K\rvert$ 的隐因子向量各自集合成了相对应的 3 个隐因子矩阵 $\boldsymbol{U}^{\lvert I\rvert\times R}$、$\boldsymbol{S}^{\lvert J\rvert\times R}$、$\boldsymbol{W}^{\lvert K\rvert\times R}$。3 个隐因子矩阵的第 r 个列向量耦合成第 r 个秩一张量 X_r。最后，将这 R 个秩一张量累加得到近似估计张量 $\hat{T}$，表达式如下

$$\hat{T}=\sum_{r=1}^{R}X_r \tag{2}$$

其中，近似估计张量 $\hat{T}$ 中的单个元素为

$$\hat{t}_{ijk}=\sum_{r=1}^{R}u_{ir}s_{jr}w_{kr} \tag{3}$$

为了得到期望的 3 个隐因子矩阵 $\boldsymbol{U}^{\lvert I\rvert\times R}$、$\boldsymbol{S}^{\lvert J\rvert\times R}$、$\boldsymbol{W}^{\lvert K\rvert\times R}$，本文基于已知数据集合 Λ，采用隐因子分析中常用的欧几里得距离公式构建目标函数 ε，来测量目标张量 T 与近似估计张量 $\hat{T}$ 之间的差距。ε 表示为

$$\varepsilon=\frac{1}{2}\sum_{t_{ijk}\in\Lambda}(t_{ijk}-\hat{t}_{ijk})^2 \tag{4}$$

为了准确地描述时变感知 QoS 数据的非负性，对张量隐因子分解模型的决策参数进行非负性约束是非常有必要的。本文将式(3)代入式(4)，并对目标函数加上非负性约束，可将式(4)转换为

$$\begin{aligned}&\varepsilon=\frac{1}{2}\sum_{t_{ijk}\in\Lambda}\left(t_{ijk}-\sum_{r=1}^{R}u_{ir}s_{jr}w_{kr}\right)^2\\&\text{s.t. }\forall i\in I,\ j\in J,\ k\in K,\ r\in\{1,2,3,\cdots,R\},\\&\quad u_{ir}\geqslant 0,\ s_{jr}\geqslant 0,\ w_{kr}\geqslant 0\end{aligned} \tag{5}$$

由于 T 中已知数据分布不平衡以及对 $\boldsymbol{U}$、$\boldsymbol{S}$、$\boldsymbol{W}$ 的初始假设敏感，式(5)是不适定的。因此，对式(5)进行正则化防止所得的模型过拟合[24-26]。本文利用 Tikhonov 正则化，将式(5)转换为

$$\begin{aligned}&\varepsilon=\frac{1}{2}\sum_{t_{ijk}\in\Lambda}\left(\left(t_{ijk}-\sum_{r=1}^{R}u_{ir}s_{jr}w_{kr}\right)^2+\lambda_u\sum_{r=1}^{R}u_{ir}^2+\right.\\&\left.\lambda_s\sum_{r=1}^{R}s_{jr}^2+\lambda_w\sum_{r=1}^{R}w_{kr}^2\right)\\&\text{s.t. }\forall i\in I,\ j\in J,\ k\in K,\ r\in\{1,2,3,\cdots,R\},\\&\quad u_{ir}\geqslant 0,\ s_{jr}\geqslant 0,\ w_{kr}\geqslant 0\end{aligned} \tag{6}$$

其中，λ_u、λ_s、λ_w 分别是 $\boldsymbol{U}$、$\boldsymbol{S}$、$\boldsymbol{W}$ 的正则化系数。

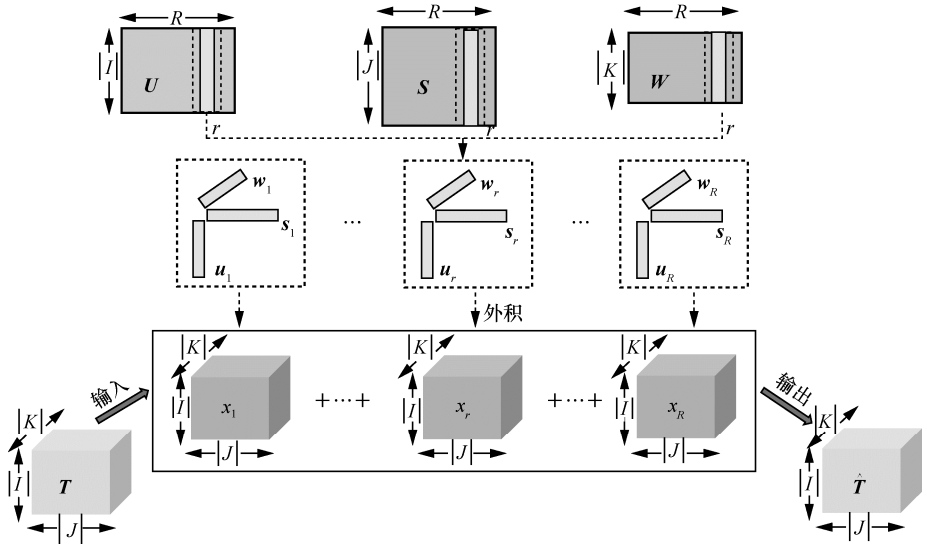


图 2　CPTF 中的隐因子矩阵和秩一张量

2　加速无约束张量隐因子分解模型

2.1　非负性约束消除法

非负性约束消除法过程如图 3 所示。本文针对 3 个隐因子矩阵 $\boldsymbol{U}$、$\boldsymbol{S}$、$\boldsymbol{W}$ 分别对应设置了 3 个决策参数矩阵 $\boldsymbol{Q}_{(\boldsymbol{U})}^{|I|\times R}$、$\boldsymbol{Q}_{(\boldsymbol{S})}^{|J|\times R}$、$\boldsymbol{Q}_{(\boldsymbol{W})}^{|K|\times R}$。接着将非负性约束从决策参数转移到输出的隐因子，并通过单元素的映射函数 f 连接它们，如式(7)所示

$$u_{ir}=f(Q_{(\boldsymbol{U})ir}),s_{jr}=f(Q_{(\boldsymbol{S})jr}),w_{kr}=f(Q_{(\boldsymbol{W})kr})$$
$$\text{s.t. } \forall i\in I,j\in J,k\in K,r\in\{1,2,3,\cdots,R\} \tag{7}$$

其中，$Q_{(\boldsymbol{U})ir}$、$Q_{(\boldsymbol{S})jr}$、$Q_{(\boldsymbol{W})kr}$ 分别表示决策参数矩阵 $\boldsymbol{Q}_{(\boldsymbol{U})}$、$\boldsymbol{Q}_{(\boldsymbol{S})}$、$\boldsymbol{Q}_{(\boldsymbol{W})}$ 中的单个元素。

基于之前的研究工作[27]，映射函数 f 须满足以下条件

$$\forall x\in\mathbb{R}:\begin{cases}y=f(x)\geqslant 0\\ \exists x=f^{-1}(y)\\ f'(x)\neq 0\end{cases} \tag{8}$$

本文采用绝对值函数作为映射函数 f 且满足式(8)中的所有条件，绝对值函数及其导数形式为

$$f(a)=|a|,f'(a)=\begin{cases}1, & a>0\\ -1, & a\leqslant 0\end{cases} \tag{9}$$

注意，当 a=0 时，绝对值函数的导数是不存在的。本文给出如下设置：当 a=0 时，令 $f'(a)=-1$。

$\hat{T}$ 中的元素由式(3)得到，由于 f 的特性是输入端符号自由且保持输出端非负，那么通过式(7)便可确保 $\hat{T}$ 中每个元素是非负的，从而保证了 QoS 估计

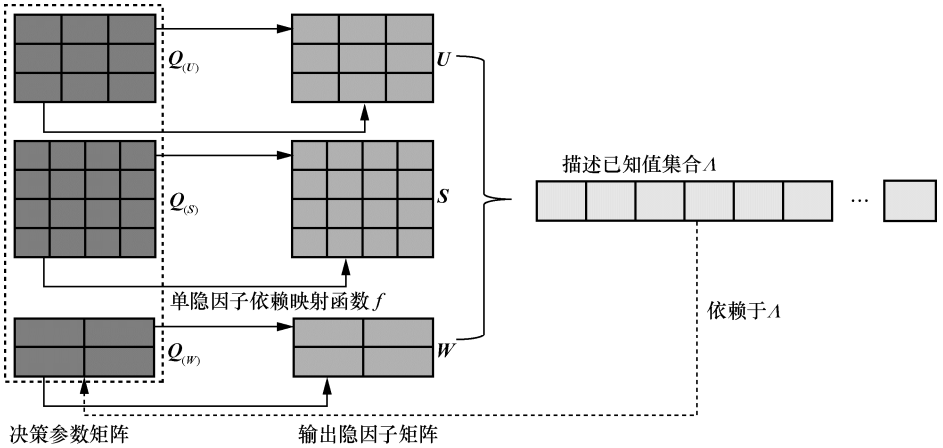


图 3　非负性约束消除法过程

值的非负性并且消除了损失函数 ε 的非负性约束。相对应地，损失函数式(6)变换为

$$\varepsilon=\frac{1}{2}\sum_{t_{ijk}\in\Lambda}\left((t_{ijk}-\hat{t}_{ijk})^2+\lambda_{\boldsymbol{Q}_{(U)}}\sum_{r=1}^{R}f(Q_{(\boldsymbol{U})ir})^2+\lambda_{\boldsymbol{Q}_{(S)}}\sum_{r=1}^{R}f(Q_{(\boldsymbol{S})jr})^2+\lambda_{Q_{(W)}}\sum_{r=1}^{R}f(Q_{(\boldsymbol{W})kr})^2\right)$$
$$\text{s.t. }\forall i\in I,\ j\in J,\ k\in K,\ r\in\{1,2,3,\cdots,R\} \tag{10}$$

其中，$\lambda_{\boldsymbol{Q}_{(U)}}$、$\lambda_{\boldsymbol{Q}_{(S)}}$、$\lambda_{\boldsymbol{Q}_{(W)}}$ 分别是 $\boldsymbol{Q}_{(U)}$、$\boldsymbol{Q}_{(S)}$、$\boldsymbol{Q}_{(W)}$ 的正则化系数。

2.2 基于随机梯度下降算法的张量隐因子分解模型

如文献[22-23]所述，在隐因子分解中采用随机梯度下降算法具有计算复杂度较低和易于实现的特点。因此，本文采用随机梯度下降算法对目标函数式(10)进行最小化，其具体更新规则如下

$$\arg\min\varepsilon(\boldsymbol{Q}_{(U)},\boldsymbol{Q}_{(S)},\boldsymbol{Q}_{(W)})\overset{\text{SGD}}{\Rightarrow}\forall i\in I,j\in J,k\in K,r\in\{1,2,3,\cdots,R\}$$
$$Q^{t}_{(\boldsymbol{U})ir}\leftarrow Q^{t-1}_{(\boldsymbol{U})ir}-\eta_{ir}\frac{\partial\varepsilon^{t-1}_{ijk}}{\partial Q^{t-1}_{(\boldsymbol{U})ir}}$$
$$Q^{t}_{(\boldsymbol{S})jr}\leftarrow Q^{t-1}_{(\boldsymbol{S})ir}-\eta_{jr}\frac{\partial\varepsilon^{t-1}_{ijk}}{\partial Q^{t-1}_{(\boldsymbol{S})jr}}$$
$$Q^{t}_{(\boldsymbol{W})kr}\leftarrow Q^{t-1}_{(\boldsymbol{W})kr}-\eta_{kr}\frac{\partial\varepsilon^{t-1}_{ijk}}{\partial Q^{t-1}_{(\boldsymbol{W})kr}} \tag{11}$$

其中，$\varepsilon_{ijk}=\frac{1}{2}\left((t_{ijk}-\hat{t}_{ijk})^2+\lambda_{\boldsymbol{Q}_{(U)}}\sum_{r=1}^{R}f(Q_{(\boldsymbol{U})ir})^2+\lambda_{\boldsymbol{Q}_{(S)}}\sum_{r=1}^{R}f(Q_{(\boldsymbol{S})jr})^2+\lambda_{\boldsymbol{Q}_{(W)}}\sum_{r=1}^{R}f(Q_{(\boldsymbol{W})kr})^2\right)$ 表示每个训练实例 $t_{ijk}\in\Lambda$ 上的瞬时损失；t 表示第 t 轮更新，η_{ir}、η_{jr}、η_{kr} 分别表示 $Q_{(\boldsymbol{U})ir}$、$Q_{(\boldsymbol{S})jr}$、$Q_{(\boldsymbol{W})kr}$ 的学习率。式(11)中的随机梯度为

$$\frac{\partial\varepsilon^{t-1}_{ijk}}{\partial Q^{t-1}_{(\boldsymbol{U})ir}}=f'\left(Q^{t-1}_{(\boldsymbol{U})ir}\right)\left(\lambda_{\boldsymbol{Q}_{(U)}}f(Q^{t-1}_{(\boldsymbol{U})ir})-(t_{ijk}-\hat{t}_{ijk})f(Q^{t-1}_{(\boldsymbol{S})jr})f(Q^{t-1}_{(\boldsymbol{W})kr})\right)$$
$$\frac{\partial\varepsilon^{t-1}_{ijk}}{\partial Q^{t-1}_{(\boldsymbol{S})jr}}=f'\left(Q^{t-1}_{(\boldsymbol{S})jr}\right)\left(\lambda_{\boldsymbol{Q}_{(S)}}f(Q^{t-1}_{(\boldsymbol{S})jr})-(t_{ijk}-\hat{t}_{ijk})f(Q^{t-1}_{(\boldsymbol{U})ir})f(Q^{t-1}_{(\boldsymbol{W})kr})\right)$$
$$\frac{\partial\varepsilon^{t-1}_{\text{ijk}}}{\partial Q^{t-1}_{(\boldsymbol{W})kr}}=f'\left(Q^{t-1}_{(\boldsymbol{W})kr}\right)\left(\lambda_{\boldsymbol{Q}_{(W)}}f(Q^{t-1}_{(\boldsymbol{W})kr})-(t_{\text{ijk}}-\hat{t}_{ijk})f(Q^{t-1}_{(\boldsymbol{U})ir})f(Q^{t-1}_{(\boldsymbol{S})jr})\right) \tag{12}$$

将式(12)代入式(11)，可得张量隐因子分解模型中随机梯度下降算法的更新规则，具体如式(13)所示。

$$(\boldsymbol{Q}_{(U)},\boldsymbol{Q}_{(S)},\boldsymbol{Q}_{(W)})=\arg\min_{\boldsymbol{Q}_{(U)},\boldsymbol{Q}_{(S)},\boldsymbol{Q}_{(W)}}\varepsilon\overset{\text{SGD}}{\Rightarrow}$$
$$\begin{cases}Q^{t}_{(\boldsymbol{U})ir}\leftarrow Q^{t-1}_{(\boldsymbol{U})ir}-\eta_{ir}f'(Q^{t-1}_{(\boldsymbol{U})ir})\cdot\\ \left(\lambda_{\boldsymbol{Q}_{(U)}}f(Q^{t-1}_{(\boldsymbol{U})ir})-\left(t_{ijk}-\hat{t}_{ijk}\right)\cdot\right.\\ \left.f(Q^{t-1}_{(\boldsymbol{S})jr})f(Q^{t-1}_{(\boldsymbol{W})kr})\right),\\ Q^{t}_{(\boldsymbol{S})jr}\leftarrow Q^{t-1}_{(\boldsymbol{S})jr}-\eta_{jr}f'(Q^{t-1}_{(\boldsymbol{S})jr})\cdot\\ \left(\lambda_{\boldsymbol{Q}_{(S)}}f(Q^{t-1}_{(\boldsymbol{S})jr})-(t_{ijk}-\hat{t}_{ijk})\cdot\right.\\ \left.f(Q^{t-1}_{(\boldsymbol{U})ir})f(Q^{t-1}_{(\boldsymbol{W})kr})\right),\\ Q^{t}_{(\boldsymbol{W})kr}\leftarrow Q^{t-1}_{(\boldsymbol{W})kr}-\eta_{kr}f'(Q^{t-1}_{(\boldsymbol{W})kr})\cdot\\ \left(\lambda_{\boldsymbol{Q}_{(W)}}f(Q^{t-1}_{(\boldsymbol{W})kr})-(t_{ijk}-\hat{t}_{ijk})\cdot\right.\\ \left.f(Q^{t-1}_{(\boldsymbol{U})ir})f(Q^{t-1}_{(\boldsymbol{S})jr})\right)\end{cases} \tag{13}$$

2.3 动量方法

动量方法是一种加速随机梯度下降算法收敛的学习策略，其原理如图 4 所示。给定一个目标函数 $H(\theta)$，结合动量方法的学习算法会先记录决策参数 θ 经过上一轮更新后的状态，并将该状态和当前梯度进行线性组合以确定本轮更新的方向。动量系数 γ 作用于每轮训练更新过程中平衡先前更新速度向量和当前更新梯度。

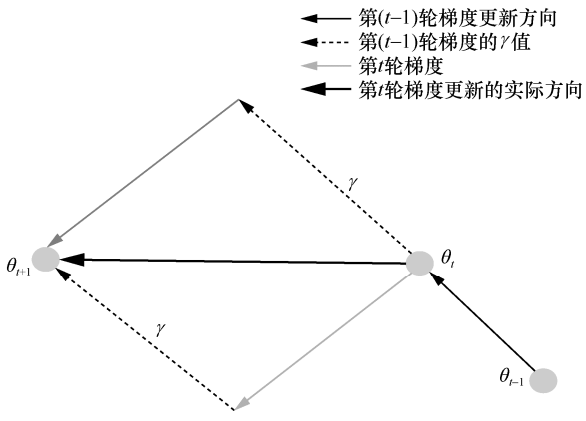


图 4 动量方法原理

上述动量方法以式(14)为更新决策参数θ的规则

$$\begin{aligned} &v_0 = 0 \\ &\boldsymbol{v}_t = \gamma \boldsymbol{v}_{t-1} + \eta \nabla_\theta H(\theta_{t-1}, q_t) \\ &\theta_t = \theta_{t-1} + v_t \end{aligned} \tag{14}$$

其中，v_0表示速度向量的初始状态，一般将其初始状态设置为0，$\boldsymbol{v}_t$和$\boldsymbol{v}_{t-1}$分别表示模型在第t轮更新以及第$t-1$轮更新的速度向量，q_t表示在第t轮更新遇到的训练实例。

2.4　加速无约束张量隐因子分解模型

为了加速张量隐因子分解模型求解的收敛速度，本文对随机梯度下降算法进行改进。基于式(14)，结合动量方法的随机梯度下降算法的更新规则如式(15)～式(17)所示。

$$\begin{aligned} &v^0_{\mathcal{Q}_{(\boldsymbol{U})ir}} = 0 \\ &v^t_{\mathcal{Q}_{(\boldsymbol{U})ir}} = \gamma v^{t-1}_{\boldsymbol{Q}_{(\boldsymbol{U})ir}} + \eta ir \frac{\partial \varepsilon^{t-1}_{ijk}}{\partial Q^{t-1}_{(\boldsymbol{U})ir}} \\ &Q^t_{(\boldsymbol{U})ir} = Q^{t-1}_{(\boldsymbol{U})ir} - v^t_{\boldsymbol{Q}_{(\boldsymbol{U})ir}} \end{aligned} \tag{15}$$

其中，γ是调整动量效应的常量系数，$v^t_{\mathcal{Q}_{(\boldsymbol{U})ir}}$是$\mathcal{Q}_{(\boldsymbol{U})ir}$第$t$轮更新的速度向量。根据随机梯度下降算法的原理，速度参数$v^t_{\mathcal{Q}_{(\boldsymbol{U})ir}}$是基于每个训练实例来更新的。类似地，$\mathcal{Q}_{(\boldsymbol{S})jr}$和$\mathcal{Q}_{(\boldsymbol{W})kr}$的更新过程如下

$$\begin{aligned} &v^0_{\mathcal{Q}_{(\boldsymbol{S})jr}} = 0 \\ &v^t_{\mathcal{Q}_{(\boldsymbol{S})jr}} = \gamma v^{t-1}_{\mathcal{Q}_{(\boldsymbol{S})jr}} + \eta_{jr} \frac{\partial \varepsilon^{t-1}_{ijk}}{\partial Q^{t-1}_{(\boldsymbol{S})jr}} \\ &Q^t_{(\boldsymbol{S})jr} = Q^{t-1}_{(\boldsymbol{S})jr} - v^t_{\mathcal{Q}_{(\boldsymbol{S})jr}} \end{aligned} \tag{16}$$

$$\begin{aligned} &v^0_{\mathcal{Q}_{(\boldsymbol{W})kr}} = 0 \\ &v^t_{\mathcal{Q}_{(\boldsymbol{W})kr}} = \gamma v^{t-1}_{\mathcal{Q}_{(\boldsymbol{W})kr}} + \eta_{kr} \frac{\partial \varepsilon^{t-1}_{ijk}}{\partial Q^{t-1}_{(\boldsymbol{W})kr}} \\ &Q^t_{(\boldsymbol{W})kr} = Q^{t-1}_{(\boldsymbol{W})kr} - v^t_{\mathcal{Q}_{(\boldsymbol{W})kr}} \end{aligned} \tag{17}$$

本文使用3个辅助矩阵$\boldsymbol{V}^{|I|\times R}_{\mathcal{Q}_{(\boldsymbol{U})}}$、$\boldsymbol{V}^{|J|\times R}_{\mathcal{Q}_{(\boldsymbol{S})}}$和$\boldsymbol{V}^{|K|\times R}_{\mathcal{Q}_{(\boldsymbol{W})}}$来分别记录决策参数矩阵$\boldsymbol{\mathcal{Q}}_{(\boldsymbol{U})}$、$\boldsymbol{\mathcal{Q}}_{(\boldsymbol{S})}$和$\boldsymbol{\mathcal{Q}}_{(\boldsymbol{W})}$的更新速度。式(15)～式(17)中的随机梯度下降表达式已在式(12)给出。结合式(13)和式(15)～式(17)，得到结合动量方法的随机梯度下降算法的更新规则如下

$$(\boldsymbol{\mathcal{Q}}_{(\boldsymbol{U})}, \boldsymbol{\mathcal{Q}}_{(\boldsymbol{S})}, \boldsymbol{\mathcal{Q}}_{(\boldsymbol{W})}, \boldsymbol{V}_{\mathcal{Q}_{(\boldsymbol{U})}}, \boldsymbol{V}_{\mathcal{Q}_{(\boldsymbol{S})}}, \boldsymbol{V}_{\mathcal{Q}_{(\boldsymbol{W})}}) = \underset{\substack{\boldsymbol{\mathcal{Q}}_{(\boldsymbol{U})}, \boldsymbol{\mathcal{Q}}_{(\boldsymbol{S})}, \boldsymbol{\mathcal{Q}}_{(\boldsymbol{W})}, \\ \boldsymbol{V}_{\mathcal{Q}_{(\boldsymbol{U})}}, \boldsymbol{V}_{\mathcal{Q}_{(\boldsymbol{S})}}, \boldsymbol{V}_{\mathcal{Q}_{(\boldsymbol{W})}}}}{\arg\min}\ \varepsilon \overset{\text{MSGD}}{\Rightarrow}$$

$$\begin{cases} v^t_{\mathcal{Q}_{(\boldsymbol{U})ir}} = \gamma v^{t-1}_{\mathcal{Q}_{(\boldsymbol{U})ir}} + \eta_{ir} f'(Q^{t-1}_{(\boldsymbol{U})ir}) \cdot \\ \left(\lambda_{\mathcal{Q}_{(\boldsymbol{U})}} f(Q^{t-1}_{(\boldsymbol{U})ir}) - (t_{ijk} - \hat{t}_{ijk}) \cdot \right. \\ \left. f(Q^{t-1}_{(\boldsymbol{S})jr}) f(Q^{t-1}_{(\boldsymbol{W})kr}) \right), \\ v^t_{\mathcal{Q}_{(\boldsymbol{S})jr}} = \gamma v^{t-1}_{\mathcal{Q}_{(\boldsymbol{S})jr}} + \eta_{jr} f'(Q^{t-1}_{(\boldsymbol{S})jr}) \cdot \\ \left(\lambda_{\mathcal{Q}_{(\boldsymbol{S})}} f(Q^{t-1}_{(\boldsymbol{S})jr}) - (t_{ijk} - \hat{t}_{ijk}) \cdot \right. \\ \left. f(Q^{t-1}_{(\boldsymbol{U})ir}) f(Q^{t-1}_{(\boldsymbol{W})kr}) \right), \\ v^t_{\mathcal{Q}_{(\boldsymbol{W})kr}} = \gamma v^{t-1}_{\mathcal{Q}_{(\boldsymbol{W})kr}} + \eta_{kr} f'(Q^{t-1}_{(\boldsymbol{W})kr}) \cdot \\ \left(\lambda_{\mathcal{Q}_{(\boldsymbol{W})}} f(Q^{t-1}_{(\boldsymbol{W})kr}) - (t_{ijk} - \hat{t}_{ijk}) \cdot \right. \\ \left. f(Q^{t-1}_{(\boldsymbol{U})ir}) f(Q^{t-1}_{(\boldsymbol{S})jr}) \right), \\ Q^t_{(\boldsymbol{U})ir} \leftarrow Q^{t-1}_{(\boldsymbol{U})ir} - v^t_{\mathcal{Q}_{(\boldsymbol{U})ir}}, \\ Q^t_{(\boldsymbol{S})jr} \leftarrow Q^{t-1}_{(\boldsymbol{S})jr} - v^t_{\mathcal{Q}_{(\boldsymbol{S})jr}}, \\ Q^t_{(\boldsymbol{W})kr} \leftarrow Q^{t-1}_{(\boldsymbol{W})kr} - v^t_{\mathcal{Q}_{(\boldsymbol{W})kr}} \end{cases} \tag{18}$$

其中，速度向量的初始状态设置为0，即$v^0_{\mathcal{Q}_{(\boldsymbol{U})ir}} = v^0_{\mathcal{Q}_{(\boldsymbol{S})jr}} = v^0_{\mathcal{Q}_{(\boldsymbol{W})kr}} = 0$。式(18)是加速无约束张量隐因子分解模型的具体更新规则，本文通过非负函数映射和结合动量方法的随机梯度下降算法，消除了张量非负隐因子分解模型的非负性约束和更新过程中的非负要求。

2.5　算法设计与分析

基于上述推导，本文给出加速无约束张量隐因子分解模型的算法伪代码，如算法1所示。

算法 1　加速无约束张量隐因子分解模型

设置隐因子矩阵$\boldsymbol{U}^{|I|\times R}$、$\boldsymbol{S}^{|J|\times R}$、$\boldsymbol{W}^{|K|\times R}$以及速度矩阵$\boldsymbol{V}^{|I|\times R}_{\mathcal{Q}_{(\boldsymbol{U})}}$、$\boldsymbol{V}^{|J|\times R}_{\mathcal{Q}_{(\boldsymbol{S})}}$、$\boldsymbol{V}^{|K|\times R}_{\mathcal{Q}_{(\boldsymbol{W})}}$为0；初始化3个决策参数矩阵$\boldsymbol{\mathcal{Q}}^{|I|\times R}_{(\boldsymbol{U})}$、$\boldsymbol{\mathcal{Q}}^{|J|\times R}_{(\boldsymbol{S})}$、$\boldsymbol{\mathcal{Q}}^{|K|\times R}_{(\boldsymbol{W})}$，其初始随机值的取值范围为$(-5\times10^{-3}, 5\times10^{-3})$；初始化更新轮数$t$和轮数上限$T$。

1) while not converge and $t \leqslant T$ do
2)　for each $t_{ijk} \in \Lambda$ do
3)　$\hat{t}_{ijk} = \sum_{r=1}^{R} f(Q^{t-1}_{(\boldsymbol{U})ir}) f(Q^{t-1}_{(\boldsymbol{S})jr}) f(Q^{t-1}_{(\boldsymbol{W})kr})$
4)　end for
5)　for r=1 to R do
6)　根据式(18)更新速度矩阵和决策参数矩阵

7) end for

8) t=t+1

9) end while

10) for r=1 to R do

11) for i=1 to $|I|$ do

12) $U_{ir} = f(Q_{(U)ir})$

13) end for

14) for j=1 to $|J|$ do

15) $S_{jr} = f(Q_{(S)jr})$

16) end for

17) for k=1 to $|K|$ do

18) $W_{kr} = f(Q_{(W)kr})$

19) end for

20) end for

21) 当 2 轮连续更新的误差差值小于既定误差阈值时算法收敛

加速无约束张量隐因子分解模型的计算复杂度为

$$C_{\text{AULFT}} = \theta\left(\left(|I|+|J|+|K|\right)4R + t|\Lambda|2R\right) \approx \theta\left(t|\Lambda|R\right) \tag{19}$$

由于输入数据源为高维不完备张量，根据其特性，通常有 $|\Lambda| \gg \max\left\{|I|,|J|,|K|\right\}$。因此，式(19)的常数和低阶项被省略。在实际情况中，式(19)中的常数 t 和 R 都是正数，因此，加速无约束张量隐因子分解模型的计算复杂度与高维不完备张量中已知元素的数量 $|\Lambda|$ 呈线性关系。

加速无约束张量隐因子分解模型的空间复杂度由以下 2 个因素决定：1) 已知元素所需的缓存空间，3 个隐因子矩阵 $\boldsymbol{U}^{|I|\times R}$、$\boldsymbol{S}^{|J|\times R}$、$\boldsymbol{W}^{|K|\times R}$ 以及对应的 3 个决策参数矩阵 $\boldsymbol{Q}_{(U)}^{|I|\times R}$、$\boldsymbol{Q}_{(S)}^{|J|\times R}$、$\boldsymbol{Q}_{(W)}^{|K|\times R}$，其存储成本为 $\theta\left(6\left(|I|+|J|+|K|\right)R+|\Lambda|\right)$；2) 3 个辅助速度矩阵 $\boldsymbol{V}_{Q_{(U)}}^{|I|\times R}$、$\boldsymbol{V}_{Q_{(S)}}^{|J|\times R}$、$\boldsymbol{V}_{Q_{(W)}}^{|K|\times R}$，其存储成本为 $\theta\left(3\left(|I|+|J|+|K|\right)R\right)$。基于这 2 种因素且将常系数和低阶项合理地忽略后，加速无约束张量隐因子分解模型的空间复杂度为

$$S_{\text{AULFT}} = \theta\left(6\left(|I|+|J|+|K|\right)R+|\Lambda|+3\left(|I|+|J|+|K|\right)R\right) \approx \theta\left(\left(|I|+|J|+|K|\right)R+|\Lambda|\right) \tag{20}$$

从式(20)中可以看出，加速无约束张量隐因子分解模型的空间复杂度与高维不完备张量中隐因子元素以及已知元素的数量呈线性关系。基于上述的分析可知，本文模型在计算和存储方面都具有高效率。

3 实验结果与分析

3.1 常规设置

3.1.1 数据集

本文实验采用 WSMonitor[24]采集的 2 个数据集，数据集的详细信息如表 2 所示。它们描述了在 64 个不同的时间节点上 142 位用户调用 4 532 个服务产生的交互关系，衡量这种交互关系和 Web 之间的服务差异对应的服务指标分别是吞吐量和响应时间。由于每个用户在不同时刻调用不同的服务并没有受到其他用户的干扰，因此，本文将所有用户作为独立的个体并没有考虑用户之间的关系。产生的吞吐量和响应时间的数据规模都包含 30 287 611 条 QoS 记录。由于三维用户−服务−时间张量相比于二维的用户−服务矩阵增加了时序结构，能够更好地描述 QoS 数据的动态性，而且张量能够描述数据的高维性和复杂性。因此，本文将产生的 2 个大小为 142×4 532×64 的用户−服务−时间 QoS 张量作为模型的输入数据源。为了解决 2 个张量的数据密度为 74.06%高于现实情况的问题，本文对 2 个数据集分别各设计了 8 个测试用例，详细配置如表 3 所示。例如，在实验中取已知数据的 5%～40%作为训练集来训练每个测试模型，用剩余的 95%～60%数据作为测试集来评估模型的性能。例如，测试用例 D1.1 的训练集数据量占 D1 的 5%，这表示从 D1 中随机划分 5%的已知数据作为训练集 95%作为测试集。为了消除分割数据可能带来的偏差，重复上述过程十次，完成十组不同的实验，最后用它们的平均值作为本文的实验结果。

表 2 数据集的详细信息

数据集	数据类型	数据范围	平均值	用户数	服务数	时间节点个数	数据量	数据格式
D1	响应时间	0～20 s	3.165 s	142	4 532	64	30 287 611	User ID | Service ID | Time Slice ID | Response Time/ Throughput
D2	吞吐量	0～1 000 kbit/s	9.609 kbit/s	142	4 532	64	30 287 611	

表 3　　本文测试用例详细配置

Dataset	编号	训练集与测试集占比	训练集数据量	测试集数据量
D1	D1.1	5%:95%	1 514 381	28 773 230
	D1.2	10%:90%	3 028 761	27 258 850
	D1.3	15%:85%	4 543 142	25 744 469
	D1.4	20%:80%	6 057 522	24 230 089
	D1.5	25%:75%	7 571 903	22 715 708
	D1.6	30%:70%	9 086 283	21 201 328
	D1.7	35%:65%	10 600 664	19 686 947
	D1.8	40%:60%	12 115 044	18 172 567
D2	D2.1	5%:95%	1 514 381	28 773 230
	D2.2	10%:90%	3 028 761	27 258 850
	D2.3	15%:85%	4 543 142	25 744 469
	D2.4	20%:80%	6 057 522	24 230 089
	D2.5	25%:75%	7 571 903	22 715 708
	D2.6	30%:70%	9 086 283	21 201 328
	D2.7	35%:65%	10 600 664	19 686 947
	D2.8	40%:60%	12 115 044	18 172 567

为了避免由分布范围过大和过小的特征主导欧氏距离计算，本文对 2 个数据集采取线性特征缩放[22-23]。给定特征值 α 的上限 $\alpha_{\max}$ 和下限 $\alpha_{\min}$，经过缩放后的特征值为

$$\tilde{\alpha}=10\frac{\alpha-\alpha_{\min}}{\alpha_{\max}-\alpha_{\min}} \tag{21}$$

经过特征缩放后，响应时间和吞吐量数据集的特征尺度分别为 0～10 s 和 0～10 kbit/s。

3.1.2　评价指标

本文主要考虑模型的估计精度和计算效率。估计精度计算估计值与实际值之间的差距，直观地反映出模型是否捕捉到输入数据的基本特征[25-27]。本文采用均方根误差（RMSE, root mean square error）和平均绝对误差（MAE, mean absolute error）作为估计精度的评估指标，如式(22)所示。

$$\begin{aligned}\mathrm{RMSE}&=\sqrt{\frac{\left(\sum\limits_{t_{ijk}\in\Omega}\left(t_{ijk}-\hat{t}_{ijk}\right)^2\right)}{|\Omega|}}\\ \mathrm{MAE}&=\frac{\left(\sum\limits_{t_{ijk}\in\Omega}\left|t_{ijk}-\hat{t}_{ijk}\right|\right)}{|\Omega|}\end{aligned} \tag{22}$$

其中，Ω 是测试集，并且测试集与训练集相互独立。对于被测试的模型，均方根误差和平均绝对误差的值越小，表示模型的估计精度越高。同时，本文通过记录每个测试模型的收敛所需的更新轮数和每轮更新的时间成本来比较被测试模型的效率。

3.1.3　模型设置

为了获得客观的实验结果，本文给出以下设置。

1) 为了使模型获得客观公平的比较，设置隐因子空间维度 R=20 以消除超参数对模型的影响。

2) 将模型的更新轮数上限设置为 1 000 轮，即模型更新轮数达到 1 000 轮时，该模型终止训练。

3) 若 2 轮连续更新的误差差值在 D1 上小于 10^{-5}，在 D2 上小于 10^{-7}，则该模型收敛。不同的数据集，误差差值的阈值不同，这是因为 D1 的数据范围从 0～20 s 到 0～10 s 只缩小至 $\frac{1}{2}$，而 D2 从 0～1 000 kbit/s 到 0～10 kbit/s 缩小至 $\frac{1}{100}$，显然 D2 的收缩程度比 D1 更加的剧烈。

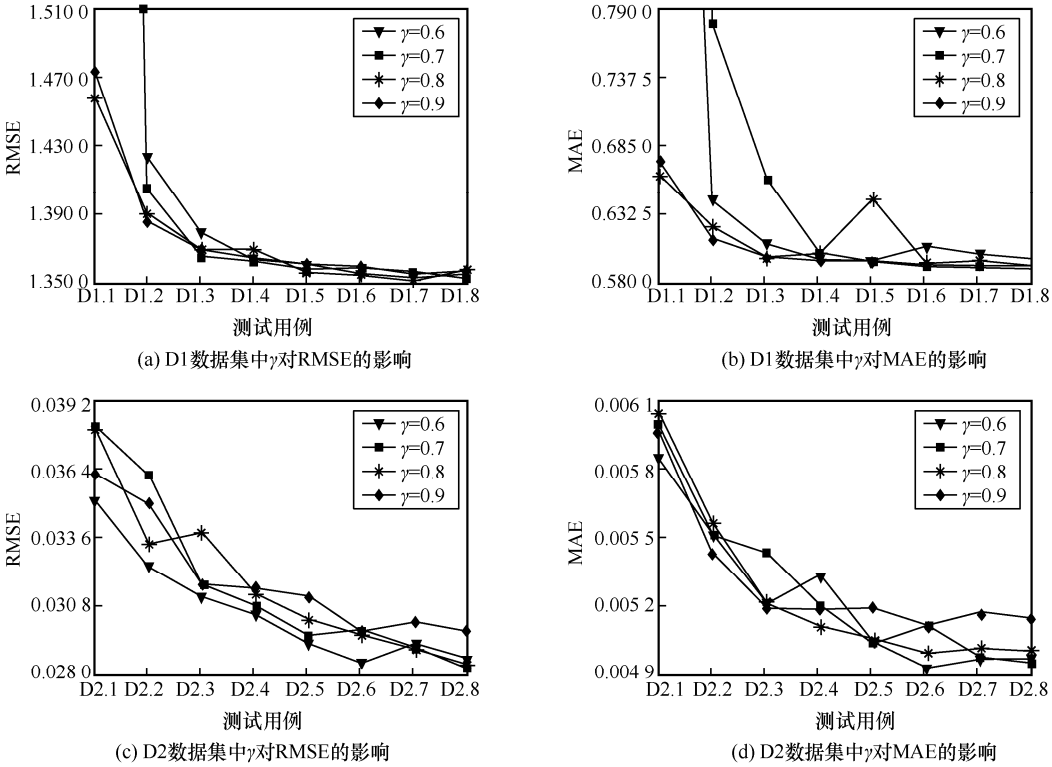
3.2　参数敏感性实验

加速无约束张量隐因子分解模型的性能依赖于动量系数 γ。在结合动量方法的随机梯度下降算法原理中，动量系数的作用是控制动量信息对整体梯度更新的影响程度，$\gamma=0$ 等效于常规的随机梯度下降算法，而 γ>1 则会造成模型无法收敛，因此，γ 取值范围为 [0,1]。本文发现，在进行实验时，$\gamma\in[0.6,0.9]$ 会使加速无约束张量隐因子分解模型表现出较理想的效果，因此将该区间中 γ 的每个取值（步长为 0.1）分别在 2 个数据集的 8 个测试用例上进行实验。为了反映动量项对模型性能的影响，本文提前在不同数据集上分别调试出了最优 η 和 λ 值，如表 4 所示。不同 γ 值在不同测试用例上对加速无约束张量隐因子分解模型性能的影响如图 5 所示。

表 4　　各数据集最优 η和λ 值

数据集	η	λ
D1	2^{-15}	10^{-7}
D2	2^{-8}	10^{-12}

图 5(a)中，γ =0.6、0.7 在 D1.1 上的 RMSE 值分别为 3.444 787、3.444 666，相较于 γ =0.8、0.9 在 D1.1 上的 RMSE 值 1.459 485、1.473 890 差距太大，因此不在图 5(a)中展示。同样地，图 5(b)中 γ =0.6、0.7 在 D1.1 上的 MAE 值分别为 1.582 652、1.582 556，相较于 γ =0.8、0.9 在 D1.1 上的 MAE 值 0.660 877、0.672 498 差距太大，故不在图 5(b)中展示。



(a) D1数据集中γ对RMSE的影响　(b) D1数据集中γ对MAE的影响

(c) D2数据集中γ对RMSE的影响　(d) D2数据集中γ对MAE的影响

图 5　不同 γ 值在不同测试用例上对加速无约束张量隐因子分解模型性能的影响

从图 5(a)和图 5(b)中可以看出，对于 D1.1～D1.3，γ =0.8 和 0.9 的表现情况明显优于 γ =0.6 和 0.7。图 5(a)中的 D1.4～D1.8 上，不同 γ 值的表现情况是非常接近的。如在 D1.8 上，当 γ =0.6, 0.7, 0.8, 0.9 时，RMSE 值分别为 1.354 938, 1.352 724, 1.356 738, 1.356 196，其值波动幅度仅为最低值的 0.29%（波动幅度=（最大值−最低值）/最低值）。然而在图 5(b)的 D1.4～D1.8 上，不同 γ 的表现情况并不接近。例如，D1.5 上表现最差的 γ =0.8 的 MAE 值（为 0.643 180）比表现最优的 γ =0.6 的 MAE 值（为 0.595 452）上升了 8%。综合图 5(a)和图 5(b)来看，γ =0.9 比其他 3 个 γ 值表现更稳定，因此本文将 γ =0.9 设置为加速无约束张量隐因子分解模型在 D1 数据集上的最优动量系数。

从图 5(c)和图 5(d)中可以看出，γ =0.6 在 D2.6 上的 RMSE 值和 MAE 值同时达到了最优值，分别是 0.028 572 和 0.004 935，并且 γ =0.6 在图 5(c)中的 D2.1～D2.6 上表现也明显优于其他值。因此，本文将 γ =0.6 设置为加速无约束张量隐因子分解模型在 D2 数据集上的最优动量系数。

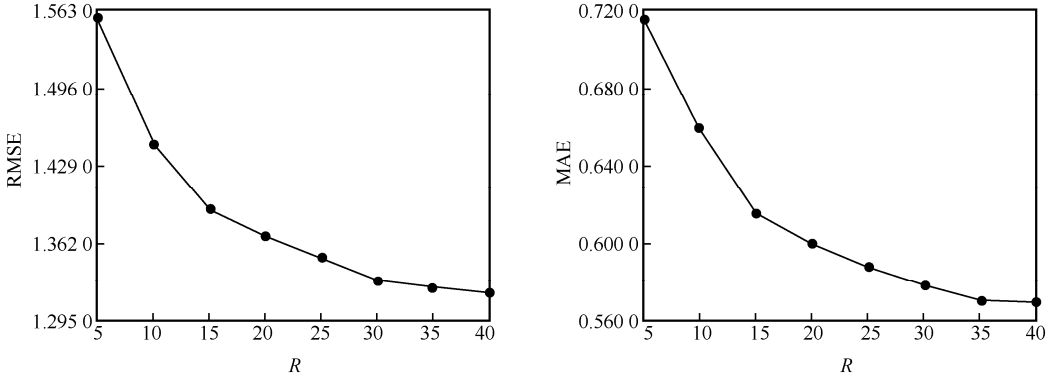
通过本节实验，本文得到了加速无约束张量隐因子分解模型在 2 个数据集上的最优动量系数。下面，本文将比较不同 R 值对加速无约束张量隐因子分解模型性能的影响。

3.3　R 的影响

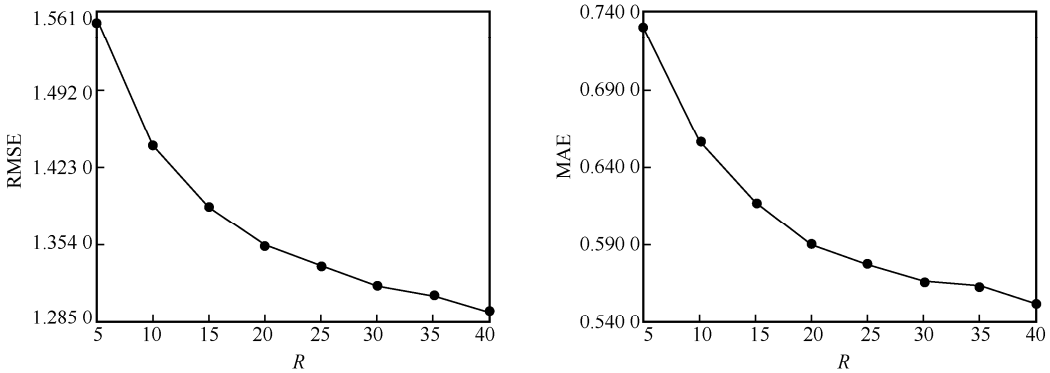
隐因子维度 R 会影响张量隐因子分解模型的性能[28]，模型的估计误差随着 R 值的增加而减小，当 R 值接近目标张量的实际秩时趋于稳定。这样的情况是否也会在加速无约束张量隐因子分解模型上出现？为了回答这个问题，本文在所有测试用例上进行了实验，维度 R 对 D1 和 D2 的影响分别如图 6 和图 7 所示。

从图 6 可以看出，在 D1 数据集上，加速无约束张量隐因子分解模型对缺失数据的估计精度与 R 值呈正相关。模型估计误差在最初时随着 R 的增大而急剧减小，但随着 R 的增大，其误差减小的趋势变缓。例如在图 6(b)中，当训练数据比为 20%时，R=5 时模型的 MAE 值为 0.714 795，R=25 时为 0.587 940，表明随着 R 从 5 增加到 25，模型的估计精度提高了 21.57%。但当 R 增加到 40 时，其 MAE 值为 0.568 896，相对于 R=25 时仅提高了 3.34%。相似的情况也出现在图 6(a)、图 6(c)和图 6(d)中。

然而在 D2 数据集上，R 值与加速无约束张量隐因子分解模型对缺失数据的估计精度却不再呈正相关。在 D2 数据集上，当 R 值增大到某一临界

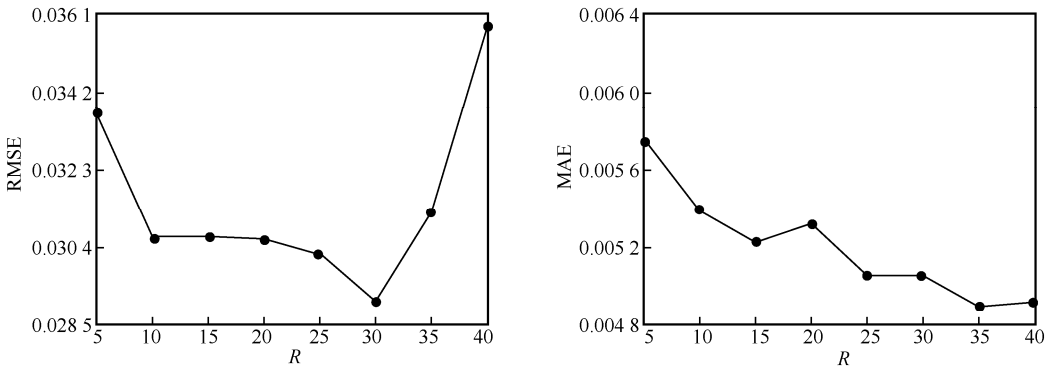


(a) 训练数据比为20%时D1的RMSE　　(b) 训练数据比为20%时D1的MAE

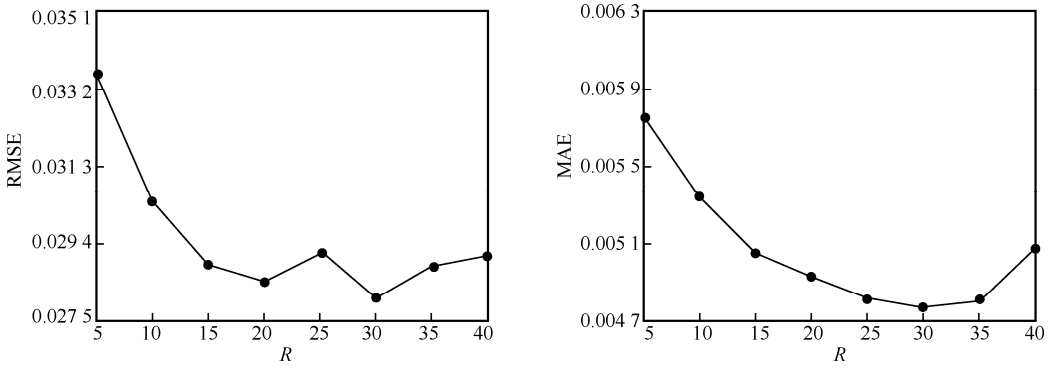


(c) 训练数据比为40%时D1的RMSE　　(d) 训练数据比为40%时D1的MAE

图 6　维度 R 对 D1 的影响



(a) 训练数据比为20%时D2的RMSE　　(b) 训练数据比为20%时D2的MAE



(c) 训练数据比为40%时D2的RMSE　　(d) 训练数据比为40%时D2的MAE

图 7　维度 R 对 D2 的影响

值（R=30 或 35）后，模型的估计精度不再提升反而出现下降趋势。这种情况在图 7(a)中尤其明显，当训练数据比为 20%时，R=30 时 RMSE=0.029 088，R=40 时 RMSE=0.035 818，R=40 相对于 R=30 时估计精度下降了 23.14%。

通过上述的讨论，本文发现隐因子空间维度 R 对加速无约束张量隐因子分解模型来说并不是越大越好。当 R 值超过高维不完备张量的实际秩时，加速无约束张量隐因子分解模型的估计精度提升不大甚至会出现下降的可能。如 2.5 节所分析的，加速无约束张量隐因子分解模型的计算复杂度与 R 呈线性关系，但其性能却与 R 呈非线性关系。因此，对于采用加速无约束张量隐因子分解模型估计缺失 QoS 数据的工业应用，有必要对于高维不完备张量的隐因子维度进行谨慎的调优。

3.4 模型间性能比较

本节分别从估计精度和平均每轮更新的时间开销两方面对加速无约束张量隐因子分解模型和主流的时变感知 QoS 估计方法进行比较。对比实验涉及的模型如下。

M1：无约束非负隐因子分析模型[29]。该模型把用户–服务–时间张量沿着时间维度进行分割，从而得到一系列用户–服务矩阵切片，并对每个矩阵切片进行无约束非负隐因子分析，该模型采用随机梯度下降算法作为学习方案。

M2：张量隐因子分解模型。该模型将正则多元张量分解法应用于用户–服务–时间张量进行张量隐因子分解。该模型采用加性梯度下降算法作为学习方案，但加性梯度下降算法不能保证 QoS 数据的非负性。

M3：基于广义牛顿加速梯度的张量有偏非负隐因子分解模型[20]。该模型通过正则多元张量分解法对用户–服务–时间张量进行张量隐因子分解，模型的初始随机数据必须是非负数且依赖于特意设计的非负广义牛顿加速梯度学习方案。

M4：有偏非负隐因子分解模型[19]。该模型通过正则多元张量分解法对用户–服务–时间张量进行张量隐因子分解，采用单隐因子依赖非负乘法更新和交替方向法融合作为非负学习方案，模型初始随机数据必须是非负数。

M5：基于神经网络的协同过滤模型[30]。该模型采用神经网络的技术来解决基于隐式反馈的协同过滤问题，通过用一个可以从数据中学习任意函数的神经结构来提取用户和服务的潜在特征，并且利用多层感知器来学习用户–服务交互特征。

M6：本文所提出的加速无约束张量隐因子分解（AULFT）模型。

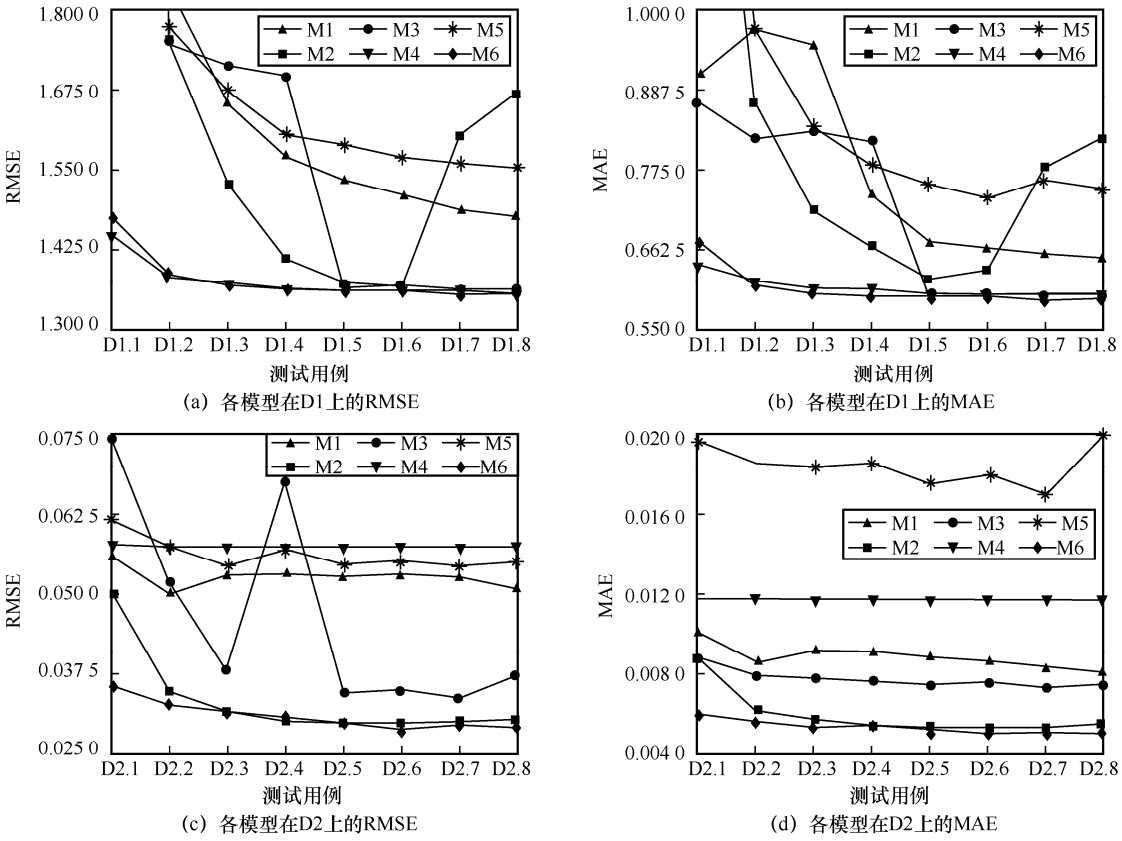
对于超参数的设置，本文为所有涉及的模型设置其隐因子空间维度 R=20 以进行公平比较。同时，本文在每组实验中对每个模型进行相同的初始化和随机生成假设以消除初始假设的影响。对于建模所需的超参数，本文首先在一组实验中为每个模型分别在 2 个数据集上调试出其最优参数值，然后再应用于其他实验中。

各模型在 D1 和 D2 的所有测试用例上的 RMSE、MAE 值如图 8 所示，所有测试模型每轮更新的时间开销以及 RMSE/MAE 收敛所需的更新轮数分别如表 5～表 7 所示。从这些结果中可以得出以下结论。

1) 正确地捕捉 QoS 数据随时间变化产生的波动性能够实现对缺失数据的高度准确地估计。M1 将高维不完备张量视为一组矩阵切片，但该模型忽视了数据随时间变化产生的波动性。同样地，M5 只考虑了用户、服务以及用户–服务的交互信息，没有考虑到数据的时间特性。这使 M1 和 M5 的估计精度比准确捕捉时间特性的 M6 差。比如，在图 8(a)的 D1.3 测试用例上，即训练数据比为 15%时，M1 的 RMSE 为 1.654 950，M5 的 RMSE 为 1.673 982，分别比 M6 的 RMSE（为 1.369 543）高了 20.84%、22.22%。即使是训练数据比增至 40%时，M1 的 RMSE 为 1.477 534，M5 的 RMSE 为 1.552 957，也依旧比 M6 的 RMSE（为 1.356 196）高了 8.95%和 14.51%。相似的情况也出现在图 8(b)～图 8(d)中。

2) 将非负性约束应用于张量隐因子分解模型来描绘 QoS 数据的非负性是必要的。M2 与 M6 同样采用了张量隐因子分解，但与 M6 的不同之处在于，M2 没有考虑到 QoS 数据的非负性，这使 M2 对于缺失数据的估计精度比 M6 差。如图 8(b)所示，在训练数据比为 10%时，M2 的 MAE 为 0.868 477，比 M6 的 MAE（为 0.612 579）高了 41.77%。即使训练数据比为 25%时，M2 的估计精度达到了最优值 0.622 145，依旧比 M6 的 MAE（为 0.596 277）高了 4.34%。类似的情况也出现在图 8(a)、图 8(c)和图 8(d)中。这个现象说明正确描述数据的非负性对基于张量隐因子分解的 QoS 估计模型是至关重要的。

3) 合适的算法不仅能够帮助张量隐因子分解模型提升估计精度，还能够在数据稀疏度发生变化



(a) 各模型在D1上的RMSE　(b) 各模型在D1上的MAE

(c) 各模型在D2上的RMSE　(d) 各模型在D2上的MAE

图 8　各模型在 D1 和 D2 的所有测试用例上的 RMSE、MAE

时保持稳定且精准的估计水平。M3 与 M6 的不同之处在于所用的算法不同，但广义牛顿加速梯度算法无法让 M3 在数据稀疏度发生变化时保持稳定和高度准确的估计精度。例如在图 8(c)中，当训练数据比增至 20%时，M3 的 RMSE 为 0.066 896，M6 的 RMSE 为 0.030 516，M3 比 M6 高 119.22%，并且 M3 比其本身在训练数据比为 15%时的 RMSE（为 0.037 725）高 77.32%。在图 8(d)中，虽然 M3 在训练数据比发生变化下保持稳定，但其估计精度明显不如 M6。

4) 当目标高维不完备张量变稀疏时，AULFT 模型仍然保持较好的估计精度。当训练数据只占已知数据总数的 5%时，此时输入的目标高维不完备张量变得稀疏，这种情况对应 D1.1 和 D2.1 这 2 种测试用例。在图 8(a)的 D1.1 上，M6 的估计精度排名位于所有模型的第二位。此时 M6 的 RMSE 为 1.473 890，相对于 M1（RMSE=1.924 484）提升了 48.74%，M2（RMSE=3.444 679）提升了 57.21%，M3（RMSE=4.872 853）提升了 69.75%，M5（RMSE= 2.636 278）提升了 44.09%，M4（RMSE= 1.444 619）略微低 6.58%。相似的情况也出现在图 8(b)的 D1.1 上。

表 5　　各模型每轮更新的时间开销

数据集	每轮更新的时间开销/ms					
	M1	M2	M3	M4	M5	M6
D1	6 427	3 523	13 620	7 047	—	11 852
D2	5 157	3 599	10 535	15 168	—	10 186

在 D2.1 测试用例中，M6 的估计精度最高。在图 8(d)的 D2.1 上，M6 的 MAE 值为 0.005 846，相对于 M1（MAE=0.009 950）提升了 41.24%，M2（MAE=0.008 682）提升了 32.66%，M3（MAE= 0.008 589）提升了 31.93%，M4（MAE=0.011 611）提升了 49.65%，M5（MAE=0.019 537）提升了 70.07%。在图 8(c)的 D2.1 测试用例上，M6 也同样占据领先地位。这些实验数据证明了 AULFT 模型能够很好地处理稀疏的高维不完备张量。

表 6　　各模型 RMSE 收敛所需的更新轮数

数据集	更新轮数/轮					
	M1	M2	M3	M4	M5	M6
D1	347	1 000	723	629	—	211
D2	1 000	1 000	202	54	—	283

表 7 各模型 MAE 收敛所需的更新轮数

数据集	更新轮数/轮					
	M1	M2	M3	M4	M5	M6
D1	408	1 000	750	623	—	212
D2	1 000	1 000	203	13	—	266

5) 计算效率。表 5～表 7 分别展示了所有模型在训练数据比为 40%时的每轮更新时间开销以及 RMSE/MAE 的收敛所需的更新轮数。由于 M5 为神经网络结构，解释性较差，不同于其余的优化模型，在此不进行对比。从表 5 可以看出，M6 每轮更新的时间开销比较大，与 M3 相当。这是由于 AULFT 模型中每个隐因子都需要经过非负映射函数转换，这样的做法会产生额外的时间成本。但从表 6 和表 7 中可以看到，M6 的收敛所需的更新轮数较少，在所有受测试模型中还是具有竞争力的。

3.5 消融实验

3.5.1 不同映射函数对加速无约束张量隐因子分解模型的影响

如 2.1 节所述，AULFT 模型通过将非负性约束从决策参数转移到输出的隐因子，并通过单元素的非负映射函数 f 连接它们，以此来消除张量非负隐因子分解模型损失函数所必要的非负性约束。那么，不同的非负映射函数是否会给 AULFT 模型带来不同的效果？为了得到答案，本文选择 3 个不同的非负映射函数进行实验。所选取的函数及其导函数如表 8 所示，在训练数据比为 40%的情况下，D1、D2 中不同映射函数对 AULFT 模型的影响如图 9 所示。通过实验，本文得到了如下结论：通过适当选择映射函数，可以使 AULFT 模型实现较高的估计精度。由图 9 可看出，AULFT_Absolute 在估计精度方面的表现明显优于 AULFT 与其他映射函数的结合。例如，在图 9(a)上，AULFT_Absolute 的最优 RMSE 值为 1.359 225，而 AULFT_Hard-Sigmoid 为 1.807 724，AULFT_ReLU 为 2.288 575，AULFT_Absolute 的提升幅度分别为24.81%、40.61%。

按照 MAE 衡量，AULFT_Absolute 提升的幅度仍然很大。如 AULFT_Absolute 在图 9(b)上的 MAE 为 0.59 3254，比 AULFT_Hardsigmoid 的 0.890 505、AULFT_RELU 的 1.145 199 分别提升了 33.38%、48.2%。类似的情况也出现在图 9(c)和图 9(d)上。

3.5.2 动量方法对加速无约束张量隐因子分解模型的影响

AULFT 模型结合动量方法进行更新训练，如式(18)所示。动量方法为该模型带来的提升是什么？为了弄清楚这个问题，在这组实验中，本文对使用动量方法和没有使用动量方法的 AULFT 模型进行比较。没有使用动量方法的 AULFT 模型采用随机梯度下降算法作为学习方案，其参数更新规则如式(13)所示。当训练数据比为 40%时，D1、D2 中动量对 AULFT 模型的影响如图 10 所示。

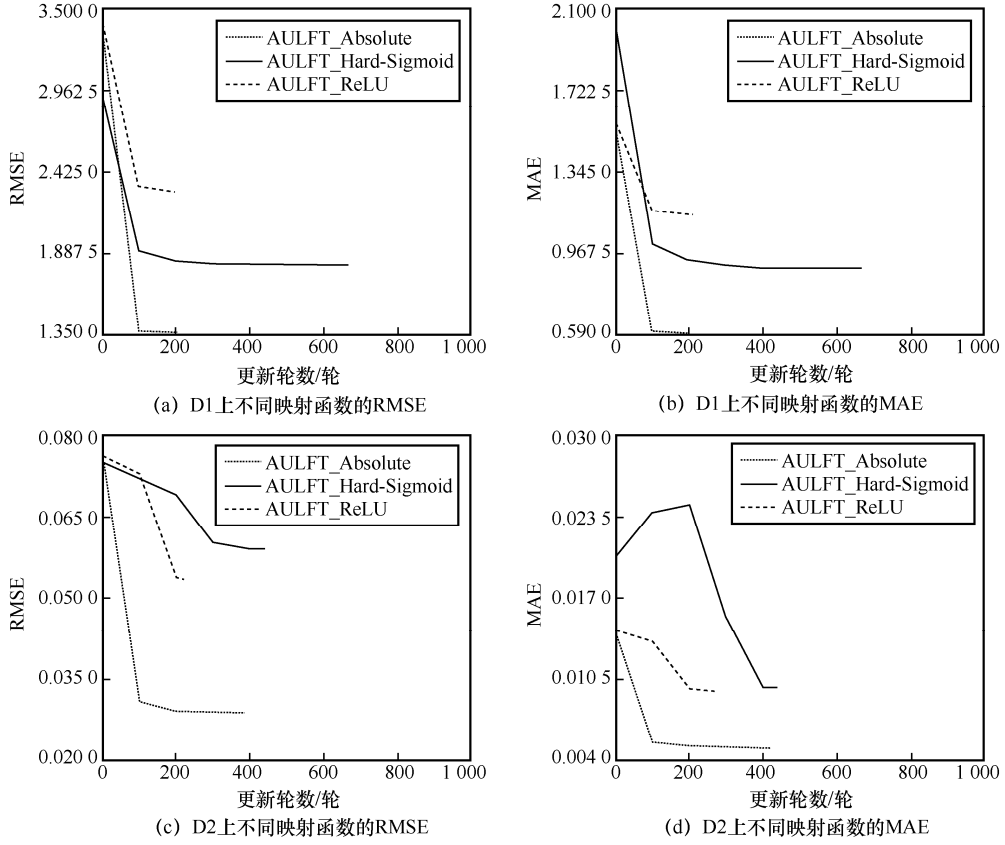
从图 10 可以直观地看到，动量方法不仅大大减少了模型的收敛更新轮数，还提高了模型对缺失数据的估计精度，极大地提升了 AULFT 模型的总体性能。例如，从图 10(c)中，使用动量方法的 AULFT 模型在 D2 上的最优 RMSE 值和收敛所需的更新轮数分别是 0.028 661、383 轮，对比没有使用动量方法的 AULFT 模型的 0.039 548、1 000 轮分别提升了 27.53%和 61.7%。类似的情况也出现在图 10(a)、图 10(b)和图 10(d)上。如图 10 所示，没有使用动量方法的 AULFT 模型容易遭受长尾收敛，因此使用动量方法对 AULFT 模型实现快速收敛是十分必要的。

3.6 实验总结

本文根据上述一系列实验，可以得出如下结论：由于能够正确地对隐藏在 QoS 数据中的时间动

表 8 非负映射函数及其导函数

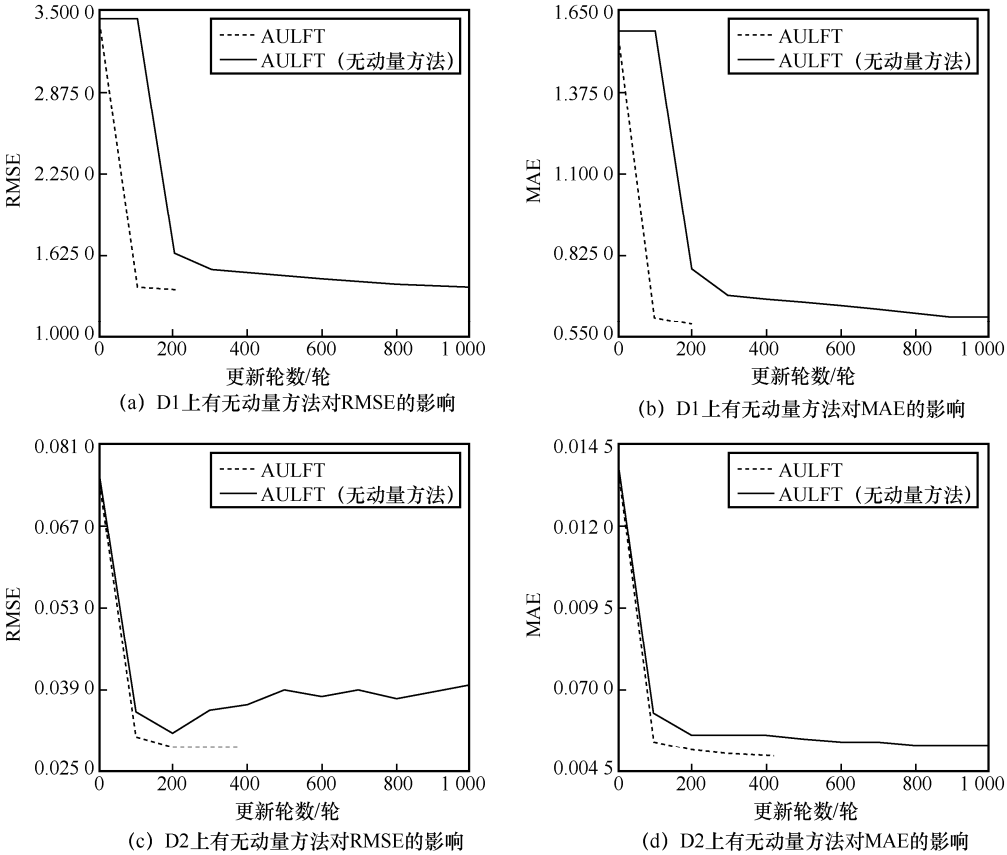
函数名	映射函数表达式	导函数表达式
Absolute	$f(a)=\lvert a\rvert$	$f'(a)=\begin{cases}1, & a>0\\ -1, & a\leqslant 0\end{cases}$
Hard-Sigmoid	$f(a)=\begin{cases}0, a<-2.5\\ 0.2a+0.5, & -2.5\leqslant a\leqslant 2.5\\ 1, & a>2.5\end{cases}$	$f'(a)=\begin{cases}0, a<-2.5\\ 0.2, -2.5\leqslant a\leqslant 2.5\\ 0, a>2.5\end{cases}$
ReLU	$f(a)=\begin{cases}a, & a>0\\ 0, & a\leqslant 0\end{cases}$	$f'(a)=\begin{cases}1, & a>0\\ 0, & a\leqslant 0\end{cases}$



(a) D1上不同映射函数的RMSE　(b) D1上不同映射函数的MAE

(c) D2上不同映射函数的RMSE　(d) D2上不同映射函数的MAE

图 9　不同映射函数对 AULFT 模型的影响



(a) D1上有无动量方法对RMSE的影响　(b) D1上有无动量方法对MAE的影响

(c) D2上有无动量方法对RMSE的影响　(d) D2上有无动量方法对MAE的影响

图 10　动量对 AULFT 模型的影响

态性进行建模和保证了 QoS 估计值的非负性，使 AULFT 模型能够高度准确地估计丢失的 QoS 数据，并且合适的动量系数 γ、隐因子维度 R 可以进一步提高 AULFT 模型的估计准确性。

根据上述实验，本文在表 9 中总结了各数据集上 AULFT 模型的最优参数值。

表 9 各数据集上 AULFT 模型的最优参数值

数据集	R	η	λ	γ
D1	40	2^{-15}	10^{-7}	0.9
D2	30	2^{-8}	10^{-12}	0.6

4 结束语

为了实现时间模式感知的 QoS 数据估计，本文将目标数据建模为用户–服务–时间张量，以执行张量隐因子分解。本文提出了加速无约束张量隐因子分解模型，所提模型具有以下优点。1) 所提模型不需要考虑初始随机数据的非负性，不依赖于特定的非负训练方案。因此，该模型比张量非负隐因子分解模型具有更好的兼容性。2) 所提模型通过采用结合动量方法的随机梯度下降算法作为训练方案，使模型的估计精度与时间开销在同类 QoS 估计方法具有相当的竞争力。加速无约束张量隐因子分解模型的性能依赖于动量系数和隐因子维度，如何令其自适应调整从而使该模型表现出最优性能是尚未解决的问题。这个问题会在未来的研究工作中进行解决。

参考文献：

[1] ADELEYE O, YU J, WANG G L, et al. Constructing and evaluating evolving Web-API networks–a complex network perspective[J]. IEEE Transactions on Services Computing, 2023, 16(1): 177-190.

[2] 张鹏程, 魏芯淼, 金惠颖. 移动边缘计算下基于联邦学习的动态 QoS 优化[J]. 计算机学报, 2021, 44(12): 2431-2446.
ZHANG P C, WEI X M, JIN H Y. Dynamic QoS optimization method based on federal learning in mobile edge computing[J]. Chinese Journal of Computers, 2021, 44(12): 2431-2446.

[3] YANG Y T, ZHENG Z B, NIU X D, et al. A location-based factorization machine model for Web service QoS prediction[J]. IEEE Transactions on Services Computing, 2021, 14(5): 1264-1277.

[4] 张红霞, 武梦德, 王登岳, 等. 基于服务负载的时序 QoS 预测[J]. 计算机系统应用, 2023, 32(11): 286-293.
ZHANG H X, WU M D, WANG D Y, et al. Time-series QoS prediction based on service load[J]. Computer Systems and Applications, 2023, 32(11): 286-293.

[5] 李云, 高倩, 姚枝秀, 等. 移动边缘计算中智能服务编排和算网资源分配联合优化方法[J]. 通信学报, 2023, 44(7): 51-63.
LI Y, GAO Q, YAO Z X, et al. Joint optimization method of intelligent service arrangement and computing-networking resource allocation for MEC[J]. Journal on Communications, 2023, 44(7): 51-63.

[6] GHAFOURI S H, HASHEMI S M, HUNG P C K. A survey on Web service QoS prediction methods[J]. IEEE Transactions on Services Computing, 2022, 15(4): 2439-2454.

[7] CHEN M Z, WU H. Efficient representation to dynamic QoS data via momentum-incorporated biased nonnegative and adaptive latent factorization of tensors[C]//Proceedings of the 2021 International Conference on Cyber-Physical Social Intelligence (ICCSI). Piscataway: IEEE Press, 2021: 1-6.

[8] WU D, LUO X, SHANG M S, et al. A data-characteristic-aware latent factor model for Web services QoS prediction[J]. IEEE Transactions on Knowledge and Data Engineering, 2022, 34(6): 2525-2538.

[9] WU D, HE Q, LUO X, et al. A posterior-neighborhood-regularized latent factor model for highly accurate Web service QoS prediction[J]. IEEE Transactions on Services Computing, 2022, 15(2): 793-805.

[10] LIU A, SHEN X D, LI Z X, et al. Differential private collaborative Web services QoS prediction[J]. World Wide Web, 2019, 22(6): 2697-2720.

[11] 李元诚, 秦永泰. 基于深度强化学习的软件定义安全中台 QoS 实时优化算法[J]. 通信学报, 2023, 44(5): 181-192.
LI Y C, QIN Y T. Deep reinforcement learning based algorithm for real-time QoS optimization of software-defined security middle platform[J]. Journal on Communications, 2023, 44(5): 181-192.

[12] LUO X, ZHOU M C, XIA Y N, et al. Generating highly accurate predictions for missing QoS data via aggregating nonnegative latent factor models[J]. IEEE Transactions on Neural Networks and Learning Systems, 2016, 27(3): 524-537.

[13] 刘建勋, 丁领航, 康国胜, 等. 基于特征深度融合的 Web 服务 QoS 联合预测[J]. 通信学报, 2022, 43(7): 215-226.
LIU J X, DING L H, KANG G S, et al. Joint QoS prediction for Web services based on deep fusion of features[J]. Journal on Communications, 2022, 43(7): 215-226.

[14] CHEN Z, SHEN L M, LI F, et al. Your neighbors alleviate cold-start: on geographical neighborhood influence to collaborative Web service QoS prediction[J]. Knowledge-Based Systems, 2017, 138: 188-201.

[15] TANG M D, ZHENG Z B, KANG G S, et al. Collaborative Web service quality prediction via exploiting matrix factorization and network map[J]. IEEE Transactions on Network and Service Management, 2016, 13(1): 126-137.

[16] FENG Y, HUANG B Q. Cloud manufacturing service QoS prediction based on neighborhood enhanced matrix factorization[J]. Journal of Intelligent Manufacturing, 2020, 31(7): 1649-1660.

[17] SENTHIL KUMAR S, MARGRET ANOUNCIA S. QoS-based concurrent user-service grouping for Web service recommendation[J]. Automatic Control and Computer Sciences, 2018, 52(3): 220-230.

[18] RYU D, LEE K, BAIK J. Location-based Web service QoS prediction via preference propagation to address cold start problem[J]. IEEE Transactions on Services Computing, 2021, 14(3): 736-746.

[19] LUO X, WU H, YUAN H Q, et al. Temporal pattern-aware QoS prediction via biased non-negative latent factorization of tensors[J]. IEEE Transactions on Cybernetics, 2020, 50(5): 1798-1809.

[20] CHEN M Z, LUO X. Efficient representation to dynamic QoS data via generalized nesterov's accelerated gradient-incorporated biased

non-negative latent factorization of tensors[C]//Proceedings of the 2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC). Piscataway: IEEE Press, 2021: 576-581.

[21] CHEN M Z, HE C L, LUO X. MNL: a highly-efficient model for large-scale dynamic weighted directed network representation[J]. IEEE Transactions on Big Data, 2023, 9(3): 889-903.

[22] MAEHARA T, HAYASHI K, KAWARABAYASHI K I. Expected tensor decomposition with stochastic gradient descent[C]//Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence. New York: ACM Press, 2016: 1919-1925.

[23] LUO X, WANG D X, ZHOU M C, et al. Latent factor-based recommenders relying on extended stochastic gradient descent algorithms[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2021, 51(2): 916-926.

[24] ZHANG Y L, ZHENG Z B, LYU M R. WSPred: a time-aware personalized QoS prediction framework for Web services[C]//Proceedings of the 2011 IEEE 22nd International Symposium on Software Reliability Engineering. Piscataway: IEEE Press, 2011: 210-219.

[25] 许建龙, 林健, 黎宇森, 等. 分布式用户隐私保护可调节的云服务个性化 QoS 预测模型[J]. 网络与信息安全学报, 2023, 9(2): 70-80.
XU J L, LIN J, LI Y S, et al. Distributed user privacy preserving adjustable personalized QoS prediction model for cloud services[J]. Chinese Journal of Network and Information Security, 2023, 9(2): 70-80.

[26] 高文斌, 王睿, 王田丰, 等. 基于深度强化学习的 QoS 感知 Web 服务组合[J]. 计算机技术与发展, 2022, 32(6): 92-98.
GAO W B, WANG R, WANG T F, et al. QoS-aware service composition based on deep reinforcement learning[J]. Computer Technology and Development, 2022, 32(6): 92-98.

[27] WU D, ZHANG P, HE Y, et al. A double-space and double-norm ensembled latent factor model for highly accurate Web service QoS prediction[J]. IEEE Transactions on Services Computing, 2023, 16(2): 802-814.

[28] WANG Q X, CHEN M Z, SHANG M S, et al. A momentum-incorporated latent factorization of tensors model for temporal-aware QoS missing data prediction[J]. Neurocomputing, 2019, 367(C): 299-307.

[29] LUO X, ZHOU M C, LI S, et al. Algorithms of unconstrained non-negative latent factor analysis for recommender systems[J]. IEEE Transactions on Big Data, 2021, 7(1): 227-240.

[30] HE X N, LIAO L Z, ZHANG H W, et al. Neural collaborative filtering[C]//Proceedings of the 26th International Conference on World Wide Web. New York: ACM Press, 2017: 173-182.

[作者简介]

林铭炜（1985− ），男，福建莆田人，博士，福建师范大学教授，主要研究方向为服务计算、大数据分析、智能决策等。

李文强（1997− ），男，福建泉州人，福建师范大学硕士生，主要研究方向为服务计算、机器学习、统计数据建模和张量方法。

许秀琴（1995− ），女，福建莆田人，福建师范大学博士生，主要研究方向为服务计算、机器学习、统计数据建模和张量方法。

刘健（1988− ），男，福建莆田人，博士，福建师范大学讲师，主要研究方向为自然语言处理。