

# 面向服务的软件工程中 QoS 研究课题报告

2018 级 赵玉琦 2018102160002 {Email: Yuqizhao@whu.edu.cn}

## 1.研究背景

Web 服务作为一种新型的分布式计算模型和协作关系，为动态、跨组织的资源协同共享创造了前所未有的机会，得到了工业界和学术界的极大关注。随着 Web 服务标准和技术逐步成熟，开发和部署 Web 服务已经变得更加容易。Web 服务技术的最终目标是在面向服务的体系架构（Service Oriented Architecture, SOA）的基础上根据用户提出的需求将分散、独立的 Web 服务进行无缝自动的组合，以满足用户日益增长的需求<sup>[1,2]</sup>，同时能够根据用户的需求和服务的 QoS 给出预测和推荐。

## 2.介绍

随着面向服务的计算（SOC）的广泛发展，越来越多的软件应用程序通过重用现有的 Web 服务来开发，这些 Web 服务可以被发现和组合，以交付需要的功能，从而加速 Web 服务或 Web API 的创建和部署。越来越多的 Web 服务为开发人员在构建基于服务的系统中选择合适的服务带来了沉重的负担。服务选择的最终目标是找到能够同时满足用户功能和非功能需求的服务。随着功能相似服务的不断增加，服务的非功能属性因此成为决策服务选择的主要考虑因素。

服务质量 QoS 指的是 Web 服务的非功能特性，如响应时间、吞吐量等。它作为确定可以选择或推荐哪些服务的基础。一般来说，QoS 值可以在提供方端和用户端上测量。由于服务用户的动态网络环境和不同的地理位置，不同用户在调用相同的服务时可能会观察到不同的 QoS 值，用户端 QoS 评估应该得到更多的关注，它与 web 服务上的个性化用户体验更加相关。获取用户端 QoS 值的直接方法是让用户进行实际调用，这在实践中是不可行的，因为过程耗时过多开销过

大。因此，如何准确地预测不同用户的服务的未知 QoS 值成为服务计算中的一个迫切问题。

### 3.相关工作

Web 的 QoS 度量一直是面向服务的软件工程中的一个重要研究点，下面回顾下近年来一些具有代表性的研究成果。

Limam N 和 Boutaba R<sup>[3]</sup>提出了一种基于信誉度感知的服务评价和选择框架。框架中的服务评价采用了市场营销学（Market Science）中期望失验理论（Expectancy-disconfirmation Theory）用于研究用户满意度与服务质量之间的关系来得到用户的反馈（该反馈来自服务质量感知函数（Perception Function）和失验函数（Disconfirmation Function）的综合，而不是用户的直接反馈）。为了克服用户对所用服务的质量期望（Expectation）和感知（Perception）掺杂的主观感觉以使其达到客观、公平和可度量，作者使用服务水平协商（SLA）来量化用户的服务质量期望，采用质量监测量化用户服务质量感知。

Malik Z 和 Bouguettaya A<sup>[4]</sup>针对新服务用户的可参考或有记录的数据信息较少而难以准确地计算出其信誉度的情况，提出了一种对 Web 服务新用户信誉度的度量方法，即信誉度自举法（Reputation Bootstrapping）。该方法通过建立两个模型来度量用户的信誉度，即对特定领域内（注册在同一个社区）的 Web 服务可以通过用户之间的互相评估来度量出一个新用户的初始声誉。

Conner W 等人<sup>[5]</sup>提出了一个面向开放分布式服务环境的基于信誉度的服务可信管理框架，可信是可信管理服务（Trust Management Service, TMS）。TMS 不仅支持多种不同实体的信任关系，同时还支持每个实体采用不同的洗浴度评分函数（Scoring Function）对同样的反馈数据进行评估。该框架的一个显著优点是支持多种信誉度评估方法。但也存在缺点，虽然 TMS 将包含客户端、服务、反馈至（等级）最优属性集中元素项在内的服务调用历史记录用于信誉度评估比原有的直接计算信誉度更加准确，但是该记录难以对用户反馈中包含的主观感觉、用户偏好等对用户反馈影响较大的因素信息进行识别、记录或处理，导致信誉度评估中充斥大量的主观因素。

邵凌霜等人<sup>[6]</sup>基于协同过滤的个性化 QoS 预测方法的不足,即在相似度计算中使用 Pearson 相关系数一般用于度量数据集之间的线性相关关系,并不能很好的解释客观 QoS 数据之间的相似性。搞研究改进了相关的相似度计算算法,并给出次相似度技术方法提出了新的预测方法,进一步提高 QoS 预测的准确性,效果较好,但对于用户上下文在 QoS 预测中的积极因素考虑较少。

作为动态环境下 Web 服务推荐和选择的关键技术, QoS 预测在过去十年中得到了深入的研究。针对这一问题的现有研究可以分为两类,即是否可以处理时间序列的 QoS 数据。本领域的第一部分研究主要集中在静态时间间隔下的个性化 QoS 预测,不利用过去的 QoS 预测。协同过滤方法是该场景中应用最为广泛的技术,可以大致分为两类:基于邻居的和基于模型的。基于邻居的方法根据用户过去的 QoS 体验度量用户或服务的相似性,然后利用相似邻居的 QoS 信息,对候选 Web 服务上的用户进行个性化的 QoS 预测。基于邻居的协同过滤方法可以进一步分为基于用户的、基于服务的和混合的方法<sup>[7]</sup>。基于模型的方法采用机器学习技术来学习基于训练数据的未来预测模型。基质因素化是这种类型的代表,例如 CloudPred<sup>[8]</sup>和 NMF<sup>[9]</sup>。已有文献表明,基于 MF 的方法在 QoS 预测性能上优于基于邻居的方法,然而,这些方法不能捕获 QoS 值的时间动态,这使得它们很难获得理想的结果。

另一组研究,时间感知的 QoS 预测方法,训练模型以适应过去的 QoS 值,然后预测它们的未来趋势。例如, Godse 等人<sup>[10]</sup>提出了基于 ARIMA 模型来预测服务性能。Amin<sup>[11]</sup>等提出了一种 QoS 预测方法,将 ARIMA 与 GARCH 模型集成在一起,以解决 ARIMA 恒定变化假设的局限性。然而,基于 ARIMA 的模型主要关注于预测单个用户观察到的每个 Web 服务的 QoS 值,这使得它们难以预测多个用户在服务选择和个性化推荐过程中观察到的多个服务。因此引入协同过滤方法(CF)来缓解这个问题。例如 Hu 等人<sup>[12]</sup>提出的模型是将 CF 与基于 ARIMA 和 Kalman 滤波的改进时间序列预测相结合的模型。使用用户服务时间张量对时间感知 QoS 预测问题进行形式化,并进行张量分解。然而,当维数变大时,张量分解就变得难以处理和低效。Silic 等人提出将历史 QoS 数据进行聚类,并对每个簇的平均信度进行哈希用于预测。Zhu 等人采用了类似的策略,提出了一种新的具体情境的矩阵分解方法。

## 4.QoS 相关数据集及方法--WSDREAM

### 4.1 数据集介绍

在服务计算 QoS 相关的研究中有众多优秀的数据集，其中以香港中文大学发布的 WS-DREAM(Distributed REliability Assessment Mechanism)数据集最为流行。WS-DREAM 数据集主要包含三个组件：Web 服务 QoS 预测组件、Web 服务日志管理组件和 Web 服务审查挖掘组件组成。Web 服务 QoS 预测组件，允许用户以协作方式执行 Web 服务的可靠性和质量评估；Web 服务日志管理组件，它通过日志提供服务系统可靠性增强的端到端框架；Web 服务审查挖掘组件，它通过自然语言处理技术自动分析用户评论。目前关于该数据的出版物总共被引用了 2000 多次，并应用于 200 多篇高水平论文。WS-DREAM 存储库维护着 3 组数据：(1)QoS（服务质量）数据集；（2）日志数据集；（3）审查数据集。

### 4.2 基础方法

基于 WS-DREAM 数据集开展了大量 QoS 预测和服务推荐的相关的工作，其中基础的方法 UMEAN 基于用户平均的方法和 IMEAN 基于物品（服务）平均的方法。

### 4.3 近邻方法

如图 1 所示，利用用户相似性计算近邻的思路，提出了将两种基础方法结合在一起的 UIPCC<sup>[7]</sup>方法，该方法通过相似度来分别计算相似用户和相似的服务，并通过相似邻居的值来预测缺失值，具有很好的效果。

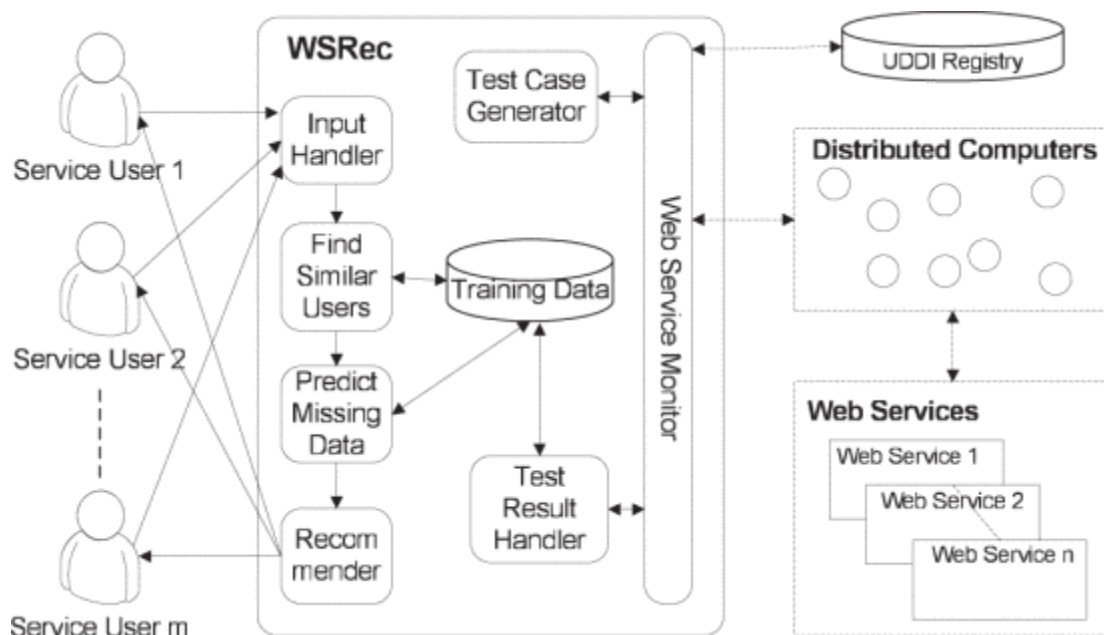


图 1 UIPCC 方法示意图

相似的方法还有 ADF<sup>[13]</sup>和 NRCF<sup>[14]</sup>方法等，他们均采用了通过历史信息计算相似的用户和相似的服务，然后对服务的 QoS 值进行预测的方法，介绍基于服务质量(QoS)的服务选择的概述。组合服务由多个抽象服务组成，每个抽象可以通过一组功能等效的具体服务来实现。基于 QoS 的服务选择的目标是从每一组具体服务中选择候选服务，以满足端到端的 QoS 需求，并优化组合 QoS。

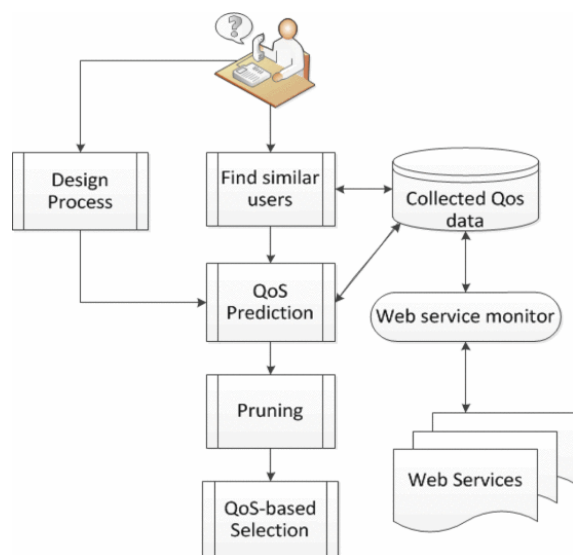


图 2 ADF 方法示意图

图 2 给出了该方法的体系结构，其中给出了一组功能等效的 Web 服务，Web 服务的 QoS 是请求者关心的问题。当用户的需求到达时，首先使用查找相似用

户的过程，而只有通过组合一组服务才能完成这一需求时，也应该使用设计过程的过程。在寻找相似用户的过程中，我们利用收集到的 QoS 数据来计算请求者与其他用户之间的相似性，从而找到与请求者相似的用户。在设计过程中，设计了一个抽象服务的组合过程来实现用户的需求。在找到相似用户后，利用 QoS 预测的过程，获得所有候选 Web 服务的请求者的未知 QoS 值。然后，使用修剪过程对质量较低的 Web 服务进行修剪，以减少可供选择的服务数量，而使用基于 qos 的选择过程通过实用函数选择最佳的候选服务并向请求者推荐它们。

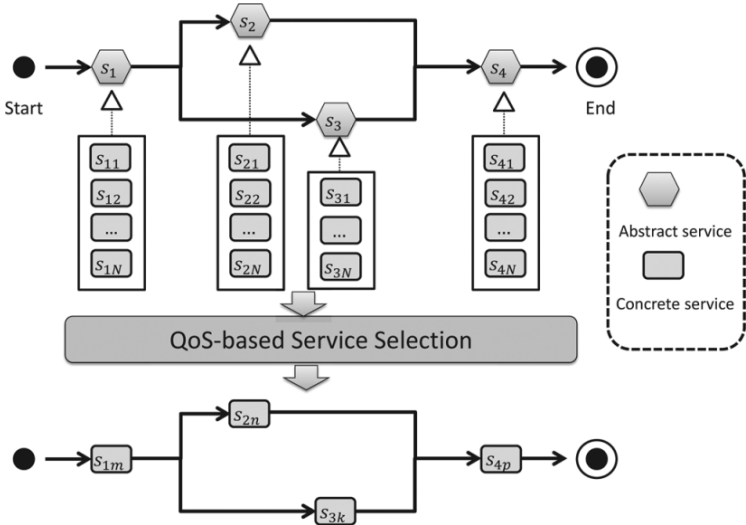


图 3 基于 QoS 的服务选择流程

图 3 介绍基于服务质量(QoS)的服务选择的概述。组合服务由多个抽象服务组成，每个抽象可以通过一组功能等效的具体服务来实现。基于 QoS 的服务选择的目的是从每一组具体服务中选择候选服务，以满足端到端的 QoS 需求，并优化组合 QoS。

#### 4.4 基于模型的方法

根据已有的 QoS 历史信息，可以对服务进行建模，通过模型的方法来进行服务的预测和推荐，为服务的选择提供帮助。例如基于矩阵分解的 BasicMF、PMF、NMF 和 LN-LFM 方法，通过矩阵分解模型<sup>[15,16]</sup>。

### 4.4.1 BasicMF（矩阵分解）

PMF (Probabilistic Matrix Factorization)即概率矩阵分解方法，RegularizedMF 是对 BasicMF 的优化，而 PMF 是在 RegularizedMF 的基础上，引入概率模型进一步优化。假设用户  $U$  和项目  $V$  的特征矩阵均服从高斯分布，通过评分矩阵已知值用 MAP（最大后验概率）和 MLE（最大似然估计）得到  $U$  和  $V$  的特征矩阵，然后用特征矩阵去预测评分矩阵中的未知值。

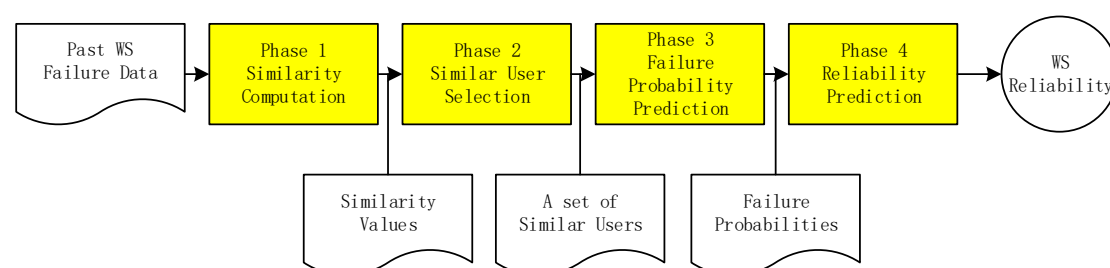


图 4 根据历史数据的矩阵分解框架示意图

失效概率预测过程通常包括一个用户项矩阵，其中用户对应于行，项目(Web 服务)对应于列。矩阵条目指定每个 Web 服务上每个用户的失败概率值。我们研究的问题是如何利用可用的失效概率值准确地预测用户项矩阵中缺失的失效概率值。

矩阵分解是指一组算法，其中一个矩阵分解成两个矩阵的乘积。在应用矩阵分解来解决我们的研究问题时，其前提是影响用户观察到的 Web 服务失败概率的因素很少，而用户在 Web 服务上观察到的失败概率取决于每个因素如何应用于用户。更具体地说，Web 服务上的用户失败概率值对应于这些因素与用户特定系数的线性组合。

### 4.4.2 LN-LFM

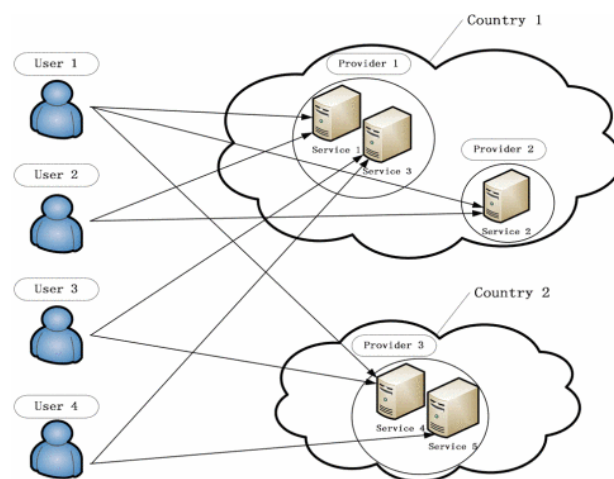
LN-LFM 是基于隐变量的推荐 (Latent Factor Model) 阐述了如何利用潜在因子模型 (LFM) 预测未知的 QoS 值。同时，我们考虑了提供者及其国家的因素，这意味着潜在的物理位置和网络状态信息，作为 web 服务集的潜在邻居。因此，建立了新的邻域因子模型，以评估每个服务用户的潜在邻居的个性化连接质量。

然后，提出了一种基于 LFM 的集成模型。最后，我们对一个大规模的真实世界 QoS 数据集进行了一组实验，结果表明该方法是有用的，特别是在数据稀疏的情况下。

假设已经有  $n$  个人对  $m$  个商品进行评价，其评价矩阵  $A_{m \times n}$  反应了每个人对每个商品的喜好程度（行为商品，列为用户）。一般来说，一个人对商品或者某种事物的直观感受会受到该商品或事物属性值的影响，比如一个用户最近打算找一个对象，在众多的服务中他选中了服务  $D$ ，而他本人喜欢服务的类型具有高可用，低延时等，这些特点就是他对服务特征的一种偏好，而每个服务在可用性、响应时间、等特征上都有一定值，对于一个用户来说，他选择的  $D$  服务一定是一个在他的需求的属性中最符合的。虽然我们从矩阵  $A$  中只看到了每个用户对每个商品的评价，但是我们可以认为这个评价值是用户对商品的属性值的偏爱程度与每个商品在这些属性上的表现综合结果，这里的属性就是我们所说的隐变量。基于这样的思想，我们可以将矩阵  $A$  分解成为“商品—商品属性”的评价矩阵  $U_{m \times k}$  与“商品属性—客户喜好”矩阵  $V_{k \times n}$  的乘积，及：

$$A_{m \times n} = U_{m \times k} V_{k \times n}$$





(a) A Simple Real-world Scenario

	S1	S2	S3	S4	S5
U1	5.0	3.2		1.2	
U2	4.0	2.4			
U3			7.0	2.8	
U4			5.0		3.6

(b) User-Service Matrix

	S1	S2	S3	S4	S5
U1	5.0	3.2	<b>6.7</b>	1.2	<b>4.2</b>
U2	4.0	2.4	<b>4.5</b>	<b>0.8</b>	<b>4.8</b>
U3	<b>5.9</b>	<b>3.4</b>	7.0	2.8	<b>4.0</b>
U4	<b>4.8</b>	<b>2.2</b>	5.0	<b>0.9</b>	3.6

(c) Predicted User-Service Matrix

图 5 LN-LFM 方法示意图

该方法试图解释如何在潜在因子空间中建立 QoS 预测问题的模型。基于直觉的一组 Web 服务发布的相同的供应商通常有相同的条件基本的网络环境,我们扩展基线评估模型,提出一个邻居模型,评估潜在的邻居为每个用户的个性化的连接质量和利用相同的潜在特征向量来收集当地信息。

## 5.利用 QoS 来进行服务选择

### 5.1 摘要介绍

基于可信和异构的 QoS 属性来进行有效的空间大数据服务选择<sup>[17]</sup>, 随着云服务的数量以惊人的速度增长, 提供类似功能的服务供应商的数量也在不断增加。

选择具有用户期望的非功能性属性（NFP）的服务变得非常重要，但是引发了许多与大数据相关的研究问题。首先，选择决策应该处理大量的服务 NFP 数据。第二，服务选择需要反映不同的用户偏好，包括定性和定量的。第三，网络和服务负载的不确定性导致 NFP 的高可变性。第四，由于服务 NFP 的信任值是通过历史用户的反馈收集的，因此它给服务的 NFP 带来了准确性维度。第五，应该平衡多个有时相互冲突的最优服务选择决策目标。一个有效的服务选择机制是需要的，它能够以集成的方式处理上述所有大数据挑战，以处理具有高度可变性的高度多样化 QoS 以及导致数据准确性的信任相关问题。现有的研究关注于用户的 QoS 偏好或者他们的信任问题，但是未能提供一个系统解决方案来将两个标准集成到选择过程中。

## 5.2 相关工作

由于服务空间庞大，用户需求多样，服务选择是服务计算的一个开放性問題。提出了一些服务选择体系结构，以保证基于服务的应用程序设计和服務組合的有效服務選擇。Ran<sup>[18]</sup>首先通过提出服务发现模型扩展了传统的 SOA 体系结构以支持 QoS。哈达德等<sup>[19]</sup>提出了一种寻找最佳 Web 服务的 QoS-Broker 体系结构，它处理 Web 服务客户端的查询以及寻找最佳 Web 服务的 QoS 要求。Benouaret 等人<sup>[20]</sup>提出了用于服务选择的 WS-Sky 框架。该框架利用 skyline 概念的两个变体来有效和高效地选择能够更好地满足用户需求的 Web 服务。

## 5.3 方法介绍

在本文中，我们通过发展一种新的多目标优化方法来处理基于异构偏好和信任的服务选择，在服务的信任值和用户的 QoS 偏好之间进行权衡决策，以根据候选云服务与用户需求的匹配程度对候选云服务进行排序。我们进行了大量的实验来评估所提出方法的有效性和效率。

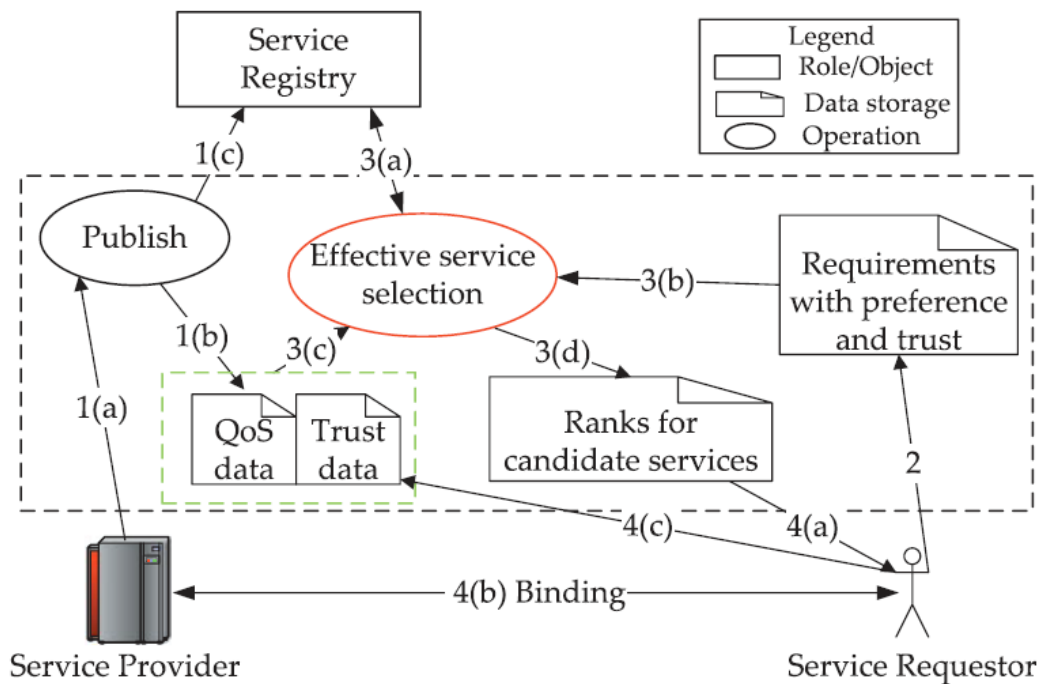


图 6 云服务选择的流程框架

问首先基于传统的面向服务的体系结构（或 SOA）标准提出了一个用于云服务选择的标准流程。如图 6 所示，该过程包括以下四个步骤：

1)当服务提供商发布他们的云服务时，我们将在 QoS 数据中收集和存储 QoS 属性值，然后将功能细节发布到服务注册中心（参见图 6，第 3 行(a)-(c)）。

2)当服务请求者需要云服务时， he 可以用他的定性/定量偏好以及信任等价物来指定云服务（图 6，第 2 行）。

3)在进行选择时，首先根据功能属性在服务注册表中找到候选云服务(图 6，第 3 行(a))。其次，我们对每个候选服务进行排序，然后根据包含 QoS 属性值和信任信息的服务 NFP 选择最佳服务给服务请求者（图 6，第 3 行（b）-（d））。

4)在服务请求者完成一个调用过程之后，我们将把服务 QoS 的用户反馈存储在信任数据中（图 6，第 4 行（c））。通过这种方法，虚线矩形中的部分充当客户端和提供程序之间的中介。

提出了集成服务选择方法来解决大数据挑战。为了处理海量且种类繁多的 NFP 数据，我们扩展了处理异构信息的方法。结果，可以使用统一的度量来计算和比较每个 QoS 属性的用户需求的匹配程度。对于可变性和准确性的挑战，通过综合考虑 QoS 偏好和服务信任来计算匹配度。为了处理多个决策目标之间的

冲突，采用多目标优化模型进行服务选择决策。

总之，所提出的云服务选择方法包括以下三个阶段。首先，我们通过无量纲的加性效用函数（即在 $[0,1]$ 的范围内）来处理不一致的定量 QoS 特性。其次，通过将信任与用户约束相结合，计算候选服务的每个定量/定性 QoS 属性的 QoS 匹配度。特别地，定性 QoS 属性不是数值。当表示用户的偏好时，它依赖于其他定性 QoS 属性。我们不能简单地标准化定性 QoS 属性。在计算它们的 QoS 匹配度时，将特别处理这些属性。第三，通过多目标约束模型定义一个线性函数来计算每个候选服务的排序值。

## 5.4 实验设计

为了探究新提出的方法在基于大数据云服务的应用中的实用性，文章针对提出的方法和数据进行了两个实验，（1）在用户满意度方面特别是在与其他竞争方法的对比上，新提出的方法的有效性如何？（2）这种集成的服务选择方法的计算复杂度如何？

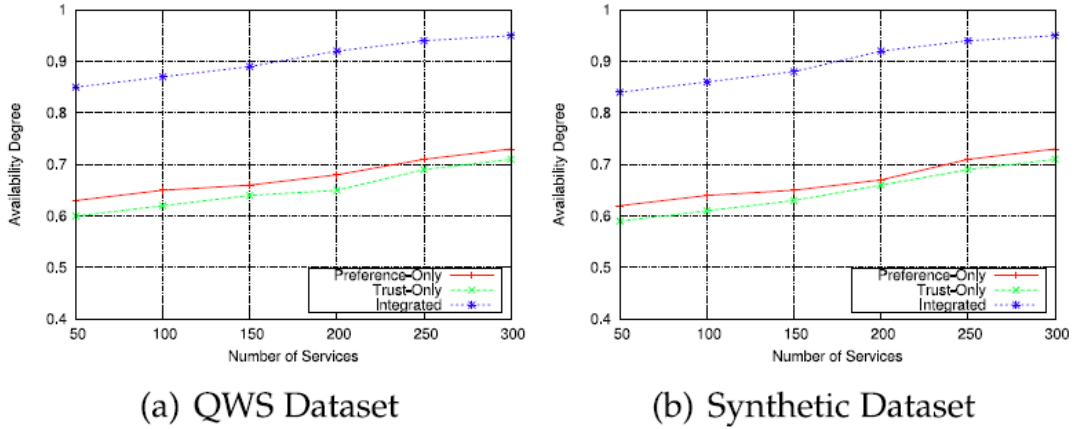


图 7 集成方法与独立方法在 AAD 上的表现

在选择相同数量的服务时，集成服务选择方法的 AAD 值高于仅偏好和仅信任。这表明集成方法在用户对 QoS 的偏好和服务的信任值之间进行了有价值的权衡。同时，仅偏好和只信任的 AAD 值非常接近。该观察表明，当综合考虑用户偏好和服务信任时，仅偏好和可信度表现出类似的性能。

总之，在相同数据集上进行的不同方法的有效性实验结果显示出较大的差异。为了满足用户偏好，并选择具有良好 QoS 的服务，集成服务选择方法显示出比

其他两种方法更好的性能。另一方面，对不同数据集进行相同方法的差异较小。这说明合成数据集较好地逼近了真实数据，基于合成数据集的实验信息量大。需要注意的是，我们还可以为不同的应用程序使用不同的方法（例如，只偏好或只信任）。我们应该分析具体的应用需求来选择最合适的方法，并且应该分别考虑每种方法的优点和缺点。

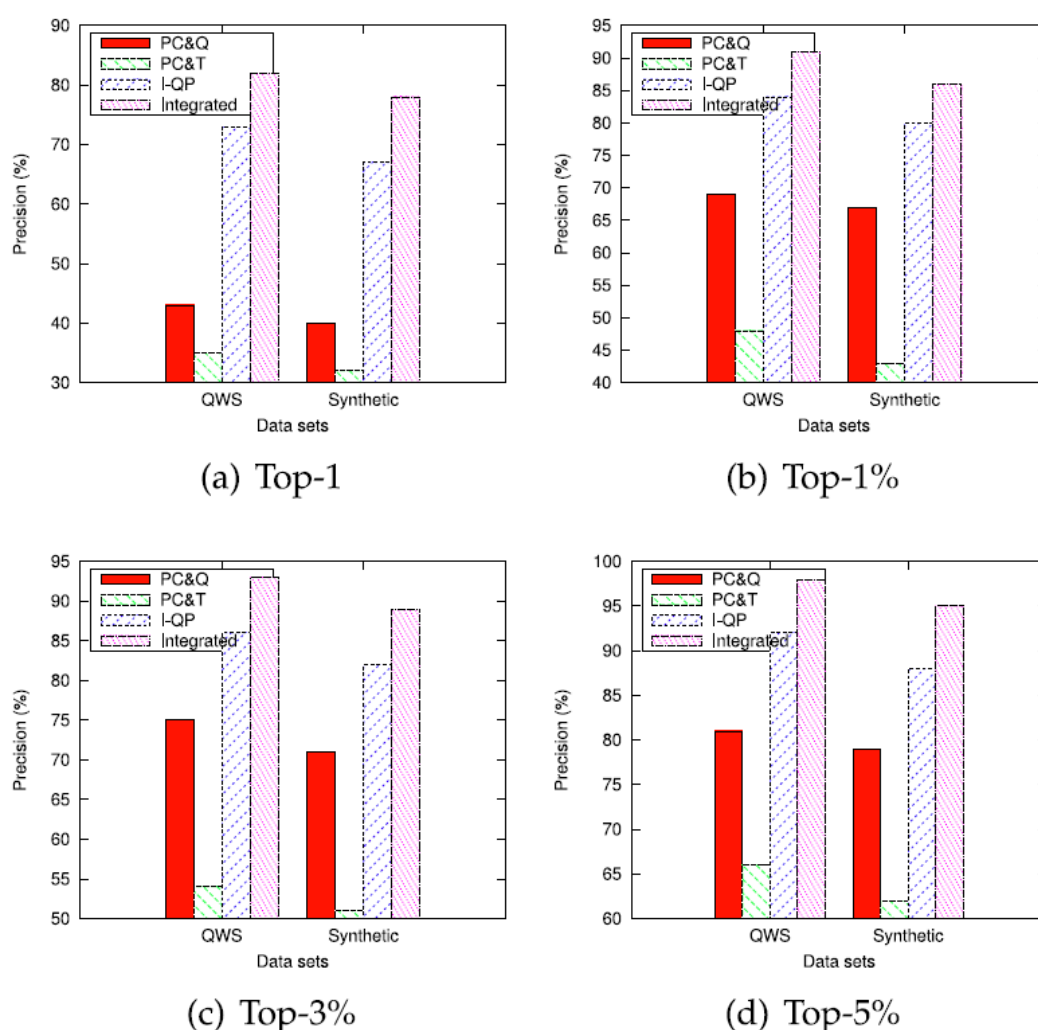


图 8 在精度上与现有方法的对比

从图8中可以看出,(1)综合方法的精度略高于I-QP,远高于PC&Q和PC&T。这是因为当对服务进行评估决策时,PC&Q和PC&T虽然考虑了用户的主观偏好(即相对重要性)和服务的客观性能(即Q值)。将QoS和历史用户对性能的评价作为信任值)综合考虑,通过多目标约束模型综合考虑上述多个因素,还可以对用户主观判断和多个决策目标引起的冲突进行权衡。此外,集成方法还可以支持用户的定性偏好;(2)随着服务选择结果集中所选服务数量的增加,精度逐渐提高。

特别是当结果集中包含 Top-5%时，集成方法的精度接近 100%。这个观察意味着，我们还可以向需要的用户推荐 Top-5%的服务，并且最终的选择决策应该由用户手动做出。这为大数据时代的云计算服务选择应用提供了一个有前途的解决方案。

## 5.5 小结

在本文中，我们通过发展一种新的多目标优化方法来处理基于异构偏好和信任的服务选择，在服务的信任值和用户的 QoS 偏好之间进行权衡决策，以根据候选云服务与用户需求的匹配程度对候选云服务进行排序。我们进行了大量的实验来评估所提出方法的有效性和效率。

## 6.基于 QoS 的服务预测和推荐

基于 LSTM 的个性化矩阵分解在线 QoS 预测<sup>[21]</sup>。服务质量预测是服务计算中的一个重要课题，在过去十年中得到了广泛的研究，提出了许多时间感知的 QoS 预测方法，取得了令人激动的预测效果。但是，它们没有提供有效的模型更新机制，因此必须定期对整个模型进行重新训练，以处理即将到来的数据。如何及时更新预测模型，准确预测候选服务的 QoS 缺失值，成为亟待解决的问题，本文提出了一种基于矩阵分解的基于个性化 LSTM 的在线 QoS 预测方法。我们的方法可以捕获多个用户和服务的动态潜在表示，并且可以及时更新预测模型来处理新数据，在实际数据集上进行的实验表明，我们的方法在在线预测性能方面优于几种最先进的方法。

一般来说，由于动态网络环境和服务状态，QoS 值随时间的变化很大。基于目标用户服务请求时间来预测 QoS 值的时间感知 QoS 预测得到了越来越多的关注。针对这一问题已经提出了许多方法，取得了令人鼓舞的预测效果。在真实的 QoS 预测场景中，随着时间的推移，新的 QoS 观测不断地出现。是否需要更新经过训练的预测模型，以及如何更新，在现有文献中还没有得到充分的讨论。为此，我们首先进行了实证实验，分析了现有模型更新时间感知方法的性能。实验

中使用的数据集是 WSDream 的一种变体，它是最流行的 QoS 预测数据集，它由 140 个用户在 64 个时间段内观察到 4500 个服务的 QoS 数据组成。更多关于数据集、实验设置和评估度量的描述在第五部分中可以找到。在这里，我们只是说明了对两种代表性的时间感知 QoS 预测方法的一些观察：CLUS 和 WSPred。

根据实验可得，预测模型应该通过增加新出现的数据进行增量更新，以获得更好的预测性能。然而现有的时间感知方法很少提供有效的模型更新机制，他们必须使用大量的历史数据对模型进行训练，这使得他们很难满足在线预测的要求。因此，如何及时更新预测模型，提供在线 QoS 预测成为了一个重要问题。为了解决这一关键挑战，本文提出了一种基于矩阵分解的新型个性化 LSTM 算法，用于在线 QoS 预测。我们的模型可以用来描述多个用户和服务的动态潜在表示，并且可以根据当前的观察和有限的历史数据以及一些长期保留的信息进行更新。本文的主要贡献总结如下：我们研究 Web 服务的在线 QoS 预测问题，这在现有文献中还没有充分讨论；我们提出了一种新型的基于矩阵分解的个性化 LSTM 算法，用于在线 QoS 预测。

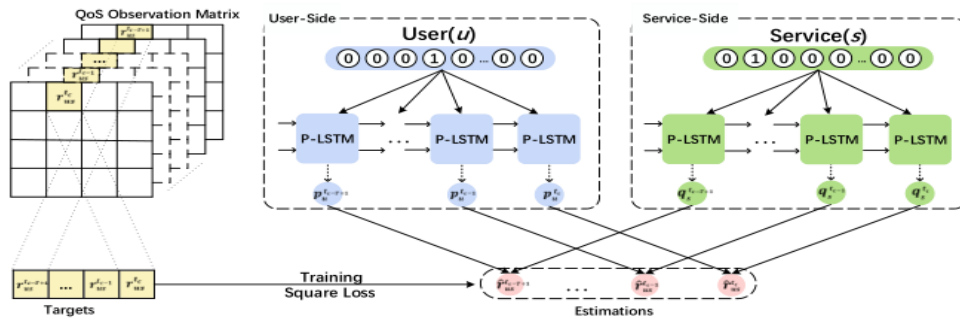


图 9 基于 LSTM 的个性化矩阵分解在线 QoS 预测

本文提出了一种基于矩阵分解的基于个性化 LSTM 的在线 QoS 预测方法，我们的方法可以捕获多个用户和服务的动态潜在表示，并且预测模型可以及时更新以处理新出现的数据。在未来，我们计划将我们的方法应用到包括更多 QoS 属性在内的其他真实数据集，我们还考虑将用户和服务的位置和 IP 地址等更多的方面信息集成到我们的模型中。



## 7.QoS 感知推荐的原子 Web 服务可靠性预测

在基于面向服务的系统构建 QoS 感知复合工作流时，有必要评估潜在服务选择候选项的非功能特性。文章提出了一种用于原子 Web 服务可靠性预测的 CLUS 模型，该模型根据以前调用中收集的数据估计正在进行的服务调用的可靠性。为了提高当前最先进的预测模型的准确性，我们结合了调用上下文的用户服务和特定于环境的参数。为了减少最先进方法中存在的可伸缩性问题，我们使用 k-means 聚类算法聚合过去的调用数据。为了评估模型的不同质量方面，我们对部署在亚马逊云不同区域的服务进行了实验。评估结果证实，与目前最先进的方法相比，我们的模型产生了更可扩展和更准确的预测。

文章[22]提出一种新型的 Web 服务可靠性预测模型 CLUS。模型基于以前收集的数据估计当前服务调用的可靠性，模型参数提供对服务调用上下文更准确的描述：user-specific, service-specific, environment-specific。提出一种新的服务使用反馈装配策略，该策略能够发现偏离假定负载分布的偏差，并用于提高 CLUS 的准确性。提出一种基于模型的协同过滤方法，该方法利用了一种无监督机器学习技术——线性回归。

CLUS 模型可以通过模型参数的调整来提供更准确的上下文描述。(1) User-specific 特定于用户的参数，将该参数与用户因素引起的服务可靠性性能波动联系起来。包括影响可靠性的各种因素，比如用户的位置、网络和设备功能。使用 k-means 聚类方法，根据从历史调用记录中获得的可靠性性能将用户聚合到各个簇中。(2) service-specific 特定于服务的参数，该参数与服务特性对服务可靠性性能的影响有关，包括影响可靠性的各种因素，比如服务的位置、计算复杂度、系统资源，使用 k-means 聚类方法，根据从历史调用记录中获得的可靠性性能将服务聚合到各个簇中。(3) environment-specific 特定于环境的参数。该参数与环境中的当前条件相关，例如调用时的服务提供者负载或网络性能。为了进行评估，这里只考虑服务负载作为环境参数，将服务负载定义为每秒接收的请求数量。服务的非功能特性，如可用性和可靠性，会受到服务负载波动的显著影响。由于 web 服务器在一天中会记录大量的负载变化，因此将一天划分为任意数量的时间窗口。



将历史调用数据分散到不同的时间窗口中。使用 k-means 聚类，根据过去的调用记录计算的服务可靠性能，将时间窗口聚合到各个簇中。

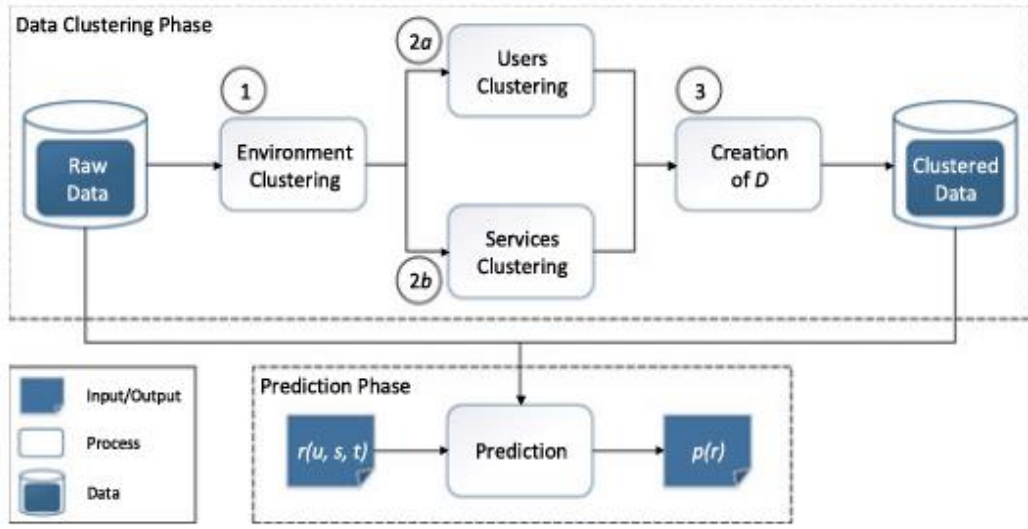


图 10 QoS 感知推荐的原子 Web 服务可靠性预测

如图所示，整个预测过程可以分为数据聚类阶段和预测阶段。数据聚类阶段主要的工作是根据过去调用示例的可靠性能，对时间窗口聚类，然后对每个时间窗口簇，根据过去调用示例的可靠性能，对用户和服务聚类。

首先对特定于环境的参数聚类环境条件定义为： $E=\{e_1, e_2, \dots, e_n\}$  其中， $e_i$ 是特定环境条件下的服务负载， $n$  是任意数目的不同环境条件。目的是将每个可用的历史调用记录与服务提供者执行时的负载关联起来，对从不同服务提供商收集的数据进行分析，可以发现在一定时间内负荷分布的规律。

首先对特定于环境的参数聚类，将一天划分成任意数目的时间窗口，时间窗口 $w_i$ 用开始时间 $t_i$  结束时间 $t_{(i+1)}$ 描述，假定环境参数在一个时间窗口内是稳定的，计算每个时间窗口的平均可靠性：

$$\overline{p_{w_i}} = \frac{1}{|W_i|} \sum_{r \in W_i} p_r, \quad (\text{公示 1})$$

其中， $W_i$ 是时间窗口中的历史调用记录的集合， $r$ 是过去调用的示例， $p_r$ 是可靠性能。首先对特定于环境的参数聚类利用 K-means 算法，根据每个时间窗口的平均可靠性聚类到 $e_i$ 中。目标函数：时间窗口到质心的距离平方和，其中， $\mu_{e_k}$ 是 $e_k$ 的质心， $\{b_{w_i, e_k}\}$ 是一个二元指示量，当 $w_i$ 属于 $e_k$ ，取 1，否则取 0。

k-means 算法的每次迭代分两个步骤：第一步，保持  $\{\mu_{e_k}\}$  不变，最小化关于  $\{b_{w_i, e_k}\}$  的目标函数，第二步，保持  $\{b_{w_i, e_k}\}$  不变，最小化关于  $\{\mu_{e_k}\}$  目标函数。利用 K-means 算法，根据每个时间窗口的平均可靠性聚类到  $e_i$  中。K-means 算法收敛时，我们通过下述的传递关系，将历史调用示例中的每个特定记录与环境条件关联起来。传递关系：如果调用记录  $r(u, s, t)$  属于时间窗口  $w_i$ ，而  $w_i$  属于环境条件  $e_i$ ，则  $r(u, s, t)$  和  $e_i$  相关联。

对特定于用户的参数聚类，定义用户组：  $U=\{u_1, u_2, \dots, u_n\}$ ，其中，  $u_i$  中的用户具有相似的调用服务可靠性，对每个用户，根据过去调用示例，计算  $n$  维可靠性向量，每一维表示在特定环境条件  $e_i$  下用户  $u$  的平均可靠性性能：利用 K-means 算法聚类，根据  $p_u$  进行用户聚类。K-means 收敛时，通过传递关系将历史调用示例中的每个特定记录和服务关联起来。对特定于服务的参数聚类，定义不同的服务构成服务组： $S=\{s_1, s_2, \dots, s_n\}$ ，对每个服务，根据过去调用示例，计算  $n$  维可靠性向量，每一维表示在特定环境条件  $e_i$  下服务  $s$  的平均可靠性性能：利用 k-means 算法聚类，根据  $p_s$  进行服务聚类。K-means 收敛时，通过传递关系将历史调用示例中的每个特定记录和服务关联起来。

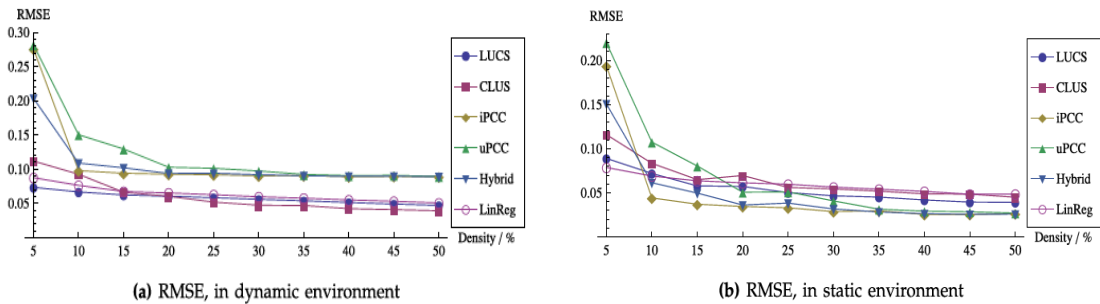


图 11 LinReg 方法与其他现有方法在不同环境下的性能对比

实验及结果数据密度的影响 (1) 通过改变实验中收集的数据的 5%到 50%之间的数据密度来模拟不同数量的收集数据，步长为 5%。(2) 我们通过改变特定于环境的参数来创建两个不同的环境。在第一种情况下,我们假设一个动态的环境中的不同的负载是请求频率从 3 次/秒到 9 次/秒。在第二种情况下,我们假设一个静态环境的恒定负载请求频率 3 次/秒。为了评估密度的影响，我们将数据分为 7 个组。(3) 首先，随机包含 5%的训练数据，然后利用各个方法预测剩余数据的可靠性，利用实验中测量到的可靠性值计算 RMSE 值，测量各个方法的训

练和预测时间，按步长为 5% 随机增加训练数据并重新计算预测和性能度量，直到密度达到 50%。

本文提出了两种新型的方法 CLUS LinReg，于提高预测精度，获得更高的可伸缩性。提高预测精度，现有的方法隐式地只考虑预测的特定于用户和服务的参数。文中提出的方法在关注特定于用户和服务的参数之外还引入了特定于环境的参数，从而显著降低了 RMSE。在数据密度较高时，CLUS 方法的性能最优(例如，它的 RMSE 比 LUCS 低 17%)，在数据密度较低的时候，LUCS 优于 CLUS 和 LinReg，但是 LinReg 的 RMSE 比 CLUS 低 21%。

## 8.总结

文章通过对面向服务的软件工程中 QoS 相关问题的探究，从 QoS 的基本概念到 QoS 在面向服务的软件工程中扮演的重要作用有了全面的了解和深刻的认识。正因如此，近年来工业界和学术界对于 QoS 的研究热度不断，从 QoS 的缺省值分析与预测，到基于 QoS 非功能属性的服务选择，到根据不同 QoS 特性进行的服务组合，再到基于 QoS 历史数据并结合当下流行的机器学习、深度学习等算法的深度融合而做的服务推荐等工作，同时我们发现，通过对于 QoS 的分析和建模等工作，可以对服务的可靠性进行很好的评估，也有利于服务的不断改进和完善。

## 9.致谢

感谢学院能够在博士培养计划中开设如此重要的课程，尤其彭老师精心的备课和精彩的授课，通过面向服务的软件工程这门课程的学习，我对服务计算产生了更加浓厚的兴趣，对整个领域有了更加全面的认识，更加牢固的掌握了各项知识点。然而面向服务的软件工程内容丰富，涉及众多，未能在课程期间掌握的知识，还需要在以后的研究实践中不断探索。

## 引用

- [1] Funk C , Mitic J , Kuhmunch C , et al. Adaptation of Composite Services in Pervasive Computing Environments[C]// IEEE International Conference on Pervasive Services. IEEE, 2007.
- [2] Ardagna D, Pernici B. Adaptive service composition in flexible processes. IEEE Transactions on Software Engineering, 2007, 33(6):369-384.
- [3] Limam N , Boutaba R . Assessing Software Service Quality and Trustworthiness at Selection Time[J]. IEEE Transactions on Software Engineering, 2010, 36(4):559-574.
- [4] Malik Z, Bouguettaya A. Reputation Bootstrapping for Trust Establishment among Web Services[J]. IEEE Internet Computing, 2009, 13(1):40-47.
- [5] Conner W, Iyengar A, Mikalsen T, et al. A trust management framework for service-oriented environments[C]// International Conference on World Wide Web. 2009.
- [6] 邵凌霜, 周立, 赵俊峰, et al. 一种 Web Service 的服务质量预测方法[J]. 软件学报, 2009, 20(8):2062-2073.
- [7] Z. Zheng, H. Ma, M. R. Lyu, and I. King, “Qos-aware web service recommendation by collaborative filtering,” IEEE Transactions on services computing, vol. 4, no. 2, pp. 140–152, 2011.
- [8] Y. Zhang, Z. Zheng, and M. R. Lyu, “Exploring latent features for memory-based qos prediction in cloud computing,” in Reliable Distributed Systems (SRDS), 2011 30th IEEE Symposium on. IEEE, 2011, pp. 1–10.
- [9] D. D. Lee and H. S. Seung, “Algorithms for non-negative matrix factorization,” in Advances in neural information processing systems, 2001, pp. 556–562.
- [10] M. Godse, U. Bellur, and R. Sonar, “Automating qos based service selection,” in Web Services (ICWS), 2010 IEEE International Conference on. IEEE, 2010, pp. 534–541.
- [11] A. Amin, A. Colman, and L. Grunske, “An approach to forecasting qos attributes of web services based on arima and garch models,” in Web Services (ICWS), 2012 IEEE 19th International Conference on. IEEE, 2012, pp. 74–81.
- [12] Y. Hu, Q. Peng, X. Hu, and R. Yang, “Web service recommendation based on time series forecasting and collaborative filtering,” in Web Services (ICWS), 2015 IEEE International Conference on. IEEE, 2015, pp. 233–240.
- [13] Jian W, Liang C, Feng Y, et al. Predicting Quality of Service for Selection by Neighborhood-Based Collaborative Filtering[J]. IEEE Transactions on Systems Man & Cybernetics Systems, 2013, 43(2):428-439.
- [14] Sun H, Zheng Z, Chen J, et al. Personalized Web Service Recommendation via Normal Recovery Collaborative Filtering[J]. IEEE Transactions on Services Computing, 2013, 6(4):573-579.
- [15] Zheng Z, Lyu M R. Personalized Reliability Prediction of Web Services[J]. Acm Transactions on Software Engineering & Methodology, 2013, 22(2):1-25.
- [16] Zheng Z, Lyu M R. Personalized Reliability Prediction of Web Services[M]. 2013.
- [17] Wang H, Chao Y, Lei W, et al. Effective BigData-Space Service Selection over Trust and Heterogeneous QoS Preferences[J]. IEEE Transactions on Services Computing, 2018, PP(99):1-1.
- [18] Ran S. A model for web services discovery with QoS[M]. 2003.
- [19] J. E. Hadad, M. Manouvrier, and M. Rukoz, “TQOS: Transactional and QoS-aware selection

- algorithm for automatic web service composition,” IEEE Trans. Serv. Comput., vol. 3, no. 1, pp. 73–85, Jan.-Mar. 2010.
- [20] K. Benouaret, D. Benslimane, and A. Hadjali, “Ws-sky: An efficient and flexible framework for QoS-aware web service selection,” in Proc. IEEE 9th Int. Conf. Serv. Comput., 2012, p. 146–153.
- [21] Xiong R , Wang J , Li Z , et al. Personalized LSTM Based Matrix Factorization for Online QoS Prediction[C]// 2018 IEEE International Conference on Web Services (ICWS). IEEE Computer Society, 2018.
- [22] Silic M, Delac G, Srblic S. Prediction of Atomic Web Services Reliability for QoS-Aware Recommendation[J]. IEEE Transactions on Services Computing, 2015, 8(3):425-438.