

云服务流定制中的个性化资源推荐方法研究

胡彬彬, 吴绍春

(上海大学 计算机工程与科学学院, 上海 200444)

摘要:文中针对数字海洋云平台服务流可视化定制过程中出现的问题,着力解决用户如何快速从海量资源中选择符合需要的资源的难题,提出了服务流定制过程中的个性化用户资源推荐模型。该模型利用已定制过的服务流提取用户的行为习惯信息,融合用户与资源、资源与资源之间的关系,同时结合伙伴用户的思想,将在使用资源方面具有相似偏好的用户联系起来,得到极具个性化的用户资源推荐列表。与此同时,文中在用户资源推荐模型的基础上提出一种个性化资源推荐算法。实验结果表明,该算法极大地提高了用户定制服务流的效率。

关键词:云服务流定制;伙伴用户;个性化;资源推荐

中图分类号:TP391

文献标识码:A

文章编号:1673-629X(2015)03-0108-06

doi:10.3969/j.issn.1673-629X.2015.03.025

Research on Personalized Resource Recommendation Method in Service-flow Customization

HU Bin-bin, WU Shao-chun

(School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China)

Abstract: Aiming at the problem during customization process in the digital ocean cloud service flow visualization, in order to solve the difficulty how to quickly address users to select the resources conforms to the need from huge amounts of resources, put forward the personalized user resources recommended models in the process of service flow customization. The model uses the customized service-flow to extract the users behavior information and integrate the relationship between users and resource, resource and resource. Furthermore, it links users with the similar interesting in resources application to get the personalized resource recommendation list through the idea of company user. Meanwhile, based on the user-resource recommendation model, put forward a personalized resource recommendation algorithm. The experimental results show that the proposed algorithm greatly improves the efficiency of users service-flow customization.

Key words: cloud service-flow customization; company user; personalized; resource recommendation

0 引言

云服务流的可视化定制是海洋公益性行业科研专项项目研究开发的数字海洋云平台的研究内容之一,是利用云服务流技术完成海洋资源计算与共享的关键环节。目前,数字海洋云平台已经实现了服务流的拖拽式可视化定制。随着海洋云平台上公有服务/数据资源和用户私有服务/数据资源数目的不断累积,用户定制服务流时选择符合需要的资源的效率将会变得越来越低。针对这个问题,文中通过分析用户已定制服务流中使用资源的行为习惯以及用户间使用资源的相似性,建立用户资源推荐模型,并在该模型的基础上提出个性化资源推荐方法,由此改善海洋云平台用户从

海量资源中选择符合需要资源定制服务流的难题。

1 相关工作

目前,研究人员对推荐技术的研究大部分集中在针对商品、文献、音乐、影视等的推荐上^[1],而其中大部分研究都是基于最基础的协同过滤技术进行的,比如文献[2-5]。软件即服务的提出使得研究人员开始考虑使用推荐技术为用户推荐服务^[6]。综合这些推荐技术,主要出现了基于向量空间模型、基于神经网络、基于用户-项目评价矩阵和基于本体论的用户模型^[7-8]。基于向量空间模型的推荐技术仅用一组关键词来表示用户喜好,使推荐结果不全面;基于神经网络模型^[9-10]

收稿日期:2014-05-12

修回日期:2014-08-14

网络出版时间:2015-01-20

基金项目:国家“十二五”规划基金项目(201105033)

作者简介:胡彬彬(1988-),男,硕士研究生,研究方向为服务推荐;吴绍春,博士,教授,研究方向为智能信息处理、数据挖掘。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20150120.2203.041.html>

的推荐效果依赖于所选网络类型和训练样本及方法,适用范围较窄且不易理解;基于用户-项目评价矩阵模型^[11]存在矩阵稀疏和冷启动等问题;基于本体论的模型由于构造语义本体本身是一件费时的工 作,而且还要依赖于构建本体的专家的知识 和经验,因此该模型的推荐准确率有很大的不确定性。

文中综合考虑云服务流的结构特点,在张少中^[12]的两层混合图模型和 Charu C. Aggarwal^[13]的基于 Horting 图理论推荐方法的启发下,提出了具有服务流特色的用户资源推荐模型。构建文中的资源推荐模型必须依赖于云服务流的描述方法。采用一种基于 XML 技术的服务流描述方式。云服务流由服务资源与数据资源按照用户需求而设定的执行逻辑构成,如图 1 所示。因此,在服务流的 XML 描述中,需包含服务资源节点、数据资源节点以及数据流向关系节点,其中数据流向关系节点是服务流的核心,它将节点按照用户的业务需求正确地连接起来,它是各个节点之间的纽带。

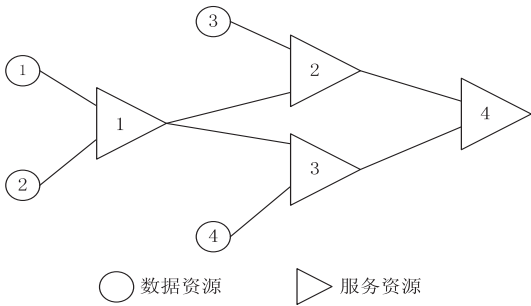


图 1 云服务流结构

文中以云服务流的结构为基础,综合考虑云平台中用户使用资源的相似性、用户定制服务流使用资源的偏好和服务流中资源之间的关系,构建用户资源推荐模型。在此模型的基础上,设计实现了资源推荐算法,改善了用户定制服务流时如何快速选择合适资源的难题。

2 用户资源推荐模型

用户资源推荐模型全面描述用户使用资源的模式。它以用户为中心,综合了用户历史使用的资源、用户的伙伴用户和用户历史使用过的资源与资源间的关系。该模型利用伙伴用户表达用户与用户之间的关系,利用使用资源列表表达用户与资源间的关系,利用模型节点关系表达用户定制服务流时使用过的资源与资源间的关系。基于用户资源推荐模型的资源推荐算法将用户自身使用资源习惯作为首要推荐依据,将与用户有相同使用资源偏好的其他用户使用的资源作为可推荐资源。最后,综合考虑资源间的输入输出约束,得到个性化的推荐资源列表。

图 2 所示为用户 U_1 定制的 2 条服务流生成的资源推荐模型。在模型的上方通过伙伴关系联系用户 U_2 ,表示用户 U_2 是 U_1 的伙伴用户。用户 U_1 在定制这两条服务流时使用的资源被保存在模型下方的用户使用资源列表中。模型节点关系部分保存了服务流中各资源之间的关系。

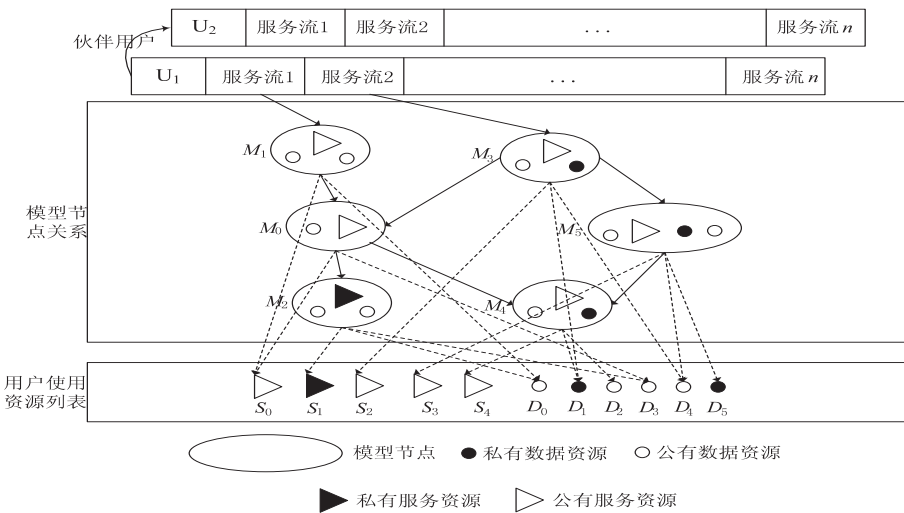


图 2 用户资源推荐模型组织结构

2.1 相关定义

- (1)模型节点(M):模型节点由一个服务资源和与它直接相连的多个数据资源组成。
- (2)资源实例(RI):模型节点中包含的资源。
- (3)后续模型节点集合(FOLLOWMs):对于模型

节点 M ,后续节点包括它的前驱和后继,设 M 的前驱和后继集合分别为 pre_M 和 $next_M$,则 M 的后续节点集合为 $FOLLOWMs = pre_M \cup next_M$ 。如图 2 所示,对于模型节点 M_0 ,其前驱集合为 $\{M_1, M_3\}$,后继集合为 $\{M_2, M_4\}$,则 M_0 的后续模型节点集合为 $\{M_1, M_2,$

$M_3, M_4\}$ 。

(4) 后续资源集合 (FOLLOWRs): 对于资源 R , 若 R 为数据资源, 设在资源推荐模型中与 R 直接相连的服务资源集合为 dir_S , 与 R 连接在同一个服务资源的其他数据资源集合为 $sames_D$, 则 R 的后续资源集合为 $FOLLOWRs = dir_S \cup sames_D$ 。如图 2 所示, 对于数据资源 D_3 , $dir_S = \{S_3\}$, $sames_D = \{D_4\}$, 则其后续资源集合为 $\{S_3, D_4\}$ 。若 R 为服务资源, 设在资源推荐模型中与 R 直接相连的数据资源集合为 dir_D , 与 R 有输入输出联系的服务资源集合为 dir_S , 则 R 的后续资源集合为 $FOLLOWRs = dir_S \cup dir_D$ 。如图 2 所示, 对于服务资源 S_0 , $dir_D = \{D_0, D_3\}$, $dir_S = \{S_0, S_1, S_2, S_4\}$, 则其后续资源集合为 $\{S_0, S_1, S_2, S_4, D_0, D_3\}$ 。

(5) 资源对用户的历史活跃度 (HA): 某个资源被某个用户历史使用的次数, 其值大于等于 0。

(6) 伙伴用户: 设用户 i, j 使用过的历史公有资源对 i, j 的 HA 向量分别为 $P(i), P(j)$, 定义阈值 $t (0 < t < 1)$, 用户 i 与用户 j 之间的相似度用 $Sim(i, j)$ 表示, 定义 $length(P)$ 表示向量的长度, $same(P_1, P_2)$ 表示向量 P_1 和 P_2 中相同属性对应值构成的向量, 若存在 $Sim(i, j) > t$, 则用户 j 为用户 i 的伙伴用户。

说明: 伙伴用户的相似度 $Sim(i, j)$ 的计算方法如公式 (1) 所示:

$$Sim(i, j) = \frac{p(i) * p(j)}{|p(i)| * |p(j)|} = \frac{\sum_{k=1}^{length(same(p(i), p(j)))} same(p(i) * p(j))_k * same(p(i) * p(j))_k}{\sqrt{\sum_{k=1}^{length(p(i))} (p_{ik})^2} \sqrt{\sum_{k=1}^{length(p(j))} (p_{jk})^2}} \quad (1)$$

伙伴用户的对称性: 由 $Sim(i, j) = Sim(j, i)$, 可得用户 i 与用户 j 互为伙伴用户。

伙伴用户不具有传递性。

(7) 资源对用户的预期活跃度 (PA): 设用户 U 有 M 个伙伴用户, 某个公有资源对 M 个用户的 HA 向量为 $HAP = (HA_1, HA_2, \dots, HA_m)$, 伙伴用户的权重向量 $WP = (W_1, W_2, \dots, W_m)$, U 与伙伴用户的相似度向量为 $SIM = (S_1, S_2, \dots, S_m)$, 则资源对用户 U 的预期活跃度 $PA = HAP * WP$ 。因为 $W = f(S)$, 且要求 $W = f(S)$ 满足 S 越大, W 越大, 且 $SIM = (1, 1, \dots, 1)_m$ 时, $PA = \sum_1^M HA_i$, 所以可取 $f(S) = S$ 。即 $PA = HAP * WP = HAP * (W_1, W_2, \dots, W_m) = HAP * SIM$ 。对于私有资源和用户自身使用的公有资源 $PA = HA$ 。

2.2 基于 XML 的用户资源推荐模型构建

用户资源推荐模型由用户使用资源列表、模型节点关系和伙伴用户三个部分构成。

用户使用资源列表包含用户历史使用的资源, 每个历史使用资源的子节点为包含该历史资源的模型节点标识列表。使用 XML 工具构建用户历史资源使用列表如下:

```
<URs>
<R rid="" rtype="" prior="" rha="" >
<MIDs>
<MID mid="" />
...
</MIDs>
</R>
...
</URs>
```

其中, URs 表示用户使用资源列表; 资源 R 包含属性资源标识 rid 、资源类型 $rtype$ 、资源权限 $prior$ 和用户对资源的历史活跃度 rha ; MIDs 表示包含资源 R 的模型节点标识列表; MID 的属性 mid 表示包含资源 R 的模型节点的标识。

模型节点关系可以通过模型节点列表和模型节点之间的后续关系来存储。使用 XML 工具构建模型节点关系如下:

```
<Ms>
<M mid="" mtimes="" >
<RIIs>
<RI rid="" rtype="" prior="" rtimes="" />
...
</RIIs>
<FOLLOWMs>
<FOLLOWM mid="" fmtimes="" />
...
</FOLLOWMs>
</M>
...
</Ms>
```

其中, Ms 表示模型节点列表; 模型节点 M 包含属性模型节点标识 mid 和模型节点出现次数 $mtimes$; RIIs 表示模型节点 M 中出现的资源实例列表; RI 表示 M 中出现的资源实例, 它的属性包括资源标识 rid 、资源类型 $rtype$ 、资源权限 $prior$ 和资源实例 RI 在模型节点 M 中出现的次数 $rtimes$; FOLLOWMs 表示模型节点 M 的后续模型节点列表; FOLLOWM 表示模型节点 M 的后续模型节点, 它的属性包括模型节点标识 mid 和模型节点 M 到后续模型节点的度 $fmtimes$ 。

伙伴用户包含与当前用户具有伙伴用户关系的用户列表, 使用 XML 工具构建伙伴用户列表如下:

```
<CUs>
<CU userid="" sim="" >
</CU>
```

...
</CUs>
其中,CUs 表示伙伴用户列表;用户 CU 包含属性伙伴用户标识 userid,伙伴用户与当前用户的相似度 sim。

综上所述,基于 XML 的用户资源推荐模型构建如下:

```
<RRM userid=" ">  
<Ms></Ms>  
<URs></URs>  
<CUs></CUs>  
</RRM>
```

其中,RRM 表示用户资源推荐模型,它的属性 userid 表示此资源推荐模型属于某个用户。

用户定制完服务流点击保存时,将定制完成的服务流以模型节点方式组织,为每个模型节点分配一个唯一标识,计算在服务流中每个模型节点的后续模型节点集合。将该服务流的所有模型节点添加到<Ms>中,若出现包含资源实例完全相同的模型节点 M_{old} 和 M_{new} ,则保留原来的模型节点 M_{old} ,将其 mtimes 加 1,同时将服务流中所有模型节点的后续模型节点集合中的 M_{new} 的标识替换成 M_{old} 的标识。将服务流中所有模型节点的后续模型节点集合添加到对应的模型节点的子节点<FOLLOWMs>中,若在<FOLLOWMs>出现相同的后续模型节点标识,则保留 FOLLOWM_{old},将其 fmtimes 加 1。将该服务流中出现的所有资源添加到用户资源列表<URs>中,若出现相同的资源,则保留 R_{old},将其 rha 加 1,同时将该服务流中出现的包含 R_{old}的模型节点的标识添加到 R_{old}的子节点<MIDs>中,若<MIDs>存在相同的模型节点标识,则保留原来的。重新计算当前用户与其他用户的相似度获得当前用户的伙伴用户列表,将当前用户的伙伴用户列表<CUs>的子节点替换成重新计算得到的伙伴用户列表。根据以上描述,可以很容易对用户资源推荐模型进行扩展。

基于 XML 的用户资源推荐模型存储方式将各资源对用户的历史活跃度保存在资源 R 的 rha 属性中,将各模型节点之间的关系保存在模型节点列表 Ms 中,将伙伴用户保存在伙伴用户列表 CUs,使得算法在推荐资源的过程中可以直接或经过简单的计算就能从资源推荐模型中获得后续资源的历史活跃度和伙伴用户在推荐过程中所占的权重,缩短了资源推荐算法的执行时间。

3 资源推荐算法 (SRR) 描述

云服务流定制的主要任务是选择用户所需要的资源,并将这些资源组合成能实现用户需求的服务流。

文中的 SRR 算法目的在于提高用户发现所需资源的效率,算法描述如下:

算法名称:云服务流定制资源推荐 (SRR)。
输入:当前用户 U,当前选择资源 CR。
输出:数据资源显示列表 dis_D,服务资源显示列表 dis_S。

- 算法步骤如下:
- 1)检测服务器端内存是否已导入资源推荐模型。若未导入,则从外存导入 XML 格式的 N 个用户的资源推荐模型文件;
 - 2)根据当前选择资源 CR (CR 可以为空),利用资源推荐模型可分别获得公有资源对用户 U 的历史活跃度向量 PUB_HA,公有资源对伙伴用户的历史活跃度矩阵 PUB_MHA,U 的私有资源对 U 的 HA 向量 PRI_HA;
 - 3)利用 PUB_HA 向量、PUB_MHA 矩阵和 PRI_HA 向量,通过用户 U 的使用资源偏好、U 的伙伴用户和资源输入输出约束获得用户 U 的资源推荐列表 dis_D 和 dis_S;
 - 4)用户 U 从资源显示列表 dis_D 和 dis_S 中选择所需资源 CR,转步骤 2),循环直到服务流定制完成。

在算法步骤 2)中获得公有资源对 U 的历史活跃度向量 PUB_HA,公有资源对伙伴用户的历史活跃度矩阵 PUB_MHA 和 U 的私有资源对 U 的 HA 向量 PRI_HA 的方法如下:

(1)CR 为空时,取出用户 U 的推荐模型用户使用资源列表中公有资源和私有资源对用户 U 的历史活跃度,将其分别记录在 PUB_HA 和 PRI_HA 向量中。设其中私有服务资源 pri_x 个,公有服务资源 pub_x 个,私有数据资源 pri_y 个,公有数据资源 pub_y 个。

私有资源对用户 U 的 HA 向量表示为:
$$PRI_HA = (pri_SHA_1, \dots, pri_SHA_{pri_x}, pri_DHA_1, \dots, pri_DHA_{pri_y})$$

公有资源对用户 U 的 HA 向量表示为:
$$PUB_HA = (pub_SHA_1, \dots, pub_SHA_{pub_x}, pub_DHA_1, \dots, pub_DHA_{pub_y})$$

其中: pri_SHA_i —第 i ($0 < i \leq pri_x$) 个私有服务资源对用户 U 的 HA, $pri_SHA_i \geq 0$;

pri_DHA_i —第 i ($0 < i \leq pri_y$) 个私有数据资源对用户 U 的 HA, $pri_DHA_i \geq 0$;

pub_SHA_i —第 i ($0 < i \leq pub_x$) 个公有服务资源对用户 U 的 HA, $pub_SHA_i \geq 0$;

pub_DHA_i —第 i ($0 < i \leq pub_y$) 个公有数据资源对用户 U 的 HA, $pub_DHA_i \geq 0$ 。

(2)CR 不为空时,根据用户 U 的资源推荐模型,从用户使用资源列表中找到资源 CR,取出包含资源

CR 的模型节点标识列表,在模型节点列表中取出对应的模型节点,构成 CML。

若 CR 为数据资源,从 CML 中取出与 CR 在同一个模型节点的所有服务资源、数据资源,将对应的 $m_{\text{times}} * r_{\text{times}}$ 作为它们的历史资源活跃度,合并相同资源的历史活跃度,设其中私有服务资源 pri_x 个,公有服务资源 pub_x 个,私有数据资源 pri_y 个,公有数据资源 pub_y 个。

私有后续资源对用户 U 的 HA 向量表示为:

$$\text{PRI_HA} = (\text{pri_SHA}_1, \dots, \text{pri_SHA}_{\text{pri}_x}, \text{pri_DHA}_1, \dots, \text{pri_DHA}_{\text{pri}_y})$$

公有后续资源对用户 U 的 HA 向量表示为:

$$\text{PUB_HA} = (\text{pub_SHA}_1, \dots, \text{pub_SHA}_{\text{pub}_x}, \text{pub_DHA}_1, \dots, \text{pub_DHA}_{\text{pub}_y})$$

若 CR 为服务资源,从 CML 中取出与 CR 在同一个模型节点的数据资源,将对应的 $m_{\text{times}} * r_{\text{times}}$ 作为它们的历史资源活跃度,合并相同数据资源的历史活跃度,设其中私有数据资源 pri_y 个,公有数据资源 pub_y 个。从 CML 中取出包含资源 CR 的模型节点的后续模型节点列表,取出其中的服务资源,将对应的 f_{times} 作为服务资源的历史活跃度,合并相同服务资源的历史活跃度,设其中私有服务资源 pri_x 个,公有服务资源 pub_x 个。

私有后续资源对用户 U 的 HA 向量表示为:

$$\text{PRI_HA} = (\text{pri_SHA}_1, \dots, \text{pri_SHA}_{\text{pri}_x}, \text{pri_DHA}_1, \dots, \text{pri_DHA}_{\text{pri}_y})$$

公有后续资源对用户 U 的 HA 向量表示为:

$$\text{PUB_HA} = (\text{pub_SHA}_1, \dots, \text{pub_SHA}_{\text{pub}_x}, \text{pub_DHA}_1, \dots, \text{pub_DHA}_{\text{pub}_y})$$

其中: pri_SHA_i —第 i ($0 < i \leq \text{pri}_x$) 个私有后续服务资源对用户 U 的 HA, $\text{pri_SHA}_i \geq 0$;

pri_DHA_i —第 i ($0 < i \leq \text{pri}_y$) 个私有后续数据资源对用户 U 的 HA, $\text{pri_DHA}_i \geq 0$;

pub_SHA_i —第 i ($0 < i \leq \text{pub}_x$) 个公有后续服务资源对用户 U 的 HA, $\text{pub_SHA}_i \geq 0$;

pub_DHA_i —第 i ($0 < i \leq \text{pub}_y$) 个公有后续数据资源对用户 U 的 HA, $\text{pub_DHA}_i \geq 0$ 。

(3) 利用同样的方法从伙伴用户的资源推荐模型中获取公有资源对伙伴用户的 PUB_HA 形成 PUB_MHA 矩阵。

在算法步骤 3) 中获得资源推荐列表 dis_D 和 dis_S , 方法如下:

(1) 利用 PUB_HA 和 PUB_MHA , 通过 U 的使用资源偏好、U 的伙伴用户和输入输出约束获得公有资源列表 pub_D 和 pub_S ;

(2) 利用 PRI_HA , 通过 U 的使用资源偏好和输入

输出约束获得私有资源列表 pri_D 和 pri_S ;

(3) 将 pri_S 和 pub_S 合并, 并按照 PA 从大到小排序, 得到 dis_S ; 将 pri_D 和 pub_D 合并, 并按照资源 PA 从大到小排序, 得到 dis_D 。

在算法步骤 3) 中利用 PUB_MHA 和 PUB_HA 获得公有资源列表 pub_D 和 pub_S , 方法如下:

(1) 取出所有当前用户 U 的伙伴用户与当前用户的相似度 sim 构成相似度向量 SIM , 计算资源对用户 U 的 PA 向量 $\text{PUB_PA} = \text{PUB_MHA} * \text{SIM}$;

(2) 将当前用户 U 的 PUB_HA 按照 PA 进行排序, 得到对应的有序的服务资源列表 pub_S 与数据资源列表 pub_D , 将其 PA 加上 PUB_PA 中的最大值;

(3) 将 PUB_PA 中对应的服务资源和数据资源按照计算得到的 PA 排序, 得到有序的服务资源列表和数据资源列表分别添加到 pub_S 和 pub_D 后面, 并删除尾端重复项, 若 CR 为空, 转(5);

(4) 取出系统中所有满足作为 CR 后续的服务资源和数据资源, 并将其 PA 设为 0, 然后分别添加到 pub_S 和 pub_D 的尾端, 最后对整个序列删除尾端重复项;

(5) 将系统中未在 pub_S 和 pub_D 中出现资源的 PA 设为 -1, 对应添加到 pub_S 和 pub_D 的尾端。

在算法步骤 3) 中利用 PRI_HA 获得私有资源列表 pri_D 和 pri_S , 方法如下:

(1) 对用户 U 的 PRI_HA 按照 PA 从大到小排序, 得到对应的有序私有数据资源列表 pri_D 和有序私有服务资源列表 pri_S , 将其 PA 加上 PUB_PA 中的最大值。若 CR 为空, 转(3)。

(2) 先取出用户 U 的所有满足 CR 的输出格式作为其输入的服务资源, 再取出输出满足 CR 的输入的服务资源和数据资源, 最后将这些资源的 PA 设为 0, 分别添加到 pri_S 和 pri_D 的尾端, 并且对整个序列删除尾端重复项。

(3) 将系统中未在 pri_S 和 pri_D 中出现的 U 的私有资源 PA 设为 -1, 对应添加到 pri_S 和 pri_D 尾端。

SRR 算法实现资源的推荐包括对用户私有资源的推荐和海洋云平台上公有资源的推荐两方面。对私有资源的推荐依据的是资源对用户的 HA 的大小, HA 越大的私有资源在推荐资源列表中排在越前端。对公有资源的推荐综合了资源对用户的 HA 和资源对伙伴用户的 HA 计算出的 PA, 将用户自身经常使用的资源排在前端, 然后将基于伙伴用户推荐的资源排在后端。当用户选择的资源 CR 不为空时, SRR 算法将与 CR 有输入输出约束的其他资源的 PA 设为 0, 置于上述两种推荐方法得到的资源列表的尾端, 一定程度上提高了用户发现资源的效率。

4 实验及结果分析

周涛等在文献[14]中提出了平均排序分量方法,具体指的是用户选择的产品在推荐列表中所处的位置与产品总数的比值,比值越小,说明推荐效果越好。在系统中,服务流定制中个性化资源推荐算法主要解决的是用户选择服务与数据所面临的困难,换言之,该算法是用来提高用户选择数据和服务效率的。由文献[14]得到启发,文中选取用户选择需要的服务和数据时所要浏览的资源数目来衡量该算法的有效性。

实验环境构建如下:

(1)云平台搭建:硬件包括15台PC机和一台服务器。PC机均安装CentOS 5.6操作系统,其中3台为Lenovo家用机,配置为4 GB内存,Pentium(R) Dual-core CPU E5700 3.0 GHz,500 GB硬盘;12台为Dell的廉价家用机,Dell机器的配置为2 GB内存,Pentium(R) 4 3.0 GHz Dual-core CPU,80 GB硬盘;服务器作为云服务流引擎,用于接收用户提交的服务流。软件包括tomcat 6.0和Hadoop 0.2。

(2)用户与资源注册:利用海洋云平台注册100个用户,同时为每个用户分别上传50个私有数据资源和50个私有服务资源,另外在云平台上上传5 000个公有数据资源和5 000个公有服务资源。

结果分析:

系统中每个用户分别用三种不同的算法定制服务流规模从2~15的服务流140条,其中每个规模分别定制10条服务流。图3为用户U使用随机呈现、协同过滤和SRR算法定制不同规模的服务流时,浏览资源数目的整体分布,其中曲线表示定制不同规模服务流需要浏览资源数目的平均值连线。图4为在一定用户定制服务流的累积下,随着用户U定制服务流数量的增加,三种算法平均选择一个资源需要浏览资源的数目变化曲线。图5表示用户的数量与SRR算法推荐资源效率的关系。

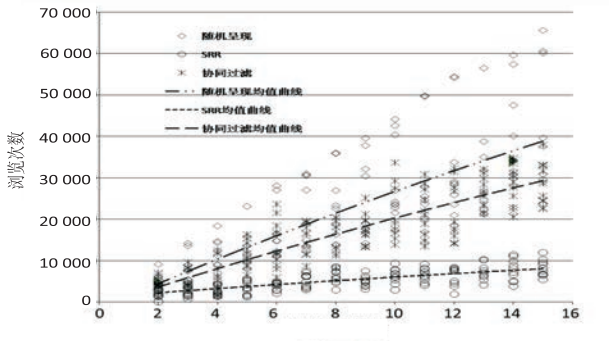


图3 用户U使用随机呈现、SRR和协同过滤算法定制服务流时,浏览资源数目分布

对于不同规模的服务流,根据图3可以得出,使用

SRR算法定制服务流要浏览资源的次数比协同过滤和随机呈现要少。从图中的曲线可以看出,SRR算法对应的曲线在协同过滤和随机呈现之下,说明SRR算法的推荐效率比协同过滤和随机呈现好。

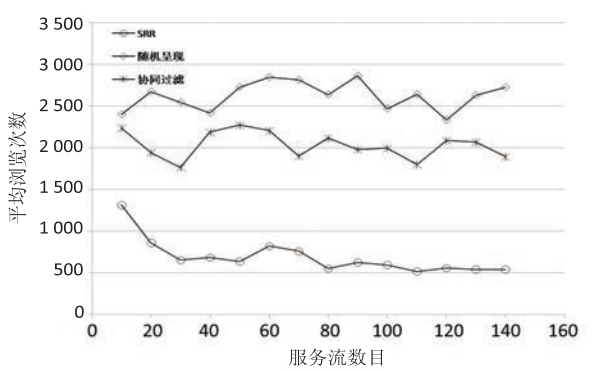


图4 用户U使用三种算法定制服务流数目与平均浏览资源数目的关系

随着用户U定制服务流数量的增加,从图4可以看出,SRR算法平均选择一个资源需要浏览资源的数目整体呈下降趋势,且SRR算法定制服务流需要浏览的平均资源数比随机呈现和协同过滤要少。

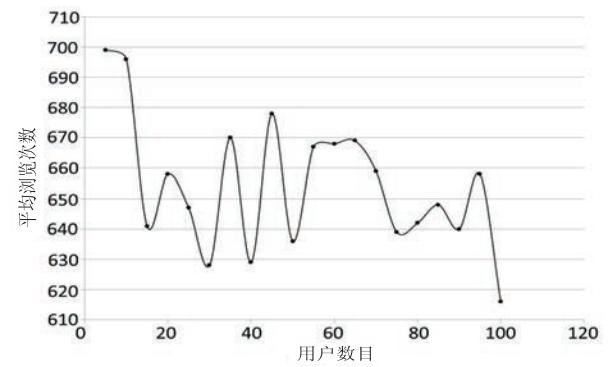


图5 用户U使用SRR算法定制服务流时用户数目与平均浏览资源数目的关系

根据图5的变化曲线可以看出用户数量的多少,对SRR算法的推荐效率影响不大。

实验结果表明,考虑了资源之间关联和伙伴用户的SRR算法比协同过滤和随机呈现算法的效果好,而且用户定制的服务流越多,SRR推荐算法效果越明显,而用户数量的多少,对算法的推荐效率影响不大。

5 结束语

在云服务流定制资源推荐过程中,服务与服务,服务与数据之间本身就存在输入输出约束,通过这种约束可以排除许多不满足约束条件的数据和服务。在用户资源推荐模型中,通过保存用户自身使用资源的习惯和基于伙伴用户的推荐,可以提高资源推荐的智能化和人性化。考虑到用户选择资源时,各资源之间可

(下转第117页)

表1 算法1和算法2在 t,r 取不同值下的实验结果比较

(t,r)	算法1	算法2
(5,10)	step = 2 932	step = 2 448
	$T = 1.23$	$T = 1.02$
(20,30)	step = 2 088	step = 2 076
	$T = 1.20$	$T = 1.14$
(40,40)	step = 984	step = 963
	$T = 0.74$	$T = 0.65$
(80,60)	step = 2 259	step = 2 149
	$T = 2.29$	$T = 2.16$
(50,100)	step = 2 288	step = 2 357
	$T = 2.36$	$T = 2.23$

4 结束语

为了提高解决分裂可行问题的算法效率,文中对所研究的修正松弛 CQ 算法^[13]的步长进行了进一步的改进。改进的修正松弛 CQ 算法不需要计算矩阵的逆和最大特征值。虽然在确定新步长时需要一些额外相对简单的计算,但可以达到提高算法效率的目的,同时给出了新的改进松弛 CQ 算法的收敛性证明。数值实验表明,新算法比原来的修正松弛 CQ 算法在解决实际问题中优越性更明显。

参考文献:

[1] Censor Y, Elfving T. A multi-projection algorithm using Bregman projections in a product space[J]. Number Algorithms, 1994, 8(2-4): 221-239.

(上接第 113 页)

能存在的隐含关联,利用用户资源推荐模型挖掘资源之间的隐含关系来进行资源推荐,在一定程度上提高了用户定制服务流的效率。文中算法的实现采用的是串行编程,旨在说明服务流定制用户资源推荐模型的有效性,下一步工作将集中在对算法的并行化改进,以达到缩短算法运行时间的目的。

参考文献:

[1] 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.

[2] Aditya S T, Dabeer O, Dey B K. A channel coding perspective of collaborative filtering[J]. IEEE Transactions on Information Theory, 2011, 57(4): 2327-2341.

[3] Gong Songjie. A collaborative filtering recommendation algorithm based on user clustering and item clustering[J]. Journal of Software, 2010, 5(7): 745-752.

[4] 黄创光,印 鉴,汪 静,等. 不确定近邻的协同过滤推荐算法[J]. 计算机学报,2010,33(8): 1369-1377.

[5] 宋真真,王 浩,杨 静. 协同过滤技术在个性化推荐中的运

[2] Byrne C. A unified treatment of some iterative algorithms in signal processing and image reconstruction[J]. Inverse Problems, 2004, 20(1): 103-120.

[3] Censor Y. Parallel application of block iterative methods in medical imaging and radiation therapy[J]. Mathematical Programming, 1988, 42(1): 307-325.

[4] Byrne C. Iterative oblique projection onto convex sets and the split feasibility problem[J]. Inverse Problems, 2002, 18(2): 441-453.

[5] Yang Qingzhi. The relaxed CQ algorithm solving the split feasibility problem[J]. Inverse Problems, 2004, 20(4): 1261-1266.

[6] Qu Biao, Xiu Naihua. A note on the CQ algorithm for the split feasibility problem[J]. Inverse Problems, 2005, 21(5): 1655-1665.

[7] 何炳生. 论求解单调变分不等式的一些投影收缩算法[J]. 计算数学, 1996(1): 97-101.

[8] 王少玲. 求解多集合分裂可行性问题的新投影算法[D]. 南京: 南京邮电大学, 2012.

[9] 王新艳, 屈 彪. 求解分裂可行问题逆问题的算法推广[J]. 泰山学院学报, 2010(6): 10-14.

[10] 张忠威. 多集合分裂可行问题的算法研究[D]. 南京: 南京邮电大学, 2011.

[11] 徐成贤, 陈志平, 李乃成. 近代优化方法[M]. 北京: 科学出版社, 2002: 18-35.

[12] 李 雷, 吴从忻. 集值分析[M]. 北京: 科学出版社, 2003: 15-36.

[13] 杨庆之, 赵金玲. 分裂可行问题(SFP)的投影算法[J]. 计算数学, 2006, 28(2): 121-132.

用[J]. 合肥工业大学学报: 自然科学版, 2008, 31(7): 1059-1062.

[6] 朱 锐, 王怀民, 冯大为. 基于偏好推荐的可信服务选择[J]. 软件学报, 2011, 22(5): 852-864.

[7] 汪英姿. 基于本体的个性化图书推荐方法研究[J]. 现代图书情报技术, 2012(12): 72-78.

[8] 陆晓敏, 崇志宏, 陈国庆. 基于本体的商品推荐方法[J]. 计算机技术与发展, 2012, 22(10): 10-14.

[9] 张 磊, 陈俊亮, 孟祥武, 等. 基于 BP 神经网络的协作过滤推荐算法[J]. 北京邮电大学学报, 2009, 32(6): 42-46.

[10] 王 宁. 一种基于 BP 神经网络的即时在线推荐系统[J]. 计算机技术与发展, 2009, 19(7): 230-233.


[11] 邓爱林, 朱杨勇, 施伯乐. 基于项目评分预测的协同过滤推荐算法[J]. 软件学报, 2003, 14(9): 1621-1628.

[12] 张少中, 陈德人. 面向个性化推荐的两层混合图模型[J]. 计算机系统应用, 2009, 18(6): 26-26.

[13] Aggarwal H C, Wolf J L, Wu Kun-Lung, et al. Horting hatches an egg: a new graph - theoretic approach to collaborative filtering [C]//Proc of the KDD 1999. [s. l.]: [s. n.], 1999: 201-212.

[14] Zhou T, Jiang L L, Su R Q, et al. Effect of initial configuration on network-based recommendation[J]. EPL, 2008, 81: 58004.

云服务流定制中的个性化资源推荐方法研究

作者：[胡彬彬](#)，[吴绍春](#)，[HU Bin-bin](#)，[WU Shao-chun](#)
作者单位：[上海大学 计算机工程与科学学院, 上海, 200444](#)
刊名：[计算机技术与发展](#)
英文刊名：[Computer Technology and Development](#)
年，卷(期)：2015(3)

引用本文格式：[胡彬彬](#).[吴绍春](#).[HU Bin-bin](#).[WU Shao-chun](#) 云服务流定制中的个性化资源推荐方法研究[期刊论文]

-[计算机技术与发展](#) 2015(3)