

基于本体的 Web 资源个性化推荐算法

梁俊杰, 刘琼妮*, 余敦辉

(湖北大学 计算机与信息工程学院, 武汉 430062)

(* 通信作者电子邮箱 joni0223@foxmail.com)

摘要:为提高 Web 资源推荐的准确度,提出基于本体的 Web 资源个性化推荐算法(BO-RM)。设计 Web 资源主题抽取算法和相似性度量方法,利用本体语义推理机制实现资源聚类,在推荐过程中通过实时分析用户浏览行为捕获用户个性化偏好的变化,动态实时推荐内容。与基于情境的协同过滤算法(CFR-RM)和基于模型的个性化预测算法(BM-RM)进行对比,结果显示 BO-RM 的时间开销相对稳定,在平均排序倒数(MRR)和平均准确率(MAP)上均取得了较好的效果。实验结果表明:BO-RM 离线完成海量 Web 资源的分析聚类,有效提高了运行效率,实用性比较强;BO-RM 实时捕捉用户兴趣变化,动态更新推荐列表,更加贴近用户的真实需求。

关键词:领域本体;资源聚类;动态推荐;个性化推荐算法

中图分类号: TP311.5 **文献标志码:** A

Personalization recommendation algorithm for Web resources based on ontology

LIANG Junjie, LIU Qiongni*, YU Dunhui

(School of Computer and Information Engineering, Hubei University, Wuhan Hubei 430062, China)

Abstract: To improve the accuracy of recommended Web resources, a personalized recommendation algorithm based on ontology, named BO-RM, was proposed. Subject extraction and similarity measurement methods were designed, and ontology semantic was used to cluster Web resources. With a user's browser tracks captured, the tendency of preferences and recommendation were adjusted dynamically. Comparison experiments with collaborative filtering algorithm based on situation named CFR-RM and personalized prediction algorithm based on model were given. The results show that BO-RM has relatively stable overhead time and good performance in Mean Reciprocal Rank (MRR) and Mean Average Precision (MAP). The results prove that BO-RM improves the efficiency by using offline data analysis for large Web resources, thus it is practical. In addition, BO-RM captures the users' interest in real-time to updates the recommendation list dynamically, which meets the real needs of users.

Key words: domain ontology; resource clustering; dynamic recommendation; personalized recommendation algorithm

0 引言

随着互联网的发展,在线获取资源已经成为人们增长知识的重要途径。当前人们处在一个大规模生产、分享和应用数据的时代——大数据时代^[1],Web 资源随互联网的发展呈爆炸式增长。人们面临的不是资源匮乏,而是资源过载,一不小心就会淹没在数据的“海洋”之中^[2]。如何快速高效地对海量资源进行分类管理,为用户按需推荐资源,是学术界的研究热点。

对于个性化推荐,当前使用最广泛的是协同过滤推荐技术^[3]。该算法通过寻找有相同偏好的几个邻居,将邻居感兴趣而用户未浏览过的资源推荐给用户。黄创光等^[4]提出一种不确定近邻的协同过滤推荐算法,结合用户以及产品的推荐结果,通过不确定近邻因子及调和参数去计算基于用户和产品的预测评分并产生推荐。王海艳等^[5]提出了一种基于可信联盟的服务推荐方法,引入服务推荐的个性属性特征,根

据改进的相似度计算方法与服务推荐行为的信任度构建出邻居用户的可信联盟。以上方法虽然改进了协同过滤算法的局限性,但协同过滤推荐是基于“用户-用户”的个性化推荐技术,过于依赖用户对资源的评分,并且由于用户兴趣的动态变化,会导致推荐结果的不准确性^[6]。郭磊等^[7]提出了一种结合推荐对象间关联关系进行推荐的算法,并通过实验证明推荐对象间的关联关系在推荐过程中起到了重要作用,是用户选择推荐结果的重要依据之一。

语义 Web 服务通过对 Web 服务进行本体语义标识,提高服务的描述能力,便于计算机对 Web 服务进行自动处理和集成。本文基于本体的语义推理优势,采用基于内容的个性化推荐技术,提出基于本体的 Web 资源个性化推荐算法。它首先通过主题抽取技术确定 Web 资源的主题,再基于领域本体将 Web 资源按主题聚类,进而通过对用户浏览行为的隐式分析,挖掘用户的兴趣点,为用户推荐最感兴趣的 Top-N 个资

收稿日期: 2014-06-05; **修回日期:** 2014-08-11。 **基金项目:** 国家自然科学基金资助项目(61273216, 61272111, 61202031, 61202032); 湖北省自然科学基金资助项目(2013CFA115, 2013CFB002); 武汉市科技攻关计划项目(201210421132, 201210621214)。

作者简介: 梁俊杰(1974-),女,湖北武汉人,副教授,博士,主要研究方向:数据分析、云计算; 刘琼妮(1990-),女,湖北武汉人,硕士研究生,主要研究方向:Web 服务发现; 余敦辉(1974-),男,湖北武汉人,副教授,博士,CCF 会员,主要研究方向:服务计算、大数据。

的有向边的数目^[11]。 F_{c1} 与 F_{c2} 是同义词时,其相似度为 1, $Dis(F_{c1}, F_{c2}) = 0$ 。

2) $|h_{c1} - h_{c2}|$ 表示 F_{c1} 与 F_{c2} 所处的层次差, h_{c1} 、 h_{c2} 为 F_{c1} 与 F_{c2} 所分别对应的层次深度。随着本体树层次深度的增加,节点的划分更为具体。因此,相同语义距离的两个概念,其相似度随所处层次之和的增加而增加,随所处层次之差的增加而减小。

3) $C(F_{c1}) \cap C(F_{c2})$ 表示 F_{c1} 与 F_{c2} 相对于本体层次树的语义重合度,是在本体层次树中描述两个概念从根节点出发所共同经过的节点在经过的所有节点中所占的比重。概念间的语义重合度越大,其相似度越高。

基于上面三个因素的考虑,则基于本体层次树计算两个概念 F_{c1} 与 F_{c2} 之间的相似度公式如式(4)所示:

$$Sim_c(F_{c1}, F_{c2}) = \frac{C(F_{c1}) \cap C(F_{c2})}{Dis(F_{c1}, F_{c2}) \times (\lambda \times |h_{c1} - h_{c2}| + 1)} \quad (4)$$

其中 λ 为调节因子。

2.2.2 基于领域本体的 Web 资源聚类

Web 资源聚类,就是将主题相同或将近的资源聚合,即通过计算资源之间的相似度,将相似度处在一个阈值内的资源聚合。根据算法 1,资源 R_i 的 n 维特征向量表示为 $t(r_i) = \{(f_{i1}, w_{i1}), (f_{i2}, w_{i2}), \dots, (f_{in}, w_{in})\}$ 。

利用式(4)计算出资源 R_1 中的每个特征词与资源 R_2 中的每个特征词之间加权的最大相似度之和,然后计算其平均值,从而获得 R_1 与 R_2 之间的相似度,具体计算公式如式(5)所示:

$$Sim'(R_1, R_2) = \frac{WS_{R_1-R_2} + WS_{R_2-R_1}}{Count(R_1) + Count(R_2)} \quad (5)$$

其中:

$$WS_{R_1-R_2} = \sum_{F_{1i} \in R_1} \max_{F_{2j} \in R_2} (a^{w_{1i}w_{2j}Sim_c(F_{1i}, F_{2j})} - 1)$$

$$WS_{R_2-R_1} = \sum_{F_{2j} \in R_2} \max_{F_{1i} \in R_1} (a^{w_{2j}w_{1i}Sim_c(F_{2j}, F_{1i})} - 1)$$

$Count(R_1)$ 与 $Count(R_2)$ 分别表示资源 R_1 与 R_2 中包含的特征词个数; $1 < a \leq 2$ 。

另外,从资源的结构来看,结构越相似,相似程度越高。从这个意义来看,进行相似性度量的两个资源所包含的特征词个数差值越小,则其相似度越大。为此,可根据两个相同的特征词在各自的资源中所处密度的影响,引入特征词密度的影响因子 χ ^[12] 如式(6)所示:

$$\chi = \theta \sqrt{\frac{|W_{id}(R_1) - W_{id}(R_2)|}{W_{id}(R_1) + W_{id}(R_2)}} \quad (6)$$

其中: θ 为可调节参数; $W_{id}(R_1)$ 、 $W_{id}(R_2)$ 表示资源 R_1 、 R_2 中含有特征词的个数。

引入特征词密度的影响因子后,两个资源 R_1 与 R_2 的相似度计算公式如式(7)所示:

$$Sim''(R_1, R_2) = \chi \times Sim'(R_1, R_2) \quad (7)$$

对计算结果作归一化处理,即让相似度的计算结果取值范围为 $[0, 1]$ 。归一化后, R_1 与 R_2 的相似度计算公式如式(8)所示:

$$Sim(R_1, R_2) = 1 - \mu_0^{Sim''(R_1, R_2)} \quad (8)$$

3 用户兴趣挖掘

对用户兴趣偏好的挖掘是个性化推荐过程中一个很重要的环节。只有准确把握用户的偏好和兴趣变化,推荐系统才能根据挖掘出的用户个性化特征找出其可能感兴趣的资源,产生推荐结果^[13]。

对用户兴趣的分析分为显式和隐式两种类型。本文主要通过隐式方式来挖掘用户兴趣。通过对系统日志的分析,考虑表 1 所示七种浏览行为,作为用户对资源 R 的兴趣度的参考标准,并基于以下假设:

- 1) Print 与 Bookmark 的影响相同。
- 2) Bookmark 比 Download 的影响稍强。
- 3) Bookmark 比 Drag 的影响强。
- 4) Bookmark 比 VisitFreq 的影响明显强。
- 5) Download 与 Copy 的影响相同。
- 6) VisitFreq 与 StayTime 的影响相同。

用层次分析法的思想计算出权向量,将兴趣度因子定量化。

表 1 用户的兴趣度因子

浏览行为	函数名	权向量	描述
Print	$pt(r)$	0.27	打印 R
Download	$dl(r)$	0.27	下载 R 中的元素或点击下载按钮
Bookmark	$bm(r)$	0.16	将 R 保存到 bookmark 或点击收藏、分享按钮
Copy	$cp(r)$	0.16	复制 R 中的文字
Drag	$dg(r)$	0.08	滚动 R 所在页面,即拖动滚动条或转动鼠标滚轮
VisitFreq	$vf(r)$	0.03	访问 R 的次数
StayTime	$st(r)$	0.03	浏览 R 的时间

用 $Like(r)$ 表示用户对 R 的兴趣度,其计算公式如式(9)所示。 $Like(r)$ 的取值范围为 $[0, 1]$, $Like(r) = 1$ 表示用户对 R 非常感兴趣, $Like(r) = 0$ 表示用户对 R 不感兴趣。

$$Like(r) = 0.27(pt(r) + dl(r)) + 0.16(bm(r) + cp(r)) + 0.08dg(r) + 0.03(vf(r) + st(r)) \quad (9)$$

用户兴趣挖掘主要是对用户的浏览行为进行实时分析^[14],根据兴趣度因子计算用户对当前 Web 资源的兴趣度。

算法 2 用户兴趣挖掘算法。

输入 Web 日志 $Log(r)$ 。

输出 $Like(r)$ 。

第 1 步 监测到用户访问 Web 资源 R , 记录时间 t_0 。

第 2 步 查询 $Like(r)$, 若 $Like(r) \geq 0.16$, 则 $vf(r) + 1$ 。

退出分析,直到用户离开 R 。

第 3 步 若在第 2 步中查询不到 $Like(r)$, 则初始化 $Like(r) = 0, vf(r) = 1$ 。

第 4 步 若 R 中包含 Office 文档或 PDF 文件, 记录其页数 P_R 。

第 5 步 获取 $Log(r)$, 监测用户是否出现 Print、Download、Bookmark、Copy 这四种行为之一。若是, 相应设置 $pt(r) = 1, dl(r) = 1, bm(r) = 1$ 或 $cp(r) = 1$, 退出分析,直到用户离开 R 。

第6步 若第5步中的行为没有出现,循环监测 Drag 行为,每间隔 ds , 动态更新 $dg(r)$ 、 $st(r)$ 的值如式(10)~(11)所示。并计算 $Like(r)$ 的值,直到用户离开 R 。

$$dg(r) = (p_R + df)/P_R \quad (10)$$

$$st(r) = \min(t - t_0 - \omega_{\min}, \omega_{\max}) \quad (11)$$

其中: p_R 为已看过 R 的页数; df 为 Drag 的次数; $t - t_0$ 描述 StayTime, 单位为 s; ω_{\min} 、 ω_{\max} 分别用来过滤用户误操作或已经离开的情况。

设定一个阈值 β , 动态计算出 $Like(r)$ 后, 将 $Like(r) \geq \beta$ 的 Web 资源 R 标记为用户感兴趣的。

4 Web 资源个性化推荐

本文提出基于本体的 Web 资源个性化推荐算法, 跟踪用户的兴趣变化, 动态地为用户推荐其感兴趣的资源。算法的工作流程分为离线、在线两个阶段。海量 Web 资源的分析聚类离线完成, 用户兴趣的实时挖掘和个性化推荐结果的动态更新在线完成。

算法3 基于本体的 Web 资源个性化推荐算法。

离线阶段:

第1步 确定业务范围, 收集资料并构建领域本体。

第2步 记录网站涵盖的所有 Web 资源, 解析为非结构化的数据格式。

第3步 按照算法1的步骤, 离线进行海量 Web 资源的主题抽取, 将资源 R 的主题映射为一个 n 维特征向量 $t(r) = \{(f_1, w_1), (f_2, w_2), \dots, (f_n, w_n)\}$ 。

第4步 依据式(8)计算资源间的相似度。

第5步 根据业务需求选定一个阈值 γ , 选取与 R_i 的相似度 $Sim(R_i, R_j) \geq \gamma$ 的资源集, 作为 R_i 的推荐备选集 BU_i 。当用户对 R_i 感兴趣时, BU_i 中的资源就是推荐结果。

第6步 Web 资源的聚类完成。间隔时间 t , 监测网站资源的更新, 重复第2~5步。

在线阶段:

第1步 用户访问网站后, 按照算法2的步骤进行用户兴趣的挖掘, 实时获取用户对当前所访问网站的兴趣度 $Like(r)$ 。

第2步 对标记为用户感兴趣的 Web 资源 R , 从数据库中提取 R 的推荐备选集 BU , 选取 Top- N 个作为推荐结果推送给用户。

第3步 用户离开网站后, 个性化推荐过程结束。

5 实验与分析

5.1 实验环境和参数设置

为了验证算法的有效性, 本文基于众知学堂在线课程平台进行了实验。平台用 Java 语言基于 SSH 框架开发, 运行在配置为 CPU: Intel Core i7 4770S 3.1 GHz 4 核, Memory: 4 GB, Disk: 1 TB, OS: 64 bit Windows 8.1 的 PC 上。实验数据从平台 2013 年 11 月至 2014 年 4 月生成的 160 个大型日志文件中采集, 共有 17019 个的 HTML 源代码和 53167 次用户会话, 选择 80% 的会话组成训练集, 20% 的会话作为测试集。根据对训练集的统计分析, 设置实验的默认参数如表 2 所示。

表2 默认参数设置及其含义

参数	默认值	含义
特征数	5	Web 资源的特征词个数
更新用户兴趣度时间	30 s	更新用户兴趣度的时间间隔
用户误操作	3 s	筛选误操作用户时间
用户离开	1800 s	筛选已离开用户时间
用户感兴趣	0.15	标记用户感兴趣的阈值
个性化推荐列表长度	5	用户感兴趣 Web 资源列表的长度

5.2 评价指标

实验假设用户点击了推荐列表的某个资源链接, 代表用户对此推荐的 Web 资源感兴趣, 算法推荐成功。使用了平均排序倒数 (Mean Reciprocal Rank, MRR)、平均准确率 (Mean Average Precision, MAP) 两个常用的评价指标作为推荐算法的评判度量。

1) MRR 的定义为:

$$MRR = \frac{1}{T} \sum_{i=1}^T \max\left(\frac{1}{loc_j}\right) \quad (12)$$

其中: T 表示测试样例的数量; $loc_j (1 \leq j \leq 5)$ 表示第 i 次推荐中, 推荐列表第 j 位置的 Web 资源首先被用户点击。首次被选中的 Web 资源位置越靠前, MRR 就越大, 推荐效果也越好。

2) MAP 的定义为:

$$MAP = \frac{1}{T} \sum_{i=1}^T \sum_{j=1}^L (P(L_j)/r_i) \quad (13)$$

其中: r_i 表示在第 i 次推荐中用户感兴趣的 Web 资源的数量; $P(L_j)$ 表示第 i 次推荐中推荐列表前 j 个位置中用户感兴趣的 Web 资源的数量与位置 j 的比值; L 是推荐列表的长度, 如表 2 所示, 取值为 5。如果推荐列表中的 Web 资源没有用户感兴趣的, 则 $MAP = 0$ 。

5.3 实验结果分析

实验选取基于情境的协同过滤算法^[15] (以下简称 CFRM)、基于模型的个性化预测算法^[16] (以下简称 BM-RM) 作为对比算法。这两种算法隐式提取用户偏好, 与本文算法 (以下简称 BO-RM) 具有可比性。

1) 运行效率对比。

分别比较 Web 资源规模和用户会话规模不同时, 算法运行时间的变化。

从图 2 可看出: 随着用户会话数增加, 待推荐列表不断增加, BM-RM 和 BO-RM 时间开销的增长相对平稳。

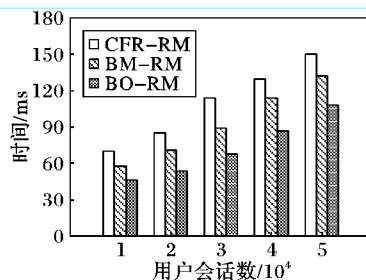


图2 不同用户会话数的时间开销

从图 3 可看出: CFRM 和 BM-RM 的时间开销受 Web 资源数量影响较为明显, BO-RM 相对比较稳定。这个结果表明, BO-RM 离线完成海量 Web 资源的分析聚类, 有效提高了运行效率, 实用性比较强。

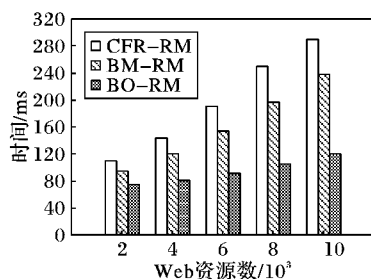


图3 不同Web资源数的时间开销

2) 推荐准确度对比。

图4~5为用户会话规模不同时,算法的MRR值和MAP值的变化。从图4~5可看出,BO-RM的平均准确率达到0.6以上,在MRR与MAP两种指标上都达到了较好的效果。这个结果表明,BO-RM实时捕捉用户兴趣变化,动态更新推荐列表,更加贴近用户的真实需求。

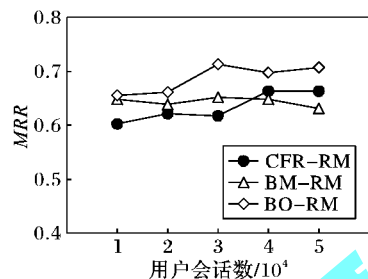


图4 算法的MRR值比较

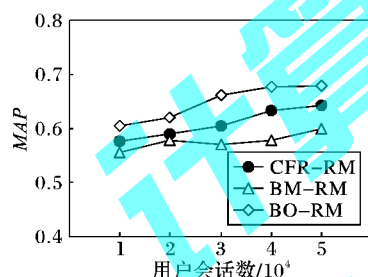


图5 算法的MAP值比较

6 结语

大数据时代带来“资源过载”问题,海量Web资源对个性化推荐算法的效率要求很高。面对海量的数据以及用户动态变化的兴趣,本文提出了基于本体的Web资源个性化推荐算法,由所构建的计算机科学的领域本体提供语义支持。根据基于领域本体的主题抽取算法和Web资源相似度计算公式,离线完成海量Web资源的分析聚类,从而提高了该算法的推荐效率,实用性比较强。用户兴趣挖掘算法实时跟踪用户的兴趣变化,动态地更新用户可能感兴趣的资源,克服了协同过滤算法中由于用户兴趣的动态变化所导致的推荐结果的不准确性,更加贴近用户的真实需求,从而有效提高个性化推荐的准确率。

参考文献:

- [1] MAYER-SCHÖNBERGER V, CUKIER K. Big data: a revolution that will transform how we live, work and think[M]. New York: Hodder Export, 2013: 3-53.
- [2] CHEN Y J, CHU H C, CHEN Y M, et al. Adapting domain ontology for personalized knowledge search and recommendation[J]. Information and Management, 2013, 50(6): 285-303.
- [3] MA H, ZHANG G, LI P. Survey of collaborative filtering algorithms

- [J]. Journal of Chinese Computer Systems, 2009, 30(7): 1282-1288. (马宏伟, 张光卫, 李鹏. 协同过滤推荐算法综述[J]. 小型微型计算机系统, 2009, 30(7): 1282-1288.)
- [4] HUANG C, YIN J, WANG J, et al. Uncertain neighbors' collaborative filtering recommendation algorithm[J]. Chinese Journal of Computers, 2010, 33(8): 1369-1377. (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] WANG H, YANG W, WANG S, et al. A service recommendation method based on trustworthy community[J]. Chinese Journal of Computers, 2014, 37(2): 301-311. (王海艳, 杨文彬, 王随昌, 等. 基于可信联盟的服务推荐方法[J]. 计算机学报, 2014, 37(2): 301-311.)
- [6] ZHANG X, LIU C, WANG L, et al. Trustworthy Web service recommendation based on collaborative filtering[J]. Journal of Computer Applications, 2014, 34(1): 213-217. (张璇, 刘聪, 王黎霞, 等. 基于协同过滤的可信Web服务推荐方法[J]. 计算机应用, 2014, 34(1): 213-217.)
- [7] GUO L, MA J, CHEN Z, et al. Incorporating item relations for social recommendation[J]. Chinese Journal of Computers, 2014, 37(1): 219-228. (郭磊, 马军, 陈竹敏, 等. 一种结合推荐对象间关联关系的社会化推荐算法[J]. 计算机学报, 2014, 37(1): 219-228.)
- [8] ZHONG X, LIU Z, DING P. Construction of knowledge base on hybrid reasoning and its application[J]. Chinese Journal of Computers, 2012, 35(4): 761-766. (钟秀琴, 刘忠, 丁盘苹. 基于混合推理的知识库的构建及其应用研究[J]. 计算机学报, 2012, 35(4): 761-766.)
- [9] OU J, LEE C, CHEN M. Efficient algorithms for incremental Web log mining with dynamic thresholds[J]. The VLDB Journal, 2008, 17(4): 827-845.
- [10] GUO Y, BAI S, YANG Z, et al. Analyzing scale of Web logs and mining users' interests[J]. Chinese Journal of Computers, 2005, 28(9): 1483-1496. (郭岩, 白硕, 杨志峰, 等. 网络日志规模分析和用户兴趣挖掘[J]. 计算机学报, 2005, 28(9): 1483-1496.)
- [11] WU J, WU Z, LI Y, et al. Web service discovery based on ontology and similarity of words[J]. Chinese Journal of Computers, 2005, 28(4): 595-602. (吴健, 吴朝晖, 李莹, 等. 基于本体论和词汇语义相似度的Web服务发现[J]. 计算机学报, 2005, 28(4): 595-602.)
- [12] YU D, HE K, LI B. Development of domain problem ontology based on model clustering algorithm[J]. Journal of Chinese Computer Systems, 2013, 34(1): 29-33. (余敦辉, 何克清, 李兵. 基于模型聚类算法的领域问题本体构建[J]. 小型微型计算机系统, 2013, 34(1): 29-33.)
- [13] HE Y, MA L, TENG G. Web log mining based on user's accessing interest[J]. Systems Engineering—Theory and Practice, 2012, 32(6): 1353-1361. (何跃, 马丽霞, 腾格尔. 基于用户访问兴趣的Web日志挖掘[J]. 系统工程理论与实践, 2012, 32(6): 1353-1361.)
- [14] CHAN N, GAALLOUL W, TATA S. A recommender system based on historical usage data for Web service discovery[J]. Service Oriented Computing and Applications, 2012, 6(1): 51-63.
- [15] SU J, YE H, YU P, et al. Music recommendation using content and context information mining[J]. IEEE Intelligent Systems, 2010, 25(1): 16-26.
- [16] PALMISANO C, TUZILIN A, GORGOLIONE M. Using context to improve predictive modeling of customers in personalization applications[J]. IEEE Transactions on Knowledge and Data Engineering, 2008, 20(11): 1535-1549.