# 面向云服务QoS预测的可调节用户隐私分布式矩阵分解模型

许建龙 林健 熊智

(汕头大学计算机系 汕头 515063)

**摘 要** 个性化服务质量(QoS)预测是构建高质量云服务系统的重要环节，传统基于协同过滤方法采用集中式的训练模式难以保护用户隐私，为了在获取高准确预测效果的同时能有效保护用户隐私，本文提出面向云服务QoS预测的可调节隐私程度的分布式矩阵分解模型(DMF-AP)，该模型允许用户通过共享模型的参数参与模型训练，并可通过调节共享模型参数比例的方式满足不同应用场景对隐私保护的需求。在此模型中，为进一步保障用户隐私的安全性，本文还提出一种本地模型初始化协议。实验结果表明，本文的方法能有效缓解集中式存储和用户隐私泄露的压力，在保护用户隐私的同时仍能保持原来的预测精度。

**关键词** 云服务；隐私保护；分布式矩阵分解；服务质量预测

**中图法分类号** TP DOI号: *投稿时不提供DOI号

# A Distributed Matrix Factorization Model with Adjustable User Privacy for Cloud Service QoS Prediction

XU Jian-Long LIN Jian XIONG Zhi

(Department of computing, Shantou University, Shantou 515063)

**Abstract** Personalized quality of service (QoS) prediction is an important part of building high-quality cloud service system. Traditional collaborative filtering method based on centralized training mode is difficult to protect user privacy. In order to effectively protect user privacy while obtaining highly accurate prediction effect, in this paper, a distributed matrix factorization model with adjustable privacy (DMF-AP) for QoS prediction of cloud services is proposed. This model allows users to participate in model training by sharing model parameters, and can meet the privacy protection requirements of different application scenarios by adjusting the proportion of shared model parameters. In this model, a local model initialization protocol is proposed to further guarantee the security of user privacy. Experimental results show that the proposed method can effectively relieve the pressure of centralized storage and user privacy disclosure, and maintain the original prediction accuracy while protecting user privacy.

**Keywords** cloud services; privacy protection; distributed matrix factorization; QoS prediction

收稿日期： - - ；最终修改稿收到日期： - - 本课题得到国家自然科学基金基金(61702318)、广东省自然科学基金（2021A1515012527，2018A030313438）、广东省普通高校重点领域专项(2020ZDZX3073)、广东省科技专项资金(“大专项+任务清单”)项目(2019ST043)、李嘉诚基金会交叉研究项目(2020LKSFG08D)资助. 许建龙(通信作者)，男，博士，讲师，计算机学会(CCF)会员(39343M)，主要研究领域为服务计算、信息安全、数据挖掘. E-mail: xujianlong@stu.edu.cn. 手机号码：18566220754. 林健，男，硕士研究生，主要研究领域为服务计算、信息安全. E-mail: 20jlin3@stu.edu.cn. 熊智，男，博士，教授，主要研究领域为云计算、信息系统安全、并行/分布式计算. E-mail:zxiong@stu.edu.cn.

## 1 引言

随着云计算技术的发展，互联网上出现了越来越多的云服务，用户可根据需要调用这些云服务来构建高质量的云计算应用系统，然而随着云服务的数量呈指数级增长，大量等价或类似功能的候选服务应运而生，比如谷歌，亚马逊等互联网公司都通过的它们云平台为开发者们提供数以万计的云服务[1,2]。为了从众多候选服务中选择最合适的服务以满足用户个性化需求，云服务的非功能性属性即服务质量(QoS)成为主要关注的指标[3]。用户可通过调用云服务后获得其QoS值(包括吞吐量、响应时间、可靠性等)来判断云服务质量的优劣，以此来筛选合适的服务。然而一个用户如果每次都调用所有的服务再通过排序其QoS值来选择出最优QoS值，将耗费大量的代价。为了解决这个问题，有效的方法是收集大量用户调用云服务的历史QoS数据，在此基础上对未知云服务的QoS值进行预测[4]。目前，众多学者已采用了以协同过滤为主的QoS预测方法[5]，为了获取更准确的QoS值，很多学者在传统方法基础上融合了地理位置[6]、时间信息[7]、上下文信息[8]等因素，构建更加优越的预测模型。

尽管这些方法能较准确预测出未知服务的QoS值，但仍然存在以下主要问题：(1) 传统的集中式训练方法需要较高的存储成本[9]，因为集中式训练方法需要中心云服务器收集分散的用户原始数据并进行建模，但用户和云服务数量的激增产生的数以百万计的QoS值将加重云中心数据存储的压力。(2) 集中式存储原始数据具有高隐私风险[10]，因为第三方可能利用收集到的用户数据推断个人信息或将用户数据转售给其它企业以谋取利益。(3) 相关法律的颁布使收集用户数据更加困难，比如一般数据保护法规(GDPR)[11]，对收集和使用用户数据有严格的规定，要求企业或组织只能为特定目的收集和处理最低数量的个人信息，这无疑使集中式存储数据带来了挑战。此外，对于分布式用户来说，在训练预测模型的过程中，若分享原始数据则存在高风险泄露隐私，若分享模型权重或梯度参数，则可降低风险，但难以权衡分享参数与预测精度，因此需要具备在不同场景对隐私程度进行调节的能力。

针对以上问题，本文提出一种面向云服务QoS预测的分布式、可调节隐私程度的矩阵分解模型DMF-AP(Distributed Matrix Factorization with Adjustable Privacy) 。本模型是分布式的，即用户数据和模型训练都各自在本地，用户间无需共享原始数据，只需用户间共享模型梯度来学习个性化的预测模型。由于不同的数据共享对隐私的影响是不一样的，相比于共享原始数据，共享模型的权重或梯度使得第三方提取用户敏感信息更加困难，降低了隐私泄露的等级[10]。特别指出的是，本模型梯度交换的数量是可以调节的，这可帮助用户平衡隐私和预测精度。

本文的主要贡献包括：(1) 提出一种分布式、可调节隐私程度的QoS预测模型，可以有效解决集中式训练方法的存储和用户隐私泄露的问题。(2) 提出了一种本地模型初始化协议，该协议规定本地模型在每轮训练前如何选取最新的全局模型参数。(3) 在真实的数据集中通过大量实验来验证本文提出方法的有效性。

本文其余部分安排如下：第2节介绍和本文相关的研究工作；第3节为本文提出的模型描述；第4节为本文提出的模型的隐私分析；第5节实验结果及分析；最后总结全文并指出未来的研究工作。

## 2 相关工作

### 2.1 基于协同过滤的QoS预测方法

目前常用的QoS预测方法是协同过滤方法，它可分为基于内存的协同过滤[12–14]和基于模型的协同过滤[2,15,16]。基于内存的协同过滤利用用户感知服务的QoS值来计算用户和服务间的相似性。典型的例子包括了基于相似用户的协同过滤[12]，基于相似服务的协同过滤[13]和基于用户-服务相似的混合协同过滤[14]。Zheng等人[14]提出了一种基于用户和服务相似度的混合的方法预测web服务的QoS值。尽管基于内存的协同过滤方法实现简单，然而该方法在数据稀疏的情况下的QoS预测的准确性较低，难以给出可靠的云服务推荐。基于模型的协同过滤的方法的出现有效地缓解了这个问题。矩阵分解[15,16]是一种普遍应用在服务推荐领域的基于模型协同过滤的技术。矩阵分解的原理是将用户-服务交互矩阵分解成两个低维的矩阵乘积。Zheng等人[15]提出了一种融合了基于邻域协同过滤并结合矩阵分解的方法为用户提供个性化的QoS预测。Zhang等人[16]提出了一个在云环境下基于矩阵分解的可信的在线QoS预测的方法。

### 2.2 隐私保护

尽管协同过滤技术可以给用户带来许多好处，但用户敏感数据的收集会导致越来越多的用户不愿意提供自己的历史数据。因此在为用户推荐高质量的服务的同时保护用户的隐私成了一个重要的问题。为了解决这个问题，已有很多学者提出了解决方案，比如利用了混淆数据方法[1,17,18]，密码学方法[19–21]，匿名化技术[22]，局部敏感哈希技术[23]或者采用联邦学习技术[24]。Zhu 等人[1]提出了一个应用数据混淆技术的隐私保护方法, 该方法通

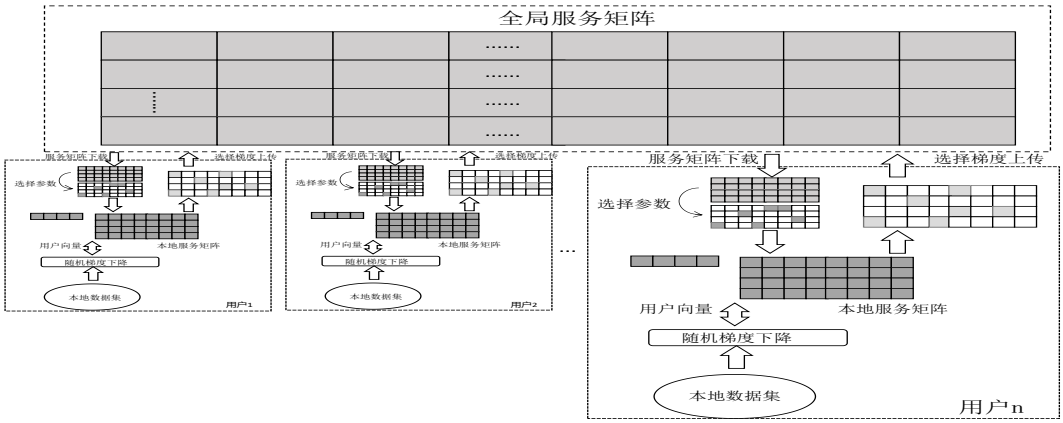



图1　具备可调节隐私的分布式矩阵分解模型图

过给每个用户观察到的QoS 值添加一个随机的噪声，然后利用基于协同过滤的方法进行预测未知的QoS值。Badsha等人[19]提出了一个利用同态加密技术保护用户的QoS值和用户位置的协议，该协议允许第三方在加密后的数据上进行计算并通过协同过滤技术预测QoS值。Qi 等人[23]提出了一个考虑时间、空间上下文的的QoS 隐私保护方法，该方法首先通过局部敏感哈希(LSH) 技术哈希原来的数据，然后通过基于内存的协同过滤方法处理哈希后的QoS数据。然而以上这些方法都是采用集中式的训练方法，集中式训练需要存储大量用户相关的数据(比如加密或混淆后的QoS值)，这会产生昂贵的存储费用。Zhang 等人[24]提出了将联邦学习应用到矩阵分解方法中，这是一种分布式的方法，该方法允许用户不传递原始数据，而是通过传递模型参数协同训练一个全局模型。尽管该方法没有直接泄露隐私，但完整的传递模型参数给第三方被认为是有风险的，因为第三方可以从共同训练的预测模型中推断出用户的原始数据。

## 3　模型

### 3.1　模型概述

图1为本文提出的具有可调节隐私的分布式矩阵分解QoS预测模型示意图，图中展示了模型的主要组件和组件间的交互过程。这些组件包括参数服务器(存储全局服务矩阵)和客户端(即用户)，其中参数服务器的作用是处理客户端请求和维护全局服务矩阵的参数更新。

模型中假设有n个客户端，每个客户端都有自己本地数据集，并且都有相同的矩阵分解模型结构和共同的优化目标。客户端利用本地数据训练从参数服务器下载的最新全局服务矩阵和保存在本地的用户向量。整个训练过程分为以下3个步骤：(1) 某个客户端向参数服务器发送请求下载全局服务矩阵，参数服务器收到请求后并发送最新的全局服务矩阵给客户端，客户端选取一定比例的全局服务矩阵元素重写本地矩阵分解模型的服务矩阵，其作用是若存在恶意的攻击者，它们也很难从客户端共享的参数中推断出客户端的敏感信息，因为它们并不能具体的知道客户端选取哪些参数初始化本地模型。(2) 客户端利用本地数据对用户向量和重写后的本地服务矩阵采用梯度下降法进行训练，训练后将满足梯度上传条件的本地服务梯度上传至参数服务器，参数服务器将接收到的梯度更新全局服务矩阵，这样就完成一轮的训练，如此训练直至模型收敛。(3) 当训练完成后，客户端可通过本地模型独自预测自己与未知服务的QoS 值。

可以看出，整个过程中，客户端不需要传递原始数据给第三方，只需要上传满足上传条件的服务梯度至参数服务器，这将有效保护客户端的隐私。特别指出的是，一方面，客户端不需要上传全部梯度，另一方面若上传梯度，其上传的比例是可以调节的。这样做的目的同样是为了防止恶意攻击者推断出客户端的敏感信息。而且，该模型也可以调节本地服务矩阵重写的比例进一步的保护客户端隐私。

### 3.2　客户端训练

假设云服务预测模型有$n$个客户端，$m$个服务，由于每个客户端的训练过程都是一样的，不失一般性，本文选取客户端$i$为例。客户端$i$存储了本地QoS数据集$R_i$以及本地的矩阵分解模型，本地模型参数包括用户向量$U_i$和本地服务矩阵$S^i$，于是客户端$i$本地的矩阵模型可以用以下公式表示：

$$R_i \approx U_i^T S^i \tag{1}$$

其中，$R_i \in 1 \times m, U_i \in d \times 1, S^i \in d \times m$ 。客户端在第一轮训练开始前，需要初始化的本地模型参数包括用户向量$U_i$和本地服务矩阵$S^i$。用户向量$U_i$由

客户端随机初始化，本地服务矩阵初始化需要完成两个步骤：(1)下载来自参数服务器最新的全局服务矩阵$S^{global}$；(2)将全局服务矩阵$S^{global}$覆盖本地服务矩阵。而在接下来的每轮训练前的本地服务矩阵的初始化并不是将下载的最新的全局服务矩阵直接覆盖本地服务矩阵，而是选择一定比例$\theta_o$的全局服务矩阵元素重写本地服务矩阵。

本文通过二进制掩码矩阵$Mask_o^i$为每个客户端选择满足重写条件的全局服务矩阵元素。步骤如下：第一步，用全局服务矩阵$S^{global}$减去本地服务矩阵$S^i$得到一个差值矩阵$d^i$，并对矩阵求绝对值，得到矩阵$D^i$，公式如下：

$$D^i = |S^{global} - S^i| \tag{2}$$

第二步，将矩阵$D^i$的元素从大到小排序，并优先记录比较大的元素的索引(元素的行和列的位置)，直至比例满足$\theta_o$，第三步，将索引对应的二进制掩码矩阵$Mask_o^i$元素赋值为1，其他元素为0。第四步，将赋值后的掩码矩阵$Mask_o^i$和全局矩阵对应元素相乘得到满足重写条件的全局服务矩阵元素，矩阵全局服务矩阵被选取的矩阵元素可以用式子$Mask_o^i \odot S^{global}$表示，其中⊙表示元素乘法。

假设ONE为元素全为1的矩阵，重写后的本地矩阵表示为：

$$S^i \leftarrow (ONE - Mask_o^i) \odot S^i + Mask_o^i \odot S^{global} \tag{3}$$

其中, ONE$\in d \times m$。值得注意的是，重写比例$\theta_o$是可以调节的，一般来说，重写的比例越高预测的精度也越高，但同时用户隐私泄露的风险也就越大，所以需要平衡用户隐私和预测精度。通过以上步骤就完成了本地模型的初始化。

接着，客户端利用本地数据(即QoS值)，训练保存在本地的用户向量$U_i$和被重写后的本地服务矩阵$S^i$。由于本地矩阵分解模型的优化目标是缩小式(1)中的误差，为了避免过拟合，通常需要加入正则项，于是用户$i$本地矩阵分解模型的损失函数表示为：

$$L_i = \frac{1}{2}\sum_{j=1}^{m} I_{ij}(R_{ij} - U_i^T S_j^i) + \frac{\lambda_U}{2}\|U_i\|_F^2 + \frac{\lambda_S}{2}\|S^i\|_F^2 \tag{4}$$

其中，$j \in \{1, 2, ..., m\}$, $I_{ij}$的作用是指示该QoS值是否被观察到，如果$I_{ij}$=1表示该QoS值被观察到，如果$I_{ij}$=0表示该QoS值没有被观察到，$\lambda_U, \lambda_S$表示正则化的程度。为了加快本地矩阵分解模型收敛，本文采用随机梯度下降(SGD)[25]来训练本地模型。对于客户端i调用服务j的每一个观察到的QoS值，都有以下损失函数：

$$l_j^i = \frac{1}{2} I_{ij}(R_{ij} - U_i^T S_j^i) + \frac{\lambda_u}{2}\|U_i\|_F^2 + \frac{\lambda_s}{2}\|S_j^i\|_F^2 \tag{5}$$

其中，$L_i = \sum_{j=1}^{m} I_{ij} l_j^i$，所以$U_i$，$S_j^i$的更新如公式(5)和(6)所示：

$$U_i \leftarrow U_i - \eta \frac{\partial l_j^i}{\partial U_i} \tag{6}$$

$$S_j^{i,t+1} \leftarrow S_j^{i,t} - \eta \frac{\partial l_j^i}{\partial S_j^{i,t}} \tag{7}$$

其中，$j \in \{1, 2, ..., m\}, S_j^{i,t}, S_j^{i,t+1}$分别为训练前后的本地服务向量，$\eta$ 为学习率，控制每一轮训练的参数的变化。训练结束后，需要上传部分梯度矩阵元素至参数服务器。本文通过二进制掩码矩阵$Mask_u^i$为每个客户端选择满足上传条件的梯度矩阵元素。首先计算训练后的本地服务矩阵元素和训练前的本地服务矩阵元素的梯度矩阵$g_i$，公式为：

$$g^i \leftarrow S^{i,t} - S^{i,t+1} \tag{8}$$

其次，计算梯度矩阵的绝对值$G_i$，将矩阵$G_i$的元素从大到小排列，优先记录比较大的元素的索引(元素的行和列的位置)，直至比例满足$\theta_u$，并将索引对应的的二进制掩码矩阵$Mask_u^i$元素赋值为1，其他元素为0。于是被上传的梯度矩阵元素可表示为：

$$g^i \leftarrow Mask_u^i \cdot g^i \tag{9}$$

这样客户端就完成了一轮的训练，训练的伪代码如算法1所述。

**算法1**. 客户端训练算法.

输入：客户端i模型$(U_i, S^i)$ 全局服务矩阵 重写比例$\theta_o$ 上传比例$\theta_u$

输出：被选择服务梯度矩阵元素

1. $d^i$ = $S^{global}$-$S^i$//差值矩阵$d^i$,全局服务矩阵$S^{global}$本地服务矩阵$S^i$
2. $D^i$ = | $d^i$ | //对矩阵取绝对值
3. $D^i$的元素从大到小排序
4. 优先记录比较大的元素的索引（元素的行和列的位置），直至比例满足$\theta_o$
5. 将索引对应的的二进制掩码矩阵$Mask_o^i$元素赋值为1，其他元素为0
6. $S^i \leftarrow (ONE - Mask_o^i) \odot S^i + Mask_o^i \odot S^{global}$//重写本地服务矩阵
7. 客户端利用本地数据通过梯度下降法训练本地服务矩阵和本地用户向量
8. $g^i \leftarrow S^{i,t} - S^{i,t+1}$//计算梯度矩阵$g^i$，$S^{i,t}, S^{i,t+1}$分别为训练前后的本地服务矩阵
8. $G^i$=| $g^i$ | //对矩阵取绝对值
9. 优先记录比较大的元素的索引（元素的行和列的位置），直至比例满足$\theta_o$
10. 将索引对应的的二进制掩码矩阵$Mask_o^i$元素赋值为1，其他元素为0
11. $g^i \leftarrow Mask_u^i \cdot g^i$//$g^i$为要上传的梯度矩阵元素

### 3.3 参数服务器

参数服务器主要有两个任务，一个是处理客户端上传服务梯度和下载服务矩阵的请求，一个是维

护全局服务矩阵。当客户端开始训练本地模型前，需向参数服务器请求下载最新的全局服务矩阵，参数服务器收到请求消息后，返回最新的全局服务矩阵。当客户端训练结束后，会向参数服务器发送满足上传条件服务梯度，然后参数服务器将客户端的上传的梯度更新全局服务矩阵，这样参数服务器和客户端就完成了一次模型的更新,公式如下：

$$S^{global} \leftarrow S^{global} - g^{i} \tag{10}$$

## 4 隐私分析

Zhu等人[26]认为在分布式系统中共享梯度是有风险的，第三方可根据客户端共享的梯度和初始化的模型参数推理出客户端原始的数据，并且认为梯度压缩[27]是一个能够避免攻击者通过模型梯度和模型参数推理出客户端原始数据的实用的方法。也就是说，造成隐私泄露可能的原因有两点：（1）客户端共享的梯度的信息；（2）客户端初始化的模型参数。另一方面，可以通过破坏共享信息的完整性降低隐私泄露的风险。于是本文提出了两个方案通过破坏共享信息的完整性弱化了这两个必要条件以防止攻击者有效的重构原始数据。首先，本文允许客户端只上传部分梯度，而不是完整的梯度信息，这使得客户端的完整的梯度信息可以得到保护，弱化了潜在的隐私泄露的风险。其次，允许客户端在每轮初始化本地模型时，只覆盖本地部分模型参数，而不是覆盖本地全部模型参数，这使得第三方从客户端的获得的可重构的信息进一步减少，加大了第三方重构原始数据的难度。客户端可通过调整共享梯度信息和初始化本地模型参数的数量平衡隐私保护等级和精度。

## 5 实验

本节将通过实验来验证本文提出方法的有效性，重点关注以下几个问题：

问题1：相比于传统集中式的矩阵分解，本文提出的模型的预测精度怎么样？

问题2：如何平衡模型预测精度和用户隐私的关系？

问题3：相比于传统集中式的矩阵分解，本文模型的效率如何？

### 5.1 数据集

本文采用由Zheng等人[28]收集和维护的WS-DREAM数据集进行实验。该数据集由339个用户，5825服务组成。用户分布在30个国家，服务分布在73个国家。本文重点关注关于响应时间(RT)的子数据集。RT数据集包含了1,974,675条QoS记录。响应时间指的是发送请求到接收响应的时间间隔，响应时间属性的范围为0-20秒。

### 5.2 评估指标

为了衡量本文模型的性能，本文通过以下两个评估指标来比较集中式梯度下降的矩阵分解模型和本文提出的模型的表现。这两个评估指标分别为MAE(平均绝对误差)，RMSE(均方根误差)。评估指标的值越小，表示该模型的性能越好。

$$MAE = \frac{\sum_{I_{ij}} | \hat{R}_{ij} - R_{ij} |}{N} \tag{11}$$

其中$I_{ij} = 0$，$N$为数据集中$I_{ij} = 0$ 的数量，$\hat{R}_{ij}$为第$i$个用户感知第$j$个服务的预测QoS值，$R_{ij}$为第$i$个用户感知第j个服务的真实QoS值。

$$RMSE = \sqrt{\frac{\sum_{I_{ij}} (\hat{R}_{ij} - R_{ij})^2}{N}} \tag{12}$$

其中$I_{ij} = 0$，N为数据集中$I_{ij} = 0$ 的数量，$\hat{R}_{ij}$为第i个用户感知第$j$个服务的预测QoS值，$R_{ij}$为第$i$个用户感知第j个服务的真实QoS值。由于RMSE等于各个QoS预测值与真实值的差值的平方的均值，所以可以通过RMSE的大小来衡量模型的稳定性。

### 5.3 参数设置

本文的实验以集中式的使用SGD方法的矩阵分解模型[29]，简称CMF-SGD(Centralised Matrix Factorization Model using SGD Method)与本文提出的面向云服务QoS预测的分布式、可调节隐私的矩阵分解模型(DMF-AP)做比较。SGD算法[25]允许每轮训练只从训练集中随机抽取小批量数据计算梯度，这种算法可以加快模型收敛速度。最简单的情况是在每轮训练随机选择一个数据样本进行训练。DMF-AP 模型的梯度上传方法有两种，第一种是本地服务梯度矩阵元素的绝对值从大到小排序，绝对值较大的元素优先上传，直至上传的元素数量占全部本地服务矩阵元素的比例为$\theta_u$。第二种将本地服务梯度矩阵元素的绝对值大于某个阈值的元素上传至参数服务器，阈值大小可以自由调整。为了方便观察实验结果与上传比例的关系，本文的实验采取第一种梯度上传方法。另一方面，DMF-AP 模型的本地服务矩阵初始化方法采取本地服务矩阵与全局服务矩阵差值较大的元素重写本地服务矩阵元素的方法，直至满足重写比例$\theta_o$。本文参数设置为，梯度上传比例$\theta_u = \{1, 0.1, 0.01, 0.001\}$，重写比例$\theta_o = \{1, 0.5, 0.2, 0.1\}$，在每轮训练选取的本地批处理样本大小设为1，隐含特征维度设为6。在实际场景下，单个用户调用的云服务只占总的云服务很少的比例，所以为了模型实际场景，本文将实验训练集的密度设为$\{2.5\%, 5\%, 7.5\%, 10\%\}$。选取10%训练数据集分为两个步骤：（1）从原数据

表1 不同模型在RT上的预测精度比较表

| 方法 | Matrix Density = 2.5% | | Matrix Density = 5.0% | | Matrix Density = 7.5% | | Matrix Density = 10% | |
|---|---|---|---|---|---|---|---|---|
| | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE |
| CMF-SGD | 0.7767 | 1.6360 | 0.6023 | 1.4116 | 0.5446 | 1.3313 | 0.5166 | 1.2900 |
| DMF-AP | 0.7754 | 1.6302 | 0.5969 | 1.4011 | 0.5402 | 1.3261 | 0.5180 | 1.2853 |
| 增益 | +0.17% | +0.35% | +0.90% | +0.74% | +0.81% | +0.39% | -0.27% | +0.36% |

表2 DMF-AP模型在RT上不同梯度上传比例的预测精度

| 上传比例 | Matrix Density = 2.5% | | Matrix Density = 5.0% | | Matrix Density = 7.5% | | Matrix Density = 10% | |
|---|---|---|---|---|---|---|---|---|
| | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE |
| 1 | 0.7829 | 1.6400 | 0.5969 | 1.4013 | 0.5398 | 1.3230 | 0.5241 | 1.2946 |
| 0.1 | 0.7754 | 1.6302 | 0.5969 | 1.4011 | 0.5402 | 1.3261 | 0.5180 | 1.2853 |
| 0.01 | 0.8145 | 1.6562 | 0.6626 | 1.4251 | 0.6142 | 1.3692 | 0.6020 | 1.3187 |
| 0.001 | 0.9145 | 1.7728 | 0.8324 | 1.5701 | 0.7990 | 1.5211 | 0.7980 | 1.5125 |

集中随机抽取占总样本数量的10%；（2）由于每个数据是由（用户，服务，QoS）三元组组成的，本文将用户元素相同的元组放在同一个用户数据集里。其它的90% 数据集样本作为测试集用来验证模型的性能。本实验将传统集中式梯度下降模型和本文所提出来的模型的学习率$\eta$ 设为0.01，正则化程度$\lambda_u, \lambda_s$ 都设为0.1。

### 5.4 预测精度(问题1)

在本节中，通过实验比较本文提出的模型（DMF-AP)与集中式的使用SGD方法的矩阵分解模型（CMF-SGD)的预测精度。本实验中设置DMF-AP模型的梯度上传比例$\theta_u$=0.1，本地服务矩阵重写比例$\theta_o$=1。表1为本文提出的模型（DMF-AP)与集中式的使用SGD方法的矩阵分解模型(CMF-SGD)在响应时间(RT)数据集上的实验结果。从表1中可看出，本文提出的模型(DMF-AP)与CMF-SGD模型相比，在不同数据集密度上两项评估指标都非常接近(相差约0.01)，而且本文提出的模型在响应时间RT数据集除了在密度10%时的MAE高于CMF-SGD模型0.27%，其余情况下，模型的预测精度都优于CMF-SGD模型，这可能是由于用户间共享部分关键梯度起到了正则化的效果，使共同训练的模型在少量训练数据的情况下可以避免过拟合现象，特别是在训练密度较低的场景下。因此，与传统矩阵分解模型相比，DMF-AP 模型可在提高预测精度的情况下同时提供一定程度的隐私保护。

### 5.5 调整梯度上传和服务矩阵重写比例(问题2)

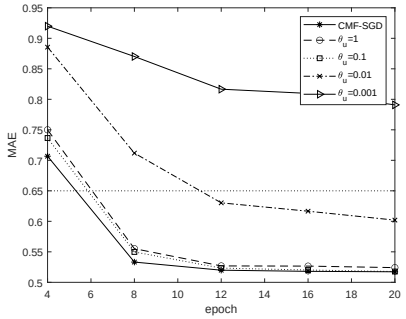
为了平衡预测精度和用户隐私，本实验分别调整梯度上传比例$\theta_u$和本地服务矩阵重写比例$\theta_o$，并观察在不同参数下DMF-AP模型的预测精度。本实验先观察梯度上传比例$\theta_u$对模型预测精度的影响，设置DMF-AP模型的梯度上传比例$\theta_u = \{1, 0.1, 0.01, 0.001\}$，重写比例$\theta_o$=1。表2为在固定重写比例$\theta_o$的情况下，不同梯度上传比例$\theta_u$在响应时间（RT）数据集上的实验结果。

从表2中可看出，梯度上传比例为0.1时，在两项评估指标上的值和上传比例为1时的值特别接近(相差约0.01)，而且在某些密度下的预测误差是最小的，这可能是因为梯度上传比例为1时造成了预测模型的过拟合。另一方面，当梯度比例上传小于0.1，随着上传梯度比例越低，预测误差越高。表2的实验结果说明了用户共享部分梯度和共享全部梯度的预测精度说接近的，甚至有时共享部分梯度的预测精度还好于共享全部梯度。因此，在同时考虑用户隐私、预测精度和计算效率的情况下，对于RT数据集，可将$\theta_u$设为0.1比较合适。

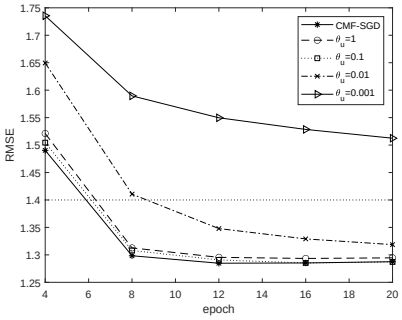
对于参数重写比例$\theta_o$对DMF-AP模型预测精度的影响来说，本实验设置本地服务矩阵元素重写比例为$\theta_o = \{1, 0.5, 0.2, 0.1\}$，对于响应时间RT的预测任务，梯度上传比例$\theta_u$设为0.1。表3为在固定梯度上传比例$\theta_u$的情况下，不同重写比例$\theta_o$在响应时间(RT)数据集上的实验结果。从表3中可以看出，

表3　DMF-AP模型在RT上不同参数重写比例的预测精度

| 重写比例 | Matrix Density = 2.5% | | Matrix Density = 5.0% | | Matrix Density = 7.5% | | Matrix Density = 10% | |
|---|---|---|---|---|---|---|---|---|
| | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE |
| 1 | 0.7754 | 1.6302 | 0.5969 | 1.4011 | 0.5402 | 1.3261 | 0.5180 | 1.2853 |
| 0.5 | 0.8312 | 1.6396 | 0.6112 | 1.3948 | 0.5507 | 1.3204 | 0.6389 | 1.3320 |
| 0.2 | 0.8634 | 1.6901 | 0.6824 | 1.4473 | 0.6318 | 1.3619 | 0.7934 | 1.5110 |
| 0.1 | 0.9131 | 1.7570 | 0.7945 | 1.5151 | 0.7696 | 1.4509 | 0.8599 | 1.5744 |




(a)　RT，$\theta_o$=1，batch_size=1，MD=10%，MAE




(b)　RT，$\theta_o$=1，batch_size=1，MD=10%，RMSE

图2　不同的梯度上传比例在RT上的收敛

随着重写比例的降低，模型的预测精度也逐渐减低。因此，可通过降低重写比例提高用户隐私保护的程度，但同时会导致预测精度的降低。

### 5.6　模型效率(问题3)

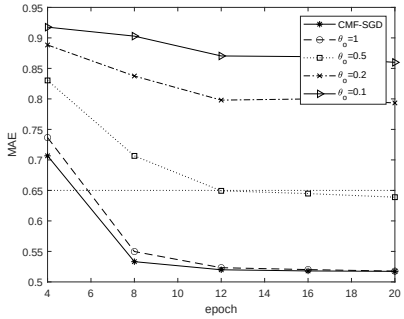
本节将通过调整不同梯度上传比例和参数重写比例观察模型的效率，并采用模型的误差到达预期误差所需最小迭代次数作为衡量模型效率的指标。

实验分两部分：(1) 检验不同梯度上传比例$\theta_u$对模型效率的影响。在响应时间RT预测任务中，设置本地服务矩阵重写比例$\theta_o$=1，预期误差MAE=0.65，RMSE=1.4，训练集密度MD=10%。图2为CMF-SGD模型和不同梯度上传比例的DMF-AP模型在响应时间（RT）数据集上的预测误差随训练迭代次数变化的情况。从图2中可以发现，随着迭代次数的增加CMF-SGD模型和DMF-AP模型的MAE和RMSE均呈现下降趋势，CMF-SGD和DMF-AP(梯度上传比例为1和0.1)的到达预期误差所需的训练迭代的次数约为6次，并且它们变化曲线基本重合。这说明本文提出的模型效率和集中式梯度下降的矩阵分解模型的效率基本相同。

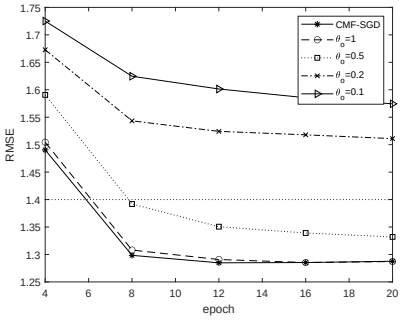
(2)检验不同参数重写比例$\theta_o$对DMF-AP模型效率的影响。在响应时间RT预测任务中，设置梯度上传比例$\theta_u$=0.1，预期误差MAE=0.65, RMSE=1.4，训练集密度MD=10%。图3为CMF-SGD模型和不同参数重写比例的DMF-AP模型在响应时间(RT)数据集上的训练迭代次数和预测误差的关系。从图3可看出，随着迭代次数的增加CMF-SGD模型和DMF-AP模型的MAE和RMSE均呈现下降趋势，模型参数重写比例为0.5的DMF-AP模型到达指定MAE的训练迭代次数增加到12次，RMSE则为8次，而当模型参数重写比例为0.2，0.1时的DMF-AP模型则无法到达预期误差。这说明了随着DMF-AP模型参数重写比例$\theta_o$的降低，模型的效率开始降低。

## 6　结论及展望

本文针对云服务个性化QoS预测中用户的隐私保护问题，提出可调节隐私保护程度的分布式矩阵分解模型，通过用户间共享模型参数来学习个性化的预测模型，用户可通过调节模型共享参数的比例对隐私和预测精度的关系进行平衡。在真实数据集中进行的大量实验结果表明，该模型在保护用户隐私的同时依然能保持原有QoS 预测精度，证明本文方法的有效性。在未来的工作中，将探索通过同态加密、差分隐私等技术加强隐私保护程度，此外，还将考虑在线情况下的用户隐私保护问题，构建在线环境下具有用户隐私保护的预测模型。




(a) RT，$\theta_u$=0.1，batch_size=1，MD=10%，MAE




(b) RT，$\theta_u$=0.1，batch_size=1，MD=10%，RMSE

图3 不同的模型参数重写比例在RT上的收敛

## 参 考 文 献


[1] Zhu J, He P, Zheng Z, Lyu M R. A privacy-preserving QoS prediction framework for web service recommendation//Proceedings of the 2015 IEEE International Conference on Web Services, New York, USA, 2015: 241-248, doi: 10.1109/ICWS.2015.41.

[2] Chen Z, Sun Y, You D, Li F, Shen L. An accurate and efficient web service QoS prediction model with wide-range awareness. Future Generation Computer Systems, 2020, 109: 275-292

[3] Zhang Y, Pan J, Qi L, He Q. Privacy-preserving quality prediction for edge-based IoT services. Future Generation Computer Systems, 2021, 114: 336-348

[4] Liu J, Chen Y. A personalized clustering-based and reliable trust-aware QoS prediction approach for cloud service recommendation in cloud manufacturing. Knowledge-Based Systems, 2019, 174: 43-56, https://doi.org/10.1016/j.knosys.2019.02.032.

[5] Wu D, He Q, Luo X, Shang M, He Y, Wang G. A posterior-neighborhood-regularized latent factor model for highly accurate web service QoS prediction. IEEE Transactions on Services Computing, doi: 10.1109/TSC.2019.2961895.

[6] Yang Y, Zheng Z, Niu X, Tang M, Lu Y,Liao X. A location-based factorization machine model for web service QoS prediction. IEEE Transactions on Services Computing, doi: 10.1109/TSC.2018.2876532.

[7] Li J, Wang J, Sun Q, Zhou A. Temporal Influences-Aware Collaborative Filtering for QoS-Based Service Recommendation. In: 2017 IEEE International Conference on Services Computing (SCC), 2017: 471-474. IEEE(2017). doi: 10.1109/SCC.2017.67.

[8] Wu H, Yue K, Li B, Zhang B, Hsu C. Collaborative QoS prediction with context-sensitive matrix factorization. Future Generation Computer Systems, 2018, 82: 669-678,. https://doi.org/10.1016/j.future.2017.06.020.

[9] Chen C, Liu Z, Zhao P, Zhou J, Li X. Privacy Preserving Point-of-Interest Recommendation Using Decentralized Matrix Factorization. Proceedings of the AAAI Conference on Artificial Intelligence, 2018:32(1).

[10] Duriakova E, Tragos E. Z., Smyth B, et al. PDMFRec: a decentralised matrix factorisation with tunable user-centric privacy. //Proceedings of the 13th ACM Conference on Recommender Systems (RecSys '19). Association for Computing Machinery, New York, NY, USA, 2019: 457 – 461. DOI:https://doi.org/10.1145/3298689.3347035

[11] Voigt P, Von dem Bussche A. The eu general data protection regulation (gdpr). A Practical Guide, 1st Ed., Cham: Springer International Publishing, 2017.

[12] Shao L, Zhang J, Wei Y, Zhao J, Xie B, Mei H. Personalized QoS prediction for web services via collaborative filtering. //Proceedings of the IEEE International Conference on Web Services (ICWS), Salt Lake City, UT, USA, 2007: 439 – 446.

[13] Sarwar B. M, Karypis G, Konstan J. A, Riedl J. Item-based collaborative filtering recommendation algorithms. Proceedings of the 10th International World Wide Web Conference (WWW), New York, NY, USA, 2001: 285 – 295.

[14] Zheng Z, Ma H, Lyu M. R, King I. QoS-aware web service recommendation by collaborative filtering. IEEE Transaction on Services Computing, 2011, 4(2): 140 – 152

[15] Zheng Z, Ma H, Lyu M. R, King I. Collaborative Web service QoS prediction via neighborhood integrated matrix factorization. IEEE Transaction on Services Computing, 2013, 6(3): 289 – 299.

[16] Zhang Y, Zhang X, Zhang P, Luo J. Credible and online qos prediction for services in unreliable cloud environment. 2020 IEEE International Conference on Services Computing (SCC), Beijing, China, 2020: 272-279, doi: 10.1109/SCC49832.2020.00043.

[17] Zhu X, et al. Similarity-maintaining privacy preservation and location-aware low-rank matrix factorization for qos prediction based web service recommendation. IEEE Transactions on Services Computing, 2021, 14(3): 889-902, doi: 10.1109/TSC.2018.2839741

[18] Zhang Peng-Cheng, Jin Hui Ying. A forward privacy-preserving QoS prediction method for mobile edge environments. Journal of Computer Science,2020,43(08):1555-1571 (in Chinese)

(张鹏程,金惠颖. 一种移动边缘环境下面向隐私保护QoS预测方法. 计算机学报,2020,43(08):1555-1571)
[19] Badsha S, et al. Privacy preserving location-aware personalized web service recommendations. IEEE Transactions on Services Computing, 2021, 14(3): 791-804, doi: 10.1109/TSC.2018.2839587
[20] Badsha S, Yi X, Khalil I, Liu D, Nepal S,Lam K. Privacy preserving user based web service recommendations. IEEE Access, 2018, 6: 56647-56657, doi: 10.1109/ACCESS.2018.2871447.
[21] Rahman M S, Khalil I, Alabdulatif A, Yi X. Privacy preserving service selection using fully homomorphic encryption scheme on untrusted cloud service platform. Knowledge-Based Systems, 2019, 180:104-115, https://doi.org/10.1016/j.knosys.2019.05.022.
[22] Zhang S, Liu Q, Lin Y, Anonymizing popularity in online social networks with full utility. Future Generation Computer Systems. 2017, 72: 227 – 238.
[23] Qi L, Zhang X, Li S, Wan S, Wen Y, Gong W. Spatial-temporal data-driven service recommendation with privacy-preservation. Information Sciences, 2020, 515: 91-102, https://doi.org/10.1016/j.ins.2019.11.021.
[24] Zhang Y, Zhang P, Luo Y, Luo J. Efficient and Privacy-Preserving Federated QoS Prediction for Cloud Services. Proceedings of the 2020 IEEE International Conference on Web Services (ICWS), 2020, pp. 549-553, doi: 10.1109/ICWS49710.2020.00079.
[25] Shapiro A, Wardi Y. Convergence analysis of gradient descent stochastic algorithms. Journal of Optimization Theory and Applications, 1996 pp. 45 – 4.
[26] Zhu L, Han S. Deep Leakage from Gradients. In: Yang Q., Fan L., Yu H. (eds) Federated Learning. Lecture Notes in Computer Science, vol 12500. Springer, Cham. https://doi.org/10.1007/978-3-030-63076-8_2
[27] Lin Y, Han S, Mao H, Wang Y, Dally W. Deep gradient compression: Reducing the communication bandwidth for distributed training. arXiv preprint arXiv:1712.01887, 2017. 8, 9
[28] Zheng Z, Zhang Y, Lyu M. R. Investigating QoS of real world Web services. IEEE Transaction on Services Computing, 2014, 7(1): 32 – 39
[29] Salakhutdinov R, Mnih A. Probabilistic matrix factorization. In Proceedings of the 20th International Conference on Neural Information Processing Systems (NIPS'07). Curran Associates Inc., Red Hook, NY, USA, 2007:1257 – 1264.

**XU Jian-Long** Ph.D., lecturer. His research interests include service computing, information security and Data mining.

**LIN Jian** M.S., candidate. His research interests include service computing and information security.

**XIONG Zhi** Ph.D., professor. His research interests include cloud computing, information system security, parallel and distributed computing.

## Background

With the development of cloud computing technology, more and more cloud services appear on the Internet. Users can call these cloud services according to their needs to build high-quality cloud computing application system. However, with the exponential growth of the number of cloud services, a large number of candidate services with equivalent or similar functions arise at the right moment, such as Google, Amazon and other Internet companies use their cloud platforms to provide developers with tens of thousands of cloud services. In order to select the most appropriate service from many candidate services to meet the personalized needs of users, the non-functional attribute of cloud services, namely quality of service (QoS), has become the main index of concern.Users can judge the quality of cloud service by obtaining its QoS value (including throughput, response time, reliability, etc.) after calling the cloud service, so as to screen the appropriate service. However, it would be costly for a user to call all the services each time and then sort their QoS values to select the optimal QoS value. In order to solve this problem, an effective method is to collect a large number of historical QoS data of users' calls to cloud services, and predict the QoS value of unknown cloud services on this basis.At present, many scholars have adopted the QoS prediction method based on collaborative filtering. In order to obtain more accurate QoS value, many scholars have integrated factors such as geographical location, time information and context information on the basis of traditional methods to build a more superior prediction model.

To address this problem, a number of solutions have been proposed in the literature, such as the use of obfuscated data methods, cryptographic methods, anonymisation techniques or locally sensitive hashing techniques all of which can protect user privacy to some extent. However, most of these methods use a centralised training approach. Although these methods can accurately estimate the unknown service QoS values, but there are still many problems as follows: (1) the traditional centralized training approaches require higher storage costs because centralized training approaches require central cloud servers to collect and model decentralized raw user data, but the proliferation of users and cloud services generates millions of QoS values that will add to the pressure on cloud-centric data storage. (2) Centralized storage of raw data has a high privacy

risk, because third parties may use the collected user data to infer personal information or resell user data to other enterprises for profits. (3) The promulgation of relevant laws makes it more difficult to collect user data. For example, the General Data Protection Regulation (GDPR), which has strict regulations on the collection and use of user data, requires enterprises or organizations to collect and process only the minimum amount of personal information for specific purposes, which undoubtedly brings challenges to centralized data storage. In addition, for distributed users, in the process of training model, if share the raw data, high privacy, if share model weights or gradient parameters, can reduce the risk, but difficult to weigh the share parameters and the prediction precision, therefore need to be adjusted to the degree of privacy in different scenarios of ability. To effectively address these two issues, we propose a distributed, matrix decomposition model with adjustable privacy levels. The model is distributed, with user data and model training being local. Users do not need to upload data to a third party, but only need to exchange local model parameters with the third party, thus improving the prediction accuracy of the model. In particular, the number of model parameters exchanged is adjustable, which helps the model to balance user privacy and prediction accuracy.

To effectively address these two issues, this paper propose a distributed, matrix decomposition model with adjustable privacy levels. The model is distributed, with user data and model training being local. Users do not need to upload data to a third party, but only need to exchange local model parameters with the third party, thus improving the prediction accuracy of the model. In particular, the number of model parameters exchanged is adjustable, which helps the model to balance user privacy and prediction accuracy.

This research was financially supported by the National Natural Science Foundation of China (61702318), Natural Science Foundation of Guangdong Province (2021A1515012527, 2018A030313438), Key Fields Special Project of Guangdong Universities (2020ZDZX3073), and in part by 2020 Li Ka Shing Foundation Cross-Disciplinary Research Grant (2020LKSFG08D).