

一种基于隐语义概率模型的个性化 Web 服务推荐方法

胡 堰^{1,2} 彭启民¹ 胡晓惠¹

¹(中国科学院软件研究所天基综合信息系统重点实验室 北京 100190)

²(中国科学院大学 北京 100049)

(huyanlh@126.com)

A Personalized Web Service Recommendation Method Based on Latent Semantic Probabilistic Model

Hu Yan^{1,2}, Peng Qimin¹, and Hu Xiaohui¹

¹(*Science and Technology on Integrated Information System Laboratory, Institute of Software, Chinese Academy of Sciences, Beijing 100190*)

²(*University of Chinese Academy of Sciences, Beijing 100049*)

Abstract In order to meet service users' personalized requirements, a latent semantic probabilistic model is proposed to predict users' criteria preferences for Web service recommendation. Users' criteria preferences are mainly affected by two key elements, users and their service situations. Firstly, the latent semantic relations among users, their criteria preferences and service situations are established with latent classes in this model. In order to describe multifaceted characteristics of users, service situations and users' criteria preferences, all of them are allowed to simultaneously belong to multiple latent classes with different probabilities. Afterwards, the expectation maximization algorithm and the consistent training data obtained by analytic hierarchy process are used to estimate the parameters of the latent semantic probabilistic model which contains latent variables. Finally, the trained model is employed to predict users' criteria preferences under specific service situations if users are unwilling to provide their criteria preferences due to lack of domain knowledge. The main advantage of the proposed latent semantic probabilistic model over the standard memory-based collaborative filtering and the collaborative filtering improved by clustering is an explicit and compact model representation. And the experimental results show that the algorithm based on the latent semantic probabilistic model can get higher prediction accuracy than both the standard and the improved collaborative filtering algorithms and can also alleviate the impact of data sparsity.

Key words Web service recommendation; personalization; latent semantic probabilistic model; criteria preference; expectation maximization algorithm

摘 要 为了满足 Web 服务使用者的个性化需求,提出了一种基于隐语义概率模型的用户指标偏好预测方法,用于个性化 Web 服务推荐. 首先,引入两个决定用户指标偏好的关键因素:用户以及用户所处的服务情境,隐语义概率模型借助隐含类别建立用户指标偏好、用户及服务情境三者之间的隐语义依赖关系,并且为描述用户、服务情境、指标偏好多方面的特征,允许这三者可同时以不同的概率隶属于多个

收稿日期:2013-01-06;修回日期:2013-09-04

基金项目:国家“八六三”高技术研究发展计划基金项目(2012AA011206)

隐含类别;然后,将期望极大(expectation maximization, EM)算法运用于由层次分析法获得的训练数据,以估计隐语义概率模型的参数;最后,使用该模型预测用户在特定服务情境下的指标偏好. 隐语义概率模型与标准的基于内存的协同过滤以及基于聚类改进的协同过滤相比,不仅具有明确的数学模型,而且实验结果表明,隐语义概率模型对用户个性化指标偏好的预测精度最高,同时可以缓解数据稀疏性带来的不良影响.

关键词 Web 服务推荐;个性化;隐语义概率模型;指标偏好;期望极大算法

中图法分类号 TP311; TP181

随着互联网技术的飞速发展,面向服务的计算(service-oriented computing, SOC)^[1]作为一种新的计算模式得到了广泛应用. Web 服务就是这样一种分布运行于 Internet 之上、支持不同平台之间互操作的松耦合软件系统. 它主要通过“发布-查找-绑定”的模式允许服务使用者和提供者之间形成松散的绑定关系,这为 Web 服务的使用奠定了基础^[2]. 但 Web 服务的使用者和提供者相分离的特性,增加了服务使用者理解服务的难度. 与此同时,随着 Internet 上运行的 Web 服务数量不断增多,服务使用者需要从众多功能相似的 Web 服务中挑选出最符合自身需求的一个或一组服务,这对大多数缺乏专业知识的服务使用者来说无疑是一项繁重的任务,所以发展有效的 Web 服务推荐技术是 Web 服务选择的必然需求. 推荐技术是一种解决信息过载的重要手段,已在电子商务等领域得到了广泛应用. 其主要贡献是将用户潜在的、模糊的需求转化为现实的、明确的需求^[3]. Web 服务推荐与传统的推荐技术相比,用户具有明确的应用需求,只是面对大量功能类似的 Web 服务很难判断服务符合用户个性化需求的程度. 故需要通过服务质量特征综合评价 Web 服务,从而帮助服务使用者选择最适合自己的服务.

服务质量(quality of service, QoS)从多个方面描述了 Web 服务的非功能属性^[4-5],包括响应时间(response time)、可靠性(reliability)、可用性(availability)、价格(price)等众多方面. Web 服务推荐通过综合比较多个功能相似服务的服务质量,给出 Web 服务排序列表,作为 Web 服务推荐的依据. 然而由于时间、成本等因素的限制,服务提供者不可能在网络中布置大量软件传感器监测每个 Web 服务的质量信息,服务使用者也不可能调用大量的 Web 服务逐一体验其性能差异. 所以近年来,大量研究工作围绕如何利用相似用户的历史经验预测未知服务 QoS 信息的问题展开,也出现了不少有效的

预测方法,最常用的是基于内存的协同过滤^[6]. 基于内存的协同过滤在推荐领域已经得到了广泛应用,它利用用户爱好之间的相似性进行推荐,简单易行,颇受欢迎. 但这种方法的缺点也很明显,比如在数据稀疏的情况下,度量用户之间相似性的效果不佳. 故另一些研究工作^[7-9]利用聚类技术来改进基本的协同过滤,一定程度上克服了数据稀疏性的影响. 但是传统的聚类分析强制将每个参与聚类的对象归入唯一的类别,所以很难描述对象多方面的特征. 并且这些方法都以如何准确预测 Web 服务的未知 QoS 信息为出发点,而很少从用户个性化的角度考虑如何综合服务的各项 QoS 指标完成 Web 服务推荐. 这种千篇一律的非个性化 Web 服务推荐显然无法满足用户日渐增长的个性化需求. 所以 Web 服务推荐技术不仅要研究如何准确预测 Web 服务的未知 QoS 信息,同时也要融入用户对不同 QoS 指标的个性化偏好,这样的 Web 服务推荐才能更好地满足用户的个性化需求.

本文针对上述问题提出了一种预测用户个性化指标偏好的隐语义概率模型. 首先引入了影响用户指标偏好的关键因素:用户及用户所处的服务情境,并借助隐含类别建立三者之间的隐语义依赖关系;然后使用 EM 算法结合观测数据迭代估计隐语义概率模型的参数;最后用训练所得的模型参数预测用户在特定服务情境下的个性化指标偏好,为个性化的 Web 服务推荐提供支持. 本文的主要贡献包括:

- 1) 引入了影响用户指标偏好的关键因素——用户及服务情境,并用隐含类别建立了用户指标偏好及其影响因素之间的隐语义依赖关系,并采用 EM 算法估计含有隐变量的隐语义概率模型的参数;

- 2) 本文的隐语义概率模型基于隐含类别发掘用户或服务情境之间的关联关系,可有效缓解数据稀疏性带来的不良影响;并允许单个用户或单个服务情境可同时以不同的概率隶属于多个隐含类别,

方便描述用户、服务情境多方面的特征;

3) 本文的隐语义概率模型可以对用户各个 QoS 维度的指标偏好进行独立预测且不破坏其归一化约束;

4) 通过仿真实验验证了基于隐语义概率模型的指标偏好预测算法在预测精度和处理数据稀疏性方面优于基于内存的协同过滤和基于聚类改进的协同过滤。

1 相关工作

开放多变的网络环境使 Web 服务构成了一个不断成长、动态变化的服务空间,这使得 Web 服务的选择成了一个复杂又耗时的问题。在这种背景下,Web 服务推荐技术应运而生。服务质量(QoS)是 Web 服务推荐的主要依据。近年来,大量研究工作都在探讨如何利用相似用户的历史经验来预测当前用户对一些服务的未知质量评价^[10]。因此如何准确度量用户之间的相似性成了 Web 服务推荐的关键问题之一。传统的相似性度量方法有皮尔逊相关度量、余弦相关度量以及它们的一些变形^[11]。为了提高相似性度量精度,很多研究者对已有的相似性度量方法作了改进和拓展。文献[12]综合利用用户相似性和服务相似性来预测用户对服务的未知质量评价。文献[6]指出计算用户和服务相似性的时候要考虑用户和服务敏感性的影响,对某些用户敏感的服务在计算这些用户相似性时贡献越大,对某些服务敏感的用户其评价在计算服务相似性时贡献越大。文献[13]指出在服务推荐过程中要充分考虑协作推荐者的声誉,通过用户之间信任传播形成类似社会网络的信任网络,在信任网络中确定具有高声誉的相似用户,以得到高质量的服务推荐。文献[14]指出应根据用户的推荐等级、领域相关度和评价相似性对相似用户的推荐信息进行过滤,这样推荐结果才更为可信。

从上述这些研究工作中可以看出,如何准确预测候选服务的 QoS 信息是 Web 服务推荐领域的一个重要研究方向。但是仅仅获得候选服务的 QoS 信息对 Web 服务推荐而言仍远远不够,因为 Web 服务通常被看作一个多维事物,每个 QoS 维度都代表 Web 服务不同方面的质量特征,所以 Web 服务选择是一个多指标决策问题^[15]。因此另一些研究者开始探讨如何综合利用服务的各个 QoS 指标完成服务推荐。在这个问题上,用户对各个指标的偏好程度

是关键。文献[16]将用户的指标偏好融入用户相似性的计算中,但仅能区分用户对各指标重视与不重视两种偏好类别,没有给出用户指标偏好的定量化表示。文献[17]通过用户对不同服务各个指标的评价计算用户对各个指标的偏好值,但只能识别各指标受用户偏好程度的排序,并根据排序给各个指标分配事先确定的用户偏好权值,这种方法操作简单,但显然精度不够高。文献[18]使用已广泛用于分析人类认知决策规律的 UTA 算法,从用户对多个服务各 QoS 维度的评价及对这些服务的总体评价中计算出用户对各个 QoS 指标的偏好权重,此方法精度比较高,但是需要用户给出对多个服务的各 QoS 维度的评价值和和服务的总评价值,加重了用户负担。虽然这些研究工作尚存在一些不足,但我们仍可以看出,用户的个性化指标偏好在 Web 服务的推荐过程中起着举足轻重的作用。

本文针对上述研究工作中存在的不足,提出了一种隐语义概率模型预测用户的个性化指标偏好。首先探讨影响用户个性化指标偏好的关键因素;然后建立数学模型,描述用户个性化指标偏好及其影响因素之间的隐含语义依赖关系;最后通过 EM 算法训练模型参数。该数学模型可以在用户主观不情愿或缺乏领域相关知识无法提供指标偏好的情况下实现对用户个性化指标偏好的定量化预测。这种预测是一种连续值的预测,并非像文献[16-17]只能识别用户对各指标重视与否,或者只给出对各个指标偏好的排序并根据排序分配事先确定的权值,所以提高了预测精度;同时也不需要像文献[18]中所述的 UTA 算法要求用户给出对多个服务的各维度的评价以及总体评价,一定程度减轻了用户负担,可以让其获得良好的服务使用体验。除此之外,本文的模型还有助于缓解数据稀疏性带来的不良影响。

2 预测用户指标偏好的隐语义概率模型

2.1 隐语义概率模型定义

通过观察与分析,我们发现用户的个性化指标偏好主要由用户及用户所处的服务情境两个因素决定。因为只有将特定的用户置于特定的服务情境下,用户才会有特定的指标偏好。所谓的“服务情境”概括来说就是在何种场景下使用何种设备调用何种功能的 Web 服务。受文献[19]的启发,本文采用隐语义概率模型对用户指标偏好及其影响要素之间的依赖关系进行建模。

模型中涉及 3 个要素:用户、服务情境及用户指标偏好. 一个好的概率模型不仅需要对模型中的每一个要素进行建模, 还需要对各个要素之间的依赖关系进行建模^[12]. 因此我们希望隐语义概率模型能够满足以下 3 个条件: 1) 对用户和服务情境分别聚类; 2) 允许用户和服务情境可同时以不同的概率隶属于多个聚类, 方便描述用户和服务情境多方面的特征; 3) 用户聚类和服务情境聚类同时决定用户的指标偏好. 之所以在该模型中引入隐含类别, 主要出于两点考虑: 1) 很难事先人为确定用户和服务情境的分类, 较好的方法是通过对历史信息进行学习自动找到这些分类; 2) 单个用户、单个服务情境以及单个指标偏好向量与上述聚类之间的关系也难以事先确定, 同样需要从历史数据中学习得到. 所以上面所提的聚类皆为隐含类别, 简称隐类. 单个用户、单个服务情境、单个指标偏好向量与隐类之间的依赖关系皆为隐语义依赖关系.

下面给出隐语义概率模型中各要素的形式化定义. 每一个用户 u 由多个属性表征, 如姓名、性别、年龄、职业、兴趣爱好等属性. 每一个服务情境 e 由 3 个属性表征, 分别记为 w_1, w_2, w_3 , 其中: w_1 表示服务完成的具体功能, 如视频功能、短信功能、导航功能、语音功能等; w_2 表示用户使用该功能的服务完成的具体目标活动, 如学术会议、系统集成、军事导航、日常聊天等; w_3 表示用户调用服务的终端设备, 如手机、个人电脑等. 这 3 个属性联合组成的每一个服务情境都必须是现实应用中合理存在的. $R = \{R_1, R_2, \dots, R_K\}$ 是 Web 服务的 QoS 指标集合, 包括吞吐量、响应时间、可靠性、可用性、价格等. 用户的指标偏好由 K 维向量 $r = (r_1, r_2, \dots, r_K)$ 表示, 每一维 r_k ($0 \leq r_k \leq 1, 1 \leq k \leq K$) 表示用户对第 k 个 QoS 指标的偏好程度. r_k 越大意味着用户对第 k 个 QoS 指标越看重, 即第 k 个 QoS 指标在针对该用户的 Web 服务推荐中起的作用越大. 每一个指标偏好向量的各维度必须满足归一化约束, 即 $\sum_{k=1}^K r_k = 1$.

基于上述 3 个基本要素的定义以及隐语义概率模型需要满足的条件, 我们给出隐语义概率模型的形式化定义. $U = \{u_1, u_2, \dots, u_N\}$ 和 $E = \{e_1, e_2, \dots, e_M\}$ 分别是系统中所有用户的集合及所有服务情境的集合. $U = \{U_1, U_2, \dots, U_I\}$ 和 $E = \{E_1, E_2, \dots, E_J\}$ 分别表示用户隐类集合和服务情境隐类集合. $P(U_i)$ ($1 \leq i \leq I$) 表示用户隐类 U_i 出现的先验概率; $P(E_j)$ ($1 \leq j \leq J$) 表示服务情境隐类 E_j 出现的

先验概率. 个体与隐类之间的关系用条件概率表示: $P(u_n | U_i)$ ($1 \leq n \leq N, 1 \leq i \leq I$) 表示给定用户隐类 U_i 时, 用户 u_n 出现的条件概率; $P(e_m | E_j)$ ($1 \leq m \leq M, 1 \leq j \leq J$) 表示给定服务情境隐类 E_j 时, 服务情境 e_m 出现的条件概率. 由于用户指标偏好同时依赖于用户隐类和服务情境隐类, 所以用 $P(r | U_i, E_j)$ ($1 \leq i \leq I, 1 \leq j \leq J$) 表示给定用户隐类 U_i 和服务情境隐类 E_j 时, 指标偏好向量 r 出现的条件概率. 上述的这些先验概率和条件概率均为隐语义概率模型的参数, 且需要通过合适的统计学习方法从观测数据中学习得到. 隐语义概率模型的图形化表示如图 1 所示:

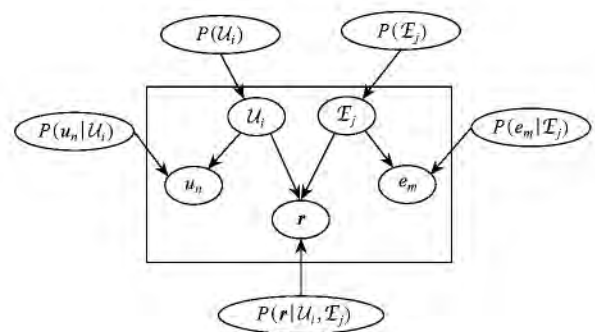


Fig. 1 Latent semantic probabilistic model.

图 1 隐语义概率模型

由于用户和服务情境都可能具有多方面的特征, 所以与传统聚类不同, 隐语义概率模型允许单个用户或者单个服务情境可以同时隶属于多个隐类, 只是属于不同隐类的概率不同. 若某用户的特征与某个隐类所代表的特征吻合得越好, 则该用户隶属于该隐类的概率越高, 反之越低. 所以一个隐类的特征往往由以较高概率属于该隐类的几个个体的特征主导. 其次考虑到只有把特定的用户置于特定的服务情境下, 才能真正决定用户对各 QoS 指标的偏好权重. 比如一个日程管理系统的集成方选取短信服务, 要考虑大量用户同时调用服务的情形, 因此会更关注该服务的吞吐量侧面; 而系统监测人员使用该短信服务进行系统异常报警则不会关心吞吐量侧面, 而更关注服务的响应时间、可靠性侧面. 所以用户隐类和服务隐类会同时影响用户的个性化指标偏好, 两者缺一不可.

2.2 历史经验数据的获取

历史经验数据是模型训练的关键之一, 因此模型建立之后, 首先要获取训练数据. 用户在其熟悉的服务情境下调用熟悉的 Web 服务时可以直接与系统交互指定自己对各个 QoS 指标的偏好权重, 这也

是本文训练数据的主要来源. 当然,若要求所有用户直接指定对每一个 QoS 指标的偏好权重是比较困难且过于理想的,而层次分析法^[20]提供了一种相对模糊的权重确定方法,可通过相对简单的判断来推测出用户对每一个 QoS 指标的偏好权重,所得结果的可信度往往比直接指定权重的可信度高. 层次分析法首先需要用户给出对指标偏好的两两比较结果,建立一个 QoS 指标偏好的两两比较矩阵. Saaty^[20]认为成对比较的等级不宜超过 9 个,各等级所代表的含义如表 1 所示:

Table 1 Meaning of Each Scale in Analytic Hierarchy Process

表 1 层次分析法中 9 个等级代表的含义

Scale	Definition
1	Equal importance
3	Moderate importance
5	Essential or strong importance
7	Very strong importance
9	Extreme importance
2,4,6,8	Intermediate values between the two adjacent judgments
Reciprocals	If criteria i has one of the above numbers assigned to it when compared with criteria j , j has the reciprocal value when compared with i

由上述方法建立的 QoS 指标偏好比较矩阵如式(1)所示:

$$C = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1K} \\ c_{21} & c_{22} & \cdots & c_{2K} \\ \vdots & \vdots & & \vdots \\ c_{K1} & c_{K2} & \cdots & c_{KK} \end{bmatrix}, \quad (1)$$

比较矩阵 C 中的每一个元素 c_{ij} 表示指标 i 相对指标 j 的重要程度.

定义 1. 完全一致性. 比较矩阵 C 中的元素如果同时满足以下 3 个约束: $c_{ii} = 1, c_{ij} = 1/c_{ji}, c_{ij} = c_{ik}/c_{jk}$, 则比较矩阵 C 具有完全一致性.

某些情况下,用户可能对自己的偏好认识不甚明确,导致比较矩阵 C 不是完全一致的. 定义一个一致性指标 CI 来衡量比较矩阵的一致性程度. 令 $CI = \frac{(\lambda_{\max} - K)}{(K - 1)}$, 其中: λ_{\max} 是比较矩阵 C 的最大特征值, K 是比较矩阵 C 的行数或列数. CI 越大表示比较矩阵 C 离完全一致性的差距越大. 引入随机一致性指标 RI , 作为一致性指标的参照, 表 2 给出了 1~8 阶的随机一致性指标 RI 的平均值^[20].

Table 2 Average of Random Consistency Index

表 2 1~8 阶的随机一致性指标的平均值

K	RI	K	RI
1	0	5	1.12
2	0	6	1.24
3	0.58	7	1.32
4	0.90	8	1.41

从表 2 中可以看出, $K < 3$ 时, 比较矩阵 C 必须完全一致; $K \geq 3$ 时, 若比较矩阵的一致性指标 CI 与随机一致性指标 RI 的比值 $CI/RI < 0.10$, 我们认为比较矩阵具有可接受的一致性, 否则, 需要重新调整比较矩阵直到其达到可接受的一致性. 当比较矩阵 C 具有完全一致性或可接受一致性之后, 计算 C 的最大特征值 λ_{\max} 对应的特征向量并归一化, 即为用户指标偏好向量.

2.3 基于 EM 算法的模型参数训练

EM 算法^[21], 即期望极大算法, 是一种迭代算法, 1977 年由 Dempster 等人总结得出, 主要用于含有隐含变量的概率模型的极大似然估计, 或极大后验概率估计, 已经被广泛应用于图像分析、自然语言处理等领域. 已有理论证明, EM 算法每一步都会使对数似然函数增大, 或达到局部极值; 文献[22]证明了 EM 算法总是收敛的, 包括似然函数收敛和模型参数收敛两层含义.

若概率模型只含有观测变量, 可直接使用极大似然估计或者贝叶斯估计法等常规统计学习方法估计模型参数; 但是当模型除了含有观测变量还含有隐变量时, 不能简单地使用这些估计方法, 而 EM 算法就是针对含有隐变量的概率模型参数的极大后验概率估计法, 适用于本文所提出的隐语义概率模型的参数估计. 假设 EM 算法训练隐语义概率模型时所需的训练数据为: $\{(u_{(1)}, e_{(1)}, r_{(1)}), (u_{(2)}, e_{(2)}, r_{(2)}), \dots, (u_{(L)}, e_{(L)}, r_{(L)})\}$, 是用户在其熟悉的服务情境下利用层次分析法给出的个性化指标偏好向量. 每个三元组 $(u_{(l)}, e_{(l)}, r_{(l)}) (1 \leq l \leq L)$ 表示用户 $u_{(l)}$ 在服务情境 $e_{(l)}$ 下对 Web 服务的 QoS 指标偏好向量为 $r_{(l)}$, 其中 $r_{(l)}$ 是一个 K 维向量 $(r_{1(l)}, r_{2(l)}, \dots, r_{K(l)})$. 因为用户的指标偏好值是介于 0~1 之间的连续值, 故为了简单起见, 假设偏好向量 r 的每一维相互独立并且服从正态分布. “独立性”假设方便对权重向量 r 的每一维进行独立分析, 降低模型复杂性; “正态分布”假设是为了使用正态分布的参数: 均值和方差, 准确描述在给定不同用户隐类和服务情境隐类时, 用户指标偏好的分布情况. 基于上述

假设,用户、服务情境和指标偏好三者的联合概率 $P(u_{(l)}, e_{(l)}, r_{(l)})$ 可用式(2)和式(3)表示:

$$P(u_{(l)}, e_{(l)}, r_{(l)}) = \sum_{i=1}^I \sum_{j=1}^J P(U_i) P(E_j) P(u_{(l)} | U_i) P(e_{(l)} | E_j) P(r_{(l)} | U_i, E_j), \quad (2)$$

其中,

$$P(r_{(l)} | U_i, E_j) = \prod_{k=1}^K P(r_{k(l)} | U_i, E_j) = \prod_{k=1}^K \frac{1}{\sqrt{2\pi}\sigma_{k,U_i,E_j}} \exp\left(-\frac{(r_{k(l)} - \mu_{k,U_i,E_j})^2}{2\sigma_{k,U_i,E_j}^2}\right). \quad (3)$$

E 步计算隐类的联合后验概率, M 步使用 E 步计算所得的后验概率重新估算模型参数. E 步和 M 步具体计算过程如式(4)~(12)所示.

E 步:

$$P(U_p, E_q | u_{(l)}, e_{(l)}, r_{(l)}) = \frac{(P(U_p) P(E_q) P(u_{(l)} | U_p) P(e_{(l)} | E_q) \times P(r_{(l)} | U_p, E_q))}{\left(\sum_{i=1}^I \sum_{j=1}^J P(U_i) P(E_j) \times P(u_{(l)} | U_i) P(e_{(l)} | E_j) P(r_{(l)} | U_i, E_j)\right)}, \quad (4)$$

其中,

$$P(r_{(l)} | U_p, E_q) = \prod_{k=1}^K \frac{1}{\sqrt{2\pi}\sigma_{k,U_p,E_q}} \exp\left(-\frac{(r_{k(l)} - \mu_{k,U_p,E_q})^2}{2\sigma_{k,U_p,E_q}^2}\right); \quad (5)$$

$$P(r_{(l)} | U_i, E_j) = \prod_{k=1}^K \frac{1}{\sqrt{2\pi}\sigma_{k,U_i,E_j}} \exp\left(-\frac{(r_{k(l)} - \mu_{k,U_i,E_j})^2}{2\sigma_{k,U_i,E_j}^2}\right). \quad (6)$$

$$P(U_p, E_q | u_{(l)}, e_{(l)}, r_{(l)}) =$$

$$\frac{\left(P(U_p) P(E_q) P(u_{(l)} | U_p) P(e_{(l)} | E_q) \prod_{k=1}^K \frac{1}{\sqrt{2\pi}\sigma_{k,U_p,E_q}} \exp\left(-\frac{(r_{k(l)} - \mu_{k,U_p,E_q})^2}{2\sigma_{k,U_p,E_q}^2}\right)\right)^b}{\sum_{i=1}^I \sum_{j=1}^J \left(P(U_i) P(E_j) P(u_{(l)} | U_i) P(e_{(l)} | E_j) \prod_{k=1}^K \frac{1}{\sqrt{2\pi}\sigma_{k,U_i,E_j}} \exp\left(-\frac{(r_{k(l)} - \mu_{k,U_i,E_j})^2}{2\sigma_{k,U_i,E_j}^2}\right)\right)^b}. \quad (13)$$

引入变量 $b(0 \leq b \leq 1)$ 的目的是为了防止隐类的后验分布在训练的开始阶段就被扭曲. 当 $b=0$ 时隐类的后验分布退化为均匀分布, 跟训练数据完全没有关系; 当 $b=1$ 时回归正常的 EM 算法. 退火 EM 算法将变量 b 由 0 逐渐增大至 1, 也就是将后验分布逐渐由均匀分布转变为由训练数据训练出来的模型, 这样可以避免在一开始就将训练数据完全投入训练造成的局部收敛问题.

2.4 用户未知指标偏好的预测

训练得到模型参数后, 当某个用户 u 在特定服务情境 e 下主观不愿意或缺乏领域知识无法给出指

M 步:

$$P(U_p) = \frac{\sum_{l=1}^L \sum_{j=1}^J P(U_p, E_j | u_{(l)}, e_{(l)}, r_{(l)})}{L}; \quad (7)$$

$$P(E_q) = \frac{\sum_{l=1}^L \sum_{i=1}^I P(U_i, E_q | u_{(l)}, e_{(l)}, r_{(l)})}{L}. \quad (8)$$

$$P(u_n | U_p) = \frac{\sum_{l: u_{(l)} = u_n} \sum_{j=1}^J P(U_p, E_j | u_{(l)}, e_{(l)}, r_{(l)})}{L \times P(U_p)}; \quad (9)$$

$$P(e_m | E_q) = \frac{\sum_{l: e_{(l)} = e_m} \sum_{i=1}^I P(U_i, E_q | u_{(l)}, e_{(l)}, r_{(l)})}{L \times P(E_q)}. \quad (10)$$

$$\mu_{k,U_p,E_q} = \frac{\sum_{l=1}^L r_{k(l)} P(U_p, E_q | u_{(l)}, e_{(l)}, r_{(l)})}{\sum_{l=1}^L P(U_p, E_q | u_{(l)}, e_{(l)}, r_{(l)})}, \quad (11)$$

$$\sigma_{k,U_p,E_q}^2 = \frac{\sum_{l=1}^L (r_{k(l)} - \mu_{k,U_p,E_q})^2 P(U_p, E_q | u_{(l)}, e_{(l)}, r_{(l)})}{\sum_{l=1}^L P(U_p, E_q | u_{(l)}, e_{(l)}, r_{(l)})}. \quad (12)$$

迭代执行 E 步和 M 步, 直至收敛. 但上述的 EM 算法往往会收敛到局部最优值, 而无法找到全局最优值, 所以我们采用退火 EM 算法^[23]来改进训练过程, 将 E 步修改为式(13):

标偏好时, 则可用式(14)~(16)对其进行预测. 因为已经假设用户对各个 QoS 指标的偏好是相互独立的, 所以预测时可以对用户的各个指标偏好值进行独立预测, 则用户 u 在服务情境 e 下对第 $k(1 \leq k \leq K)$ 个指标偏好值的预测结果为

$$R_k(u, e) = \frac{\int_0^1 r_k P(u, e, r_k) dr_k}{\int_0^1 P(u, e, r_k) dr_k}, \quad (14)$$

其中,

$$P(u, e, r_k) = \sum_{i=1}^I \sum_{j=1}^J P(U_i) P(E_j) \times$$

$$P(u|U_i)P(e|E_j)P(r_k|U_i,E_j), \quad (15)$$

$$P(r_k|U_i,E_j) = \frac{1}{\sqrt{2\pi}\sigma_{k,U_i,E_j}} \exp\left(-\frac{(r_k - \mu_{k,U_i,E_j})^2}{2\sigma_{k,U_i,E_j}^2}\right). \quad (16)$$

最后预测得到用户 u 在该服务情境 e 下对各个 QoS 指标的偏好权重: $(R_1(u,e), \dots, R_K(u,e))$, 简单记为 $(\hat{r}_1, \hat{r}_2, \dots, \hat{r}_K)$. 经证明(见附录 A), 此时 $\sum_{k=1}^K \hat{r}_k = 1$ 仍然成立.

若用户 u 和服务情境 e 都已被训练过, 则在预测过程中所有的模型参数都可直接使用; 但若用户 u 或者情境 e 没有被训练过, 只要保持模型已有参数固定不变, 只训练 $P(u|U_i) (1 \leq i \leq I)$ 和 $P(e|E_j) (1 \leq j \leq J)$ 即可. 这样, 当有新用户或者新的服务情境加入系统时, 可以方便地实现增量式训练. 考虑到服务推荐系统除了刚投入使用的阶段, 一般不会在短时间内涌现大量新用户或者新服务情境, 因此在服务推荐系统的运行过程中, 针对新用户或者新服务情境的增量式训练一般只需较少的时间开销. 且增量式训练不会影响模型已有的参数, 所以不会影响其他用户正常使用服务推荐系统. 退火 EM 算法的全局收敛性及增量式训练的特性可以保证基于隐语义概率模型的指标偏好预测算法稳定可靠地运行.

2.5 基于预测的指标偏好进行个性化服务推荐

用户 u 在服务情境 e 下会有多个功能相似的候选服务. 假设每个服务的各 QoS 指标上的性能已由已有技术预测得到, 记为 $(q_{i,1}, q_{i,2}, \dots, q_{i,K})$, 其中 $q_{i,j} (1 \leq j \leq K)$ 表示第 i 个候选服务在第 j 个 QoS 指标上的性能值. 采用文献[24]中的方法对 $(q_{i,1}, q_{i,2}, \dots, q_{i,K})$ 进行归一化. 归一化的目的是为了在综合评价 Web 服务质量的时候有一个统一的度量方法. 文献[24]将 QoS 指标分为“正 QoS 指标”和“负 QoS 指标”两类. “正 QoS 指标”是指其值越高, 服务质量越好的一类指标, 如“可靠性”、“可用性”、“吞吐量”等; 而“负 QoS 指标”是指其值越高, 服务质量越差的一类指标, 如“响应时间”、“价格”等. 现分别用式(17)和式(18)对正 QoS 指标和负 QoS 指标进行归一化:

$$q'_{i,j} = \begin{cases} \frac{q_{i,j} - Q_{\min}(j)}{Q_{\max}(j) - Q_{\min}(j)}, & Q_{\max}(j) \neq Q_{\min}(j); \\ 1, & Q_{\max}(j) = Q_{\min}(j). \end{cases} \quad (17)$$

$$q'_{i,j} = \begin{cases} \frac{Q_{\max}(j) - q_{i,j}}{Q_{\max}(j) - Q_{\min}(j)}, & Q_{\max}(j) \neq Q_{\min}(j); \\ 1, & Q_{\max}(j) = Q_{\min}(j). \end{cases} \quad (18)$$

其中 $q_{i,j}$ 和 $q'_{i,j}$ 分别表示第 i 个服务的第 j 个 QoS 指标性能归一化之前和之后的值, $Q_{\max}(j)$ 和 $Q_{\min}(j)$ 分别表示所有候选服务的第 j 个 QoS 指标性能归一化之前的最大值和最小值. 则第 i 个服务的总效用 T_i 可用式(19)表示:

$$T_i = \sum_{j=1}^K \hat{r}_j q'_{i,j}. \quad (19)$$

最后根据每个候选 Web 服务的总效用给出候选服务的排序, 作为针对用户 u 的个性化 Web 服务推荐的依据.

3 仿真实验

3.1 性能度量

平均绝对误差(mean absolute error, MAE)^[19] 是一种运用较广泛的误差度量方法, 本文也采用该方法度量隐语义概率模型对用户指标偏好的预测误差. 本文所采用的平均绝对误差定义为:

$$MAE = \frac{\sum_{i=1}^P \sum_{j=1}^K |r_{i,j} - \hat{r}_{i,j}|}{P \times K}, \quad (20)$$

其中 $r_{i,j}$ 表示第 i 个指标偏好向量中用户对第 j 个指标的实际偏好值, $\hat{r}_{i,j}$ 表示第 i 个指标偏好向量中用户对第 j 个指标的预测偏好值, P 是待预测的指标偏好向量的个数, K 是 QoS 指标个数. MAE 越小, 预测精度越高.

3.2 数据收集和实验环境

本文通过仿真系统, 结合“用户调查”^[25] 的方式获得用户指标偏好数据集, 验证本文算法的有效性. 首先实验过程中选取了 5 个常用的 QoS 指标: 响应时间、可靠性、可用性、吞吐量、价格, 来衡量 Web 服务的性能. 然后预设一定数量的服务情境, 每一种服务情境由一个三元组表示: (服务基本功能, 目标活动, 终端设备). 其中“服务基本功能”包括视频功能、短信功能、导航功能等; “目标活动”包括学术会议、系统集成、军事导航等; “终端设备”包括电脑、手机等. 经人工检验, 共得到 81 种合理可行的服务情境. 接着将这 81 种服务情境分发给 80 个不同职业、年龄、性别的 Web 服务调用者, 让他们在充分理解服务情境内涵的基础上填写指标偏好比较矩阵, 并用层次分析法计算出自己在当前服务情境下合理可靠的指标偏好向量. 最后选取 816 个(用户, 服务情境、用户指标偏好向量)三元组作为本文的实验数据集. 每个用户大约在 10 种不同的服务情境下提供了自己

的指标偏好向量,因此每个用户约对应 10 条实验数据. 本文实验的硬件环境为 Intel Core TM2 i3-2120 CPU(3.30 GHz), 2.00 GB 内存, 软件环境为 Matlab 2011b.

3.3 用户隐类和服务情境隐类个数的选取

在训练隐语义概率模型参数之前,为了在预测精度和预测时间之间进行权衡,首先需要确定用户隐类和服务情境隐类的个数. 之所以用“预测时间”而不是“训练时间”来衡量算法的时间复杂度,是因为模型参数的训练可以离线进行,而指标偏好的在线预测时间才是影响系统运行效率和用户满意度的直接因素,因此“预测时间”比“训练时间”更能衡量算法的时间性能好坏. 下面通过实验来观察隐类个数对预测精度和预测时间的影响.

从式(15)中我们可以看出,预测一个指标偏好向量需要取遍所有用户隐类和服务情境隐类的两两组合,所以理论上每个指标偏好向量的平均预测时间应和用户隐类个数与服务情境隐类个数的乘积成正比,下面我们通过实验来检验这种正比关系是否成立. 我们将用户隐类个数和服务情境隐类个数的选择范围都设定在 2~10 个之间,这样共形成 81 个不同的(用户隐类个数,服务情境隐类个数)点对. 针对每个(用户隐类个数,服务情境隐类个数)点对,我们分别从实验数据集中随机抽取 666 条作为训练数据来训练模型,剩下 150 条作为测试数据来测试模型对每个指标偏好向量的平均预测时间. 在 81 个不同的(用户隐类个数,服务情境隐类个数)点对下,平均预测时间的观测值如图 2 所示. 从图 2 中可以看出平均预测时间和两种隐类个数的乘积的确成正比关系.

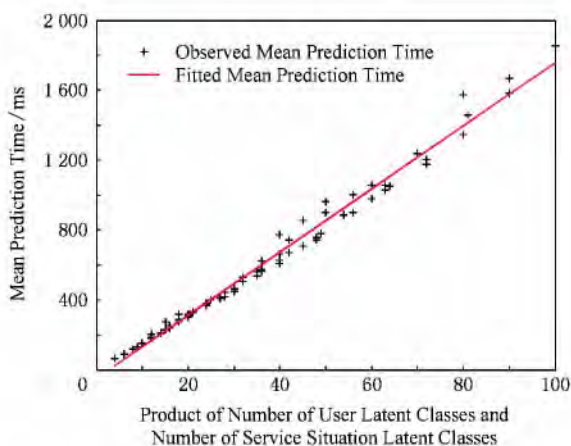


Fig. 2 Impact of the number of latent classes on mean prediction time.

图 2 隐类个数对平均预测时间的影响

下一步观察隐类个数对预测精度和预测时间的共同影响. 无论用户隐类还是服务情境隐类,其个数选为 1,显然起不到分类作用,所以两种隐类个数都必须大于等于 2;而图 2 的实验结果表明,隐类个数过多,平均预测时间会急剧增长,所以假设两种隐类个数均为 10 时对应的平均预测时间是最大可接受的预测时间. 因此用户隐类个数和服务情境隐类个数的选择范围仍设定在 2~10 个之间. 针对这 81 个不同的(用户隐类个数,服务情境隐类个数)点对,仍然从实验数据集中随机抽取 666 条作为训练数据训练模型,剩下的 150 条作为测试数据测试不同点对下模型的平均预测误差和平均预测时间. 图 3 中十点表示归一化的平均绝对误差, * 点表示归一化的平均预测时间,并用三次曲面分别对十点和 * 点进行拟合. 由于归一化操作是针对散点进行的,拟合曲面会有一部分位于 0~1 范围之外,但不影响对实验结果的分析.

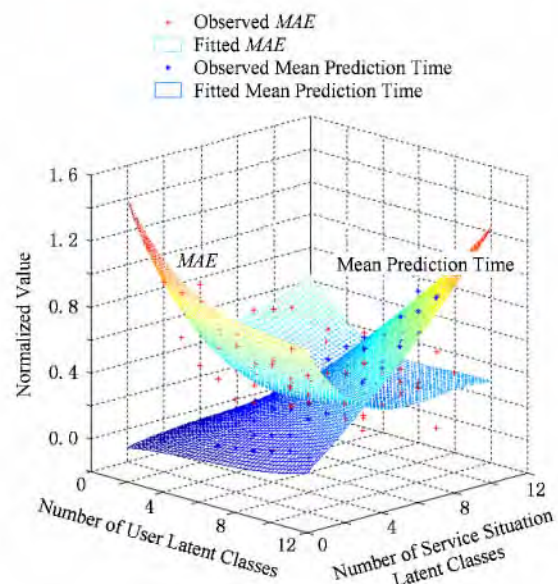


Fig. 3 Distribution of normalized MAE and normalized mean prediction time.

图 3 归一化平均绝对误差和归一化平均预测时间分布图

从图 3 中我们可以看出,平均预测时间随着两种隐类个数的增加而快速增长,而平均绝对误差在两种隐类的个数分别达到 5 左右的时候趋于平衡. 造成这种现象的原因是:每一个用户隐类描述的是用户某方面的特征,用户隐类个数选取过少,即特征数过少,模型无法准确区别具有不同特征的用户;相反,用户隐类个数过多,即特征数也过多,模型对用户作过细的区分,这样会造成分类的冗余,对预测精度的提升不再有帮助. 所以当用户隐类的个数达到

某个合适的值以后,便足够区分不同特征的用户,没有必要再增加用户隐类的个数.服务情境隐类个数的选取也是如此.

有了上述实验验证,只要将用户隐类个数和服务情境隐类个数分别取在 5 左右,既可以保证较高的预测精度,又不会消耗太多的预测时间.下面的性能比较实验中,我们将隐语义概率模型的用户隐类个数设为 4,服务情境隐类个数设为 5.

3.4 性能比较

本节中将隐语义概率模型和基于内存的混合协同过滤^[6]以及基于聚类改进的协同过滤算法^[7-9]进行对比.基于聚类改进的协同过滤算法运用于本文的个性化指标偏好预测场景时,又可以细分为两种:基于用户聚类和服务情境聚类的预测算法^[7-8](以下简称 UEKmeans)、基于指标偏好向量聚类的预测算法^[9](以下简称 PKmeans).

为了比较本文提出的基于隐语义概率模型的个

性化指标偏好预测算法和其他 3 种预测算法的预测精度,我们从 816 条实验数据中随机抽取 150 条作为测试数据,剩下的 666 条作为训练数据训练隐语义概率模型的参数.其他 3 种算法虽然不用训练模型参数,但相似性度量、聚类分析时的距离度量都要用到训练数据.为了避免单次实验中测试数据选取的局部性导致实验结果的偶然性,我们一共进行 20 次实验,每次都从整个实验数据集中随机抽取 150 条作为测试数据,余下的 666 条作为训练数据,这样基本保证整个实验数据集上的数据都有可能成为测试数据.最后汇总这 20 次实验的所有预测绝对误差(每次实验选取了 150 条测试数据,每条测试数据包含 5 个指标偏好值,所以 20 次实验总共有 15000 个绝对误差数据)绘成直方图,并进行正态拟合,这样获得的平均绝对误差可以比较客观地度量各个算法的预测精度.4 种算法的预测绝对误差直方图分布如图 4 所示:

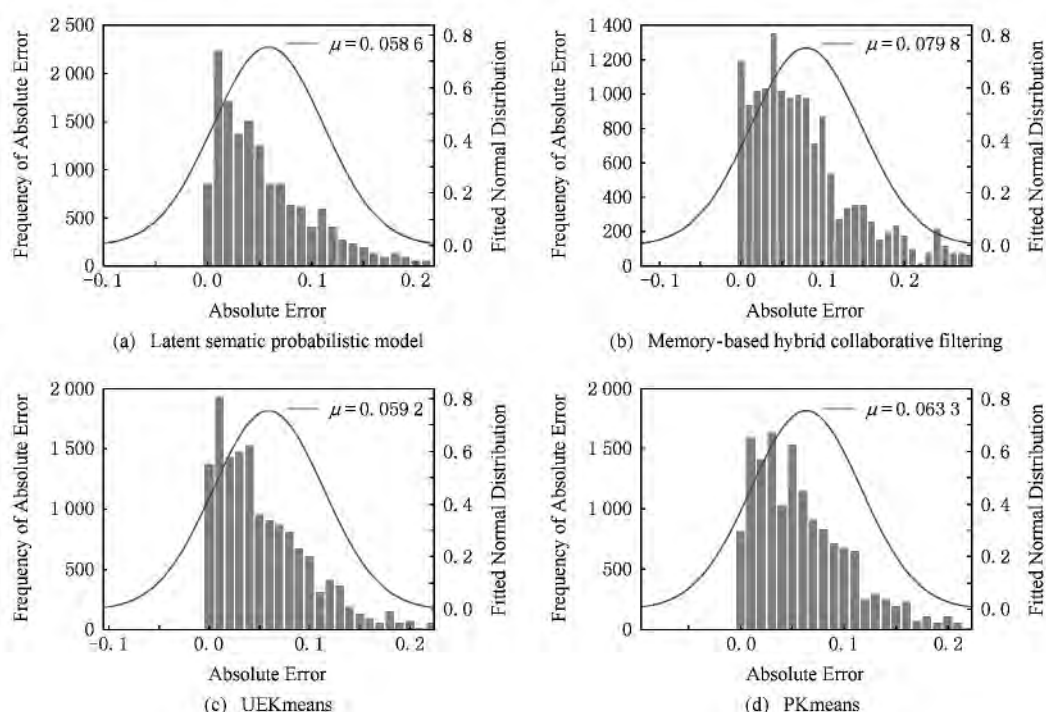


Fig. 4 Absolute error distribution comparison of different algorithms.

图 4 不同算法对用户偏好预测误差分布情况对比

从图 4 中可以看出,基于隐语义概率模型的预测算法精度最高,因为该算法允许单个用户或者单个服务情境同时隶属于多个类别,且基于类别发掘用户之间或者服务情境之间的关联性;UEKmeans 和 PKmeans 精度次之,因为这两种算法都将参与聚类的每个对象强制划分至唯一的类别,无法描述每个对象多方面的特征,所以预测精度会比基于隐语

义概率模型的预测算法低;而基于内存的混合协同过滤精度最低,这种算法不是基于类别而是基于个体发掘用户之间或者服务情境之间的相似性,所以当两个用户之间共享的服务情境很少时,或者两个服务情境共享的用户很少时,这种算法在度量相似性时很不准确,故预测精度会明显低于前 3 种算法.

为了更进一步比较在不同规模的训练数据集

下,基于隐语义概率模型的指标偏好预测算法与其他3种算法的性能优劣,我们从整个实验数据集中随机抽取的测试数据为10~270条不等,步长为30条,剩下的作为训练数据,即一共有10种测试数据规模,也对应着10种不同的训练数据规模.然后对每一种训练数据规模,都进行20次的训练数据随机抽取,以避免实验结果的偶然性.最后分别用4种算法进行测试,得到4种算法的预测绝对误差正态拟合均值,再对20次绝对误差拟合均值求算

术平均.

表3的实验结果表明隐语义概率模型的预测精度在不同规模的训练数据集下仍明显高于其他3种算法.同时,从该实验结果中我们还可以观察到,隐语义概率模型的预测精度随着抽取的测试数据的减少而增加,我们猜想这主要是由于测试数据的减少导致训练数据的增多,而训练数据增多意味着历史信息更为丰富,训练所得模型更为精确,其他3种算法也都有这个现象.

Table 3 MAE Comparison of Four Algorithms on Different Test Sets of Different Sizes

表3 4种算法在不同规模测试数据集上的平均绝对误差对比

Number of Randomly Selected Test Examples	Arithmetic Average of MAE of 20 Experiments			
	Latent Semantic Probabilistic Model	Memory-based Hybrid Collaborative Filtering	UEKmeans	PKmeans
270	0.0594	0.0822	0.0630	0.0876
240	0.0585	0.0818	0.0610	0.0766
210	0.0583	0.0812	0.0608	0.0733
180	0.0589	0.0802	0.0595	0.0636
150	0.0586	0.0798	0.0592	0.0633
120	0.0567	0.0797	0.0589	0.0616
90	0.0548	0.0792	0.0582	0.0610
60	0.0555	0.0772	0.0580	0.0605
30	0.0541	0.0795	0.0576	0.0603
10	0.0523	0.0762	0.0574	0.0600

为了验证上面的猜想,我们又进行了另一组实验.这组实验中,我们从80个测试用户中随机抽取40个用户作为测试用户,其余40个用户对应的所有指标偏好数据全部用于训练.对于所选的40个测试用户,我们共进行7组实验.在第 $m(1 \leq m \leq 7)$ 组实验过程中,从每个测试用户的不同服务情境下的指标偏好数据中随机抽取 m 条也放入训练数据集,测试用户的其余指标偏好数据作为测试数据集.再

分别用4种算法进行训练和测试,每组实验同样进行20次.

表4的实验结果证明我们的猜想是成立的.每个测试用户随机抽取7条指标偏好数据用于训练得到的预测平均绝对误差最小,而只抽取1条指标偏好数据用于训练得到的预测平均绝对误差最大,这个结论对于4种算法均成立.同时,我们还发现除了基于隐语义概率模型的预测算法外,其他3种算法

Table 4 MAE Comparison of Four Algorithms Given Different Numbers of Training Examples from Every Test User

表4 4种算法在各测试用户给定不同数目训练数据时的平均绝对误差对比

Number of Training Examples from Every Test User	Arithmetic Average of MAE of 20 Experiments			
	Latent Semantic Probabilistic Model	Memory-based Hybrid Collaborative Filtering	UEKmeans	PKmeans
7	0.0558	0.0668	0.0545	0.0561
6	0.0584	0.0724	0.0583	0.0587
5	0.0591	0.0724	0.0623	0.0596
4	0.0578	0.1076	0.0715	0.0655
3	0.0607	0.1656	0.0709	0.0719
2	0.0627	0.1757	0.0819	0.0800
1	0.0789	0.2244	0.1269	0.1075

受数据稀疏性的影响都比较大,也就是随着测试用户提供的训练数据减少,算法的预测精度下降很快.基于内存的混合协同过滤在数据稀疏时预测性能最差,因为若当前用户提供的历史信息很少,要准确地找到当前用户的相似用户以及当前服务情境的相似服务情境则比较困难;经过聚类改进的协同过滤通过聚类分析可以发现更多相似用户和相似服务情境,所以预测精度比基本的协同过滤高一些,但是仍局限于将每个用户或服务情境划入唯一的类别,不足以描述它们多方面的特征;隐语义概率模型将用户和服务情境进行隐式聚类,并用历史数据训练得到用户、服务情境、指标偏好向量与各个隐类的概率依赖关系,同时允许单个用户、单个服务情境可同时以不同的概率隶属于多个隐类,指标偏好向量的每个维度都可以在不同的隐类下有不同的正态分布特征,隐语义概率模型具有的这些优点可以让它比其他算法能更有效地缓解数据稀疏性带来的不良影响.

4 总结与展望

Web 服务由多个 QoS 指标综合决定其质量,所以 Web 服务的推荐是一个多指标决策问题.已有的 Web 服务推荐技术集中研究了如何准确度量用户之间的相似性并利用相似用户的历史经验信息预测服务的未知 QoS 信息,却没有深入考虑如何从个性化的角度综合服务的各项 QoS 指标来评价服务的总体性能.本文提出的隐语义概率模型很好地解决了这个问题.隐语义概率模型不仅对用户、服务情境、用户的指标偏好这 3 个要素进行隐式聚类,还建立了用户指标偏好和用户、服务情景之间的隐语义概率模型,并用于预测用户在特定服务情境下的个性化指标偏好.仿真实验验证了隐语义概率模型可以取得比基于内存的混合协同过滤以及聚类改进的协同过滤更高的预测精度,且能更有效地缓解数据稀疏性带来的不良影响.下一步工作将讨论如何提升基于隐语义概率模型的指标偏好预测算法的效率,包括离线训练和在线预测效率;并研究如何利用语义网等技术自动生成合理的服务情境,或者自动验证服务情境的合理性.

参 考 文 献

[1] Zhang Liangjie, Zhang Jia, Cai Hong. Services Computing [M]. Beijing: Tsinghua University Press, 2007

- [2] Zhang Cheng. Personalized service recommendation in service-oriented environment [D]. Beijing: Institute of Computing Technology, Chinese Academy of Sciences, 2006 (in Chinese)
(张程. 面向服务环境中服务的个性化推荐[D]. 北京: 中国科学院计算技术研究所, 2006)
- [3] Xu Hailing, Wu Xiao, Li Xiaodong, et al. Comparison study of Internet recommendation system [J]. Journal of Software, 2009, 20(2): 350-362 (in Chinese)
(许海玲, 吴潇, 李晓东, 等. 互联网推荐系统比较研究[J]. 软件学报, 2009, 20(2): 350-362)
- [4] Zhang Peiyun, Huang Bo, Sun Yamin. A Web service matching mechanism based on semantics and QoS-aware aspect [J]. Journal of Computer Research and Development, 2010, 47(5): 780-787 (in Chinese)
(张佩云, 黄波, 孙亚民. 一种基于语义与 QoS 感知的 Web 服务匹配机制[J]. 计算机研究与发展, 2010, 47(5): 780-787)
- [5] Wan Changlin, Shi Zhongzhi, Hu Hong, et al. QoS-aware semantic Web service modeling and discovery [J]. Journal of Computer Research and Development, 2011, 48(6): 1059-1066 (in Chinese)
(万长林, 史忠植, 胡宏, 等. 基于本体的语义 Web 服务 QoS 描述和发现[J]. 计算机研究与发展, 2011, 48(6): 1059-1066)
- [6] Zheng Zibin, Ma Hao, Lyu MR, et al. QoS-aware Web service recommendation by collaborative filtering [J]. IEEE Trans on Services Computing, 2011, 4(2): 140-152
- [7] Gong Songjie. A collaborative filtering recommendation algorithm based on user clustering and item clustering [J]. Journal of Software, 2010, 5(7): 745-752
- [8] Wen Junhao, Zhou Wei. An improved item-based collaborative filtering algorithm based on clustering method [J]. Journal of Computational Information Systems, 2012, 8(2): 571-578
- [9] Jeong B, Lee J, Cho H. Improving memory-based collaborative filtering via similarity updating and prediction modulation [J]. Information Sciences, 2010, 180(5): 602-612
- [10] Zhang Li, Zhang Bin, Huang Liping, et al. A personalized Web service quality prediction approach based on invoked feature model [J]. Journal of Computer Research and Development, 2013, 50(5): 1066-1075 (in Chinese)
(张莉, 张斌, 黄利萍, 等. 基于服务调用特征模式的个性化 Web 服务 QoS 预测方法[J]. 计算机研究与发展, 2013, 50(5): 1066-1075)
- [11] Ahn HJ. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem [J]. Information Sciences, 2008, 178(1): 37-51
- [12] Zheng Zibin, Ma Hao, Lyu M R, et al. Wsrec: A collaborative filtering based Web service recommender system [C] //Proc of the 7th IEEE Int Conf on Web Services (ICWS 2009). Piscataway, NJ: IEEE, 2009: 437-444

- [13] Pan Jing, Xu Feng, Lü Jian. Reputation-based recommender discovery approach for service selection [J]. Journal of Software, 2010, 21(2): 388-400 (in Chinese)
(潘静, 徐锋, 吕建. 面向可信服务选取的基于声誉的推荐者发现方法[J]. 软件学报, 2010, 21(2): 388-400)
- [14] Zhu Rui, Wang Huaimin, Feng Dawei. Trustworthy services selection based on preference recommendation [J]. Journal of Software, 2011, 22(5): 852-864 (in Chinese)
(朱锐, 王怀民, 冯大为. 基于偏好推荐的可靠服务选择[J]. 软件学报, 2011, 22(5): 852-864)
- [15] Shapira B. Recommender Systems Handbook [M]. Berlin: Springer, 2011: 769-803
- [16] Liu Liwei, Mehandjiev N, Xu Donglin. Multi-criteria service recommendation based on user criteria preferences [C] //Proc of the 5th ACM Conf on Recommender Systems. New York: ACM, 2011: 77-84
- [17] Maneeroj S, Samatthiyadikun P, Chalermponpong W, et al. Ranked criteria profile for multi-criteria rating recommender [C] //Proc of the 6th Int Conf on Information Systems, Technology and Management (ICISTM 2012). Berlin: Springer, 2012: 40-51
- [18] Lakiotaki K, Matsatsinis N F, Tsoukiàs A. Multicriteria user modeling in recommender systems [J]. Intelligent Systems, 2011, 26(2): 64-76
- [19] Jin R, Si L. Flexible mixture model for collaborative filtering [C] //Proc of the 20th Int Conf on Machine Learning (ICML 2003). Menlo Park, CA: AAAI, 2003: 704-711
- [20] Saaty R W. The analytic hierarchy process—What it is and how it is used [J]. Mathematical Modelling, 1987, 9(3/4/5): 161-176
- [21] Dempster A P, Laird N M, Rubin D B. Maximum likelihood from incomplete data via the EM algorithm [J]. Journal of the Royal Statistical Society: Series B Methodological, 1977, 39(1): 1-38
- [22] Wu C F. On the convergence properties of the EM algorithm [J]. The Annals of Statistics, 1983, 11(1): 95-103
- [23] Hofmann T, Puzicha J. Statistical models for co-occurrence data, AI Memo 1625 [R]. Cambridge: Artificial Intelligence Laboratory, MIT, 1998
- [24] Kang Guosheng, Liu Jianxun, Tang Mingdong, et al. AWSR: Active Web service recommendation based on usage history [C] //Proc of the IEEE 19th Int Conf on Web Services (ICWS 2012). Piscataway, NJ: IEEE, 2012: 186-193
- [25] Xiang Liang. Recommender System Practice [M]. Beijing: The People's Posts and Telecommunications Press, 2012: 20 (in Chinese)
(项亮. 推荐系统实践[M]. 北京: 人民邮电出版社, 2012: 20)



Hu Yan, born in 1988. PhD candidate. Student member of China Computer Federation. Her main research interests include Web service computing and information system integration.



Peng Qimin, born in 1969. Associate professor and master supervisor in the Science and Technology on Integrated Information System Laboratory, Institute of Software, Chinese Academy of Sciences. His main research interests include Web service computing, information system integration, image processing and pattern recognition (qimin@iscas.ac.cn).



Hu Xiaohui, born in 1960. Professor and PhD supervisor in the Science and Technology on Integrated Information System Laboratory, Institute of Software, Chinese Academy of Sciences. His main research interests include information system integration, Web service computing and system simulation (hxx@iscas.ac.cn).

附录 A. 证明 $\sum_{k=1}^K \hat{r}_k = 1$.

证明.

将正文式(14)代入所要证明的式子,得:

$$\sum_{k=1}^K \hat{r}_k = \sum_{k=1}^K \int_0^1 r_k \frac{P(u, e, r_k)}{\int_0^1 P(u, e, r_k) dr_k} dr_k = \frac{\sum_{k=1}^K \int_0^1 r_k P(u, e, r_k) dr_k}{\int_0^1 P(u, e, r_k) dr_k}. \quad (A1)$$

又因为

$$\int_0^1 P(u, e, r_k) dr_k = P(u, e), \quad (A2)$$

且有:

$$P(u, e, r_k) = \sum_{i=1}^I \sum_{j=1}^J P(U_i) P(E_j) P(u | U_i) \times P(e | E_j) P(r_k | U_i, E_j), \quad (A3)$$

将式(A2)与式(A3)代入式(A1),得:

$$\sum_{k=1}^K \hat{r}_k = \left(\sum_{k=1}^K \int_0^1 r_k \sum_{i=1}^I \sum_{j=1}^J P(U_i) P(E_j) \times P(u | U_i) P(e | E_j) P(r_k | U_i, E_j) dr_k \right) / (P(u, e)). \quad (A4)$$

将式(A4)分子的求和与积分重新排布,得:

$$\sum_{k=1}^K \hat{r}_k = \left(\sum_{i=1}^I \sum_{j=1}^J P(U_i) P(E_j) P(u | U_i) P(e | E_j) \times \right.$$

$$\sum_{k=1}^K \int_0^1 r_k P(r_k | U_i, E_j) dr_k) / (P(u, e)). \quad (A5)$$

又因为

$$\int_0^1 r_k P(r_k | U_i, E_j) dr_k = \mu_{k, U_i, E_j}, \quad (A6)$$

将式(A6)代入式(A5),得:

$$\begin{aligned} \sum_{k=1}^K \hat{r}_k &= \\ \frac{\sum_{i=1}^I \sum_{j=1}^J P(U_i) P(E_j) P(u | U_i) P(e | E_j) \sum_{k=1}^K \mu_{k, U_i, E_j}}{P(u, e)}. \end{aligned} \quad (A7)$$

将正文式(11)代入式(A7),得:

$$\begin{aligned} \sum_{k=1}^K \hat{r}_k &= \left[\sum_{i=1}^I \sum_{j=1}^J P(U_i) P(E_j) P(u | U_i) P(e | E_j) \times \right. \\ &\quad \left. \sum_{k=1}^K \frac{\sum_{l=1}^L r_{k(l)} P(U_i, E_j | u_{(l)}, e_{(l)}, \mathbf{r}_{(l)})}{\sum_{l=1}^L P(U_i, E_j | u_{(l)}, e_{(l)}, \mathbf{r}_{(l)})} \right] / P(u, e) = \end{aligned}$$

$$\begin{aligned} &\left[\sum_{i=1}^I \sum_{j=1}^J P(U_i) P(E_j) P(u | U_i) P(e | E_j) \times \right. \\ &\quad \left. \frac{\sum_{l=1}^L P(U_i, E_j | u_{(l)}, e_{(l)}, \mathbf{r}_{(l)}) \sum_{k=1}^K r_{k(l)}}{\sum_{l=1}^L P(U_i, E_j | u_{(l)}, e_{(l)}, \mathbf{r}_{(l)})} \right] / P(u, e). \end{aligned} \quad (A8)$$

又因为

$$\sum_{k=1}^K r_{k(l)} = 1, \quad (A9)$$

将式(A9)代入式(A8),得:

$$\begin{aligned} \sum_{k=1}^K \hat{r}_k &= \frac{\sum_{i=1}^I \sum_{j=1}^J P(U_i) P(E_j) P(u | U_i) P(e | E_j)}{P(u, e)} = \\ &\quad \frac{P(u, e)}{P(u, e)} = 1. \end{aligned} \quad (A10)$$

证毕.