

一种基于本体语义相似度的协同过滤推荐方法

吴正洋 汤庸 方家轩 董浩业
(华南师范大学计算机学院 广州 510631)

摘要 协同过滤推荐是一种基于用户偏好的个性化推荐方法,一般包含两个步骤:首先根据用户或项目的标注信息计算出用户或项目的相似度,确定邻居集合;然后根据相似度进行排序推荐,其核心问题在于相似度的计算。为了更好地达到这一目的,近年来关于将用户社交网络信息融入相似度计算的方法受到广泛关注。用户的注册信息、项目评分和社交信息都可以作为用户比较的依据。基于此提出了通过构建用户本体,计算本体之间的语义相似度,从而找到相似用户集合,最终实现目标用户的推荐方法。该方法为本体技术与推荐系统的结合提供了一种思路,实验表明它能够一定程度上提高推荐的准确度。

关键词 推荐系统,协同过滤,本体,语义相似度

中图分类号 TP311 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.9.039

Collaborative Filtering Recommendation Algorithm Based on Ontology Semantic Similarity

WU Zheng-yang TANG Yong FANG Jia-xuan DONG Hao-ye
(School of Computer, South Normal University of China, Guangzhou 510631, China)

Abstract Collaborative filtering recommendation is a personalized recommendation method based on users' preferences. It includes two steps. Firstly, according to the information marked by user or project, the similarity of the users or projects is calculated and the neighbor set is determined. Secondly, sorting the similarity, user or project is recommended. During those process, similarity calculation is the core problem. In recent years, the method which uses users' social network information to calculate the similarity has gotten widely attention. Users' registration information, the project score information, and social information can be used as a basis for comparing. Based on those, we built the ontology of users, calculated the semantic similarity between the ontologies, and then found a similar set of users. Through this method, we accomplished the purpose of personalized service. This method provides an idea to combine ontology technology and recommendation system. Experiments show that this method can improve the accuracy of recommendation.

Keywords Recommendation systems, Collaborative filtering, Ontology, Semantic similarity

1 引言

“协同过滤(Collaborative Filtering, CF)”思想由 Glodberg 等人于 1992 年首次提出,极大地推动了推荐系统的研究和发展。尤其随着电子商务的广泛应用,推荐系统更显示出重要作用。随着大数据时代的来临,互联网上信息量剧增,用户面临着信息过载的尴尬,因此,个性化、定制式的信息服务逐渐成为 Web 应用技术的热点。推荐系统在新闻、电影、书籍、微博等的个性化推荐中也取得了不同程度的成功。协同过滤推荐方法虽然已得到广泛使用,但存在的数据稀疏性、冷启动等问题,会对推荐质量产生一定的影响。针对以上问题,已有相关研究提出了不同的解决方法:基于 K 近邻模型的方法、通过矩阵分解技术填补用户评分的方法、基于图论的协同过滤方法,以及基于概率推理的方法,这些方法都可在

一定程度上提高推荐准确度。就协同过滤推荐算法而言,相似度计算问题是核心问题,相关的研究较多,研究人员利用语义的优点,提出了基于项目语义相似度的协同过滤推荐算法^[1],这种算法为了解决用户-项目评分矩阵稀疏的问题,利用领域本体计算评分项目之间的语义相似度,之后根据用户已评分的项目预测未评分项目的用户评分,填充用户-项目评分矩阵。该算法在一定程度上解决了数据稀疏问题,即补充了项目评分,但未较理想地解决用户相似度的计算问题。文献^[2]提出一种融合个人本体的协同过滤推荐系统,该系统运用了本体论技术,其主要思想是利用用户-项目评分矩阵构建本体,再通过把目标用户与本体比较,将最相似本体对应的偏好实例(论文)推荐给目标用户,但在用户与本体比较时,所使用的方法是将本体转化为文本后再与用户对应的信息文本做相似度对比,效果不太理想。

到稿日期:2015-03-11 返修日期:2015-05-27 本文受国家 863 计划重大项目(2013AA01A212),国家自然科学基金项目(61272067),广东省重大科技专项计划项目(2012A080104019)资助。

吴正洋(1980-),男,博士生,高级工程师,主要研究方向为本体论工程、综合评价, E-mail: wukeking@163.com; 汤庸(1964-),男,博士,教授,博士生导师,主要研究方向为大数据与社交网络、云计算, E-mail: ytang4@qq.com; 方家轩(1992-),男,硕士生,主要研究方向为社交网络、协同软件; 董浩业(1989-),男,硕士生,主要研究方向为社交网络、数据挖掘。

近年来,随着电子商务、社交网络的发展和融合,用户的数量和信量均呈指数级增长,用户通过网络途径的联系也日趋紧密,而且根据其自身特点和在网络中的角色行为逐渐形成了网络社区、虚拟团队^[3]等小群体。有学者提出一种结合社交与标签信息的协同过滤推荐算法^[4],该方法提出了小众重叠度和个体重要度两个概念,描述了“个体-小众-社区”的形成过程,再通过分析“用户-项目-标签”三元组信息获得用户间的相似度,并结合社区中的个体重要度,最终得到目标用户的偏好预测和个性化推荐。该方法通过将用户评分的项目、用户的 SNS(Social Networking Services)标签作为因子计算用户相似度,进一步提高了用户比较的准确性。文献^[5]提出了基于社交网络信息的协同过滤推荐算法,在用户相似度计算时,融合了社交网络体现的用户熟练度和对系统项目评分的相似度。文献^[6]也将目标用户在社交网络中的直接好友的项目评分作为其预测项目评分的考虑因素,实验表明该方法对缓解数据稀疏性具有较好效果。

本文第 2 节对本体语义相似度及计算模式做了介绍;第 3 节首先描述了用户本体构建,然后描述了本体语义相似度算法及推荐算法;第 4 节在真实数据集上验证了基于本体语义相似度的推荐算法,并对实验结果进行了分析;最后总结全文,并描述进一步的研究工作。

2 本体语义相似度

语义相似度是文本处理^[7]、数据挖掘等研究领域的基本问题。语义相似度计算^[8]已经被应用于词义消歧^[9]、网络信息提取^[10]、文本分类和聚类等。语义相似度的计算方法是语义相似度研究的核心问题。

2.1 语义相似度计算模式

最早的关于语义相似度的计算源于拉达等研究人员于 1989 年提出的边缘计数法^[11],这种方法的原理是:语言网络中的两个单词的语义相似度是由这两个单词所对应节点之间的最短路径决定的,两个节点之间的边越多,则对应的单词相似度越低,反之则相似度越高。随着研究的发展,目前通常把语义相似度的计算分为 3 类模式^[12],即基于结构图的模式、基于知识的模式,以及分布式语义模式^[13]。

(1)基于结构图的模式。基于结构图的语义相似度计算模式是指以语义描述的层次结构,即将对象之间的图形关系作为计算因子,考虑距离、层级数等因素。双方在可以转换成图的数据结构中进行定义和概念的比较,一般是在同一个分类或语义图中。例如,先定义“基团”的概念,再对这些“基团”中的 RDF(S)分支进行相互比较。主要通过“IS_A”表示上下级关系的轻量级本体通常可以使用这种模式。

(2)基于知识的模式。这类模式通常用来比较概念或实例,例如基于描述逻辑的更复杂的表达式定义。这种比较模式一般是在元素中定义的信息(概念或实例)不能减少到一个图形时使用,通常用于由 OWL 表达的“重量级”本体。需要注意的是基于图的语义的措施(见上文)也可以在“重量级”本体中使用,但首先要基于逻辑推理将“重量级”本体划分为“轻量级”本体。

基于知识的模式还可以进一步细分,即分为基于属性和基于概念两种。基于属性的语义相似度计算方法以对象具有的同属性的数量作为计算因子;基于概念的语义相似度计算方法则以两个概念共享的信息量作为计算因子。

(3)分布式语义模式。分布式语义模式是一种通过分析原始的文本进行语义相似度评估的模式,可用于评估术语、句子或者文档的语义相似度。这种模式是基于分布式假设语言学^[14]的,分布式假设语言学是根据语言使用情况进行语义推导的一种学说理论,其中心思想是:如果单词的使用和发生是在同一种环境中,这时的单位往往具有类似的含义。分布式语义模式是从一个大的文本语料库(如 WordNet)^[15]获取统计数据来确定两个词上下文的相似程度,这种语义相似度的计算因子对于在类似语义环境中所找到的单词是趋向于语义相似的。

2.2 本体的语义相似度计算

关于本体的语义相似度计算有多种方法^[16],其中比较常用的是基于属性的语义相似度计算模式,该模式的基本思想是:事物由其属性特征反映其本身,人们用以辨识或区分该事物的标志就是属性特征,事物之间的关联程度与其所具有的公共属性数相关。对于两个被比较概念词而言,公共属性项越多,相似度越大。Tversky 算法是该类算法的典型,但该算法只利用了相应本体的属性集信息,既没有考虑被比较概念词在分类体系树中的位置信息,也没有考虑其祖先概念节点和自身所包含的信息内容。

Banerjee、Pedersen 以及 Patwardhan 等人提出的基于概念注释的方法的思想是概念相似度或相关度值通过两个概念在本体中的注释(gloss)重叠程度来获得。由于注释的个数可能很少,因此以上办法是将注释的个数扩展至与目标概念直接相连的概念或目标概念上下文概念注释相重合的程度。这不仅涉及上下位关系,也涉及其它关系,在注释集中提取共同属性或判断属性的相似度。

3 基于本体语义相似度的协同过滤推荐系统

由于本体具有较强的语义表达能力,能够描述比较简单的抽象概念,因此本文提出利用语义本体作为反映用户属性关系的模型,并且利用本体的语义相似度计算方法,计算出相似的用户。本节第一部分描述了用户本体构建的过程;第二部分描述了用户本体相似度的计算方法;第三部分描述了基于本体的协同过滤推荐算法(BOCFR)。

3.1 用户本体的构建

构建用户本体首先需要获取用户的相关属性,系统用户的属性文档可以分为 3 类:第一类是用户注册时所提供的个人信息集文档,这些信息可以体现用户的基础属性,并能够为用户本体构建提供基础层次逻辑结构;第二类是用户的社交信息集文档,这些信息能够体现系统用户交流习惯、所属虚拟社区等特点,也可以作为构建本体的知识来源;第三类是用户的项目偏好集信息文档,如果某个用户对某类型的事物感兴趣,通常是对该类事物的某个具体项目方面感兴趣,则该用户会对他感兴趣的这个项目进行评分,可以通过用户项目评分矩阵来获取这些信息。结合以上 3 方面的知识来构建系统用户本体,便能够比较全面地反映系统中该用户的基本属性、主要特点、兴趣喜好等逻辑知识。用户本体的构建根据文献^[2]中所提出的基于 FCA(Formal Concept Analysis)的方法进行,其步骤如下:

(1)规范化关键字

首先从文档数据库中提取所选择文档的关键字集合。然后,匹配所有的文件与关键字集合。某文件如果包括特定关

键字则将被标记为“1”，以这种方式形成正式的文件和关键字的上下文关系。由于这项研究中的得分机制，关键字集合将根据用户的权重进行测序。在后面部分所提到的用户偏好将被转换成树型框架的一个部分，其出现的频率越高，越处于较高的层次结构。根据 FCA 的定义，首先将形式概念背景定义为 K ，文档集合定义为 E ，关键字集合定义为 T ，文档与关键字之间的关系定义为 R 。那么它们的关系可以定义为： $K: = \langle E, T, R \rangle$ 。

(2) 提取所有来自文档的概念 C_{doc}

定义 A 作为 E 的集合，以及 B 作为 T 的集合。那么， $A \subseteq E, B \subseteq T$ 。如果某组概念是 $A \subseteq B$ 的，那么将其记为概念 $c(A, B)$ 。对于某一个概念 $c(A, B)$ ，如果 A 与 B 中所有的关联 R 能够形成一个最大化的矩阵，那么所有概念 c 的集合记为 C_{doc} 。

(3) 基于概念 C_{doc} 生成概念格

如果所有包含关键字 B_1 的文档的集合均包括在包含了关键字 B_2 的文件集合中，则关键字 B_1 被标记为关键字 B_2 的子概念。也就是说，对于所有概念 C_{doc} ，如果 $B_1 \subseteq B_2$ ，那么 $c_1(A_1, B_1)$ 就是 $c_2(A_2, B_2)$ 的子概念，并且表示为 $(A_1, B_1) \leq (A_2, B_2)$ ，符号“ \leq ”代表概念的层次结构。

(4) 将概念格映射为本体

将概念格映射为本体的思路是：直接将概念格中概念节点的最底元素从概念节点中移走，并为每个移走下一级元素的概念节点添加子概念节点。

设元组 $F = (c, r)$ 表示一个图形结构的概念格，其中每个节点 $c \in C_d$ 表示一个在概念集合 C_d 中的概念， R_d 是概念格中的一个偏序关系集， $r \in R_d$ 表示概念节点之间的关系。设元组 $O = (C_{onto}, R_{onto})$ 表示一个本体， $Traversal(M_d)$ 是遍历概念格无向图并求层数的函数， $Del(L)$ 为删除第 L 层概念节点的函数， $Add(n, c)$ 为添加概念 c 到第 n 层概念节点的函数，算法如下：

$O = Del(Traversal(F))$ and $Add(Traversal(F) - 1, c)$

其中， C_{onto} 是所映射本体的概念集， $C_{onto} \subseteq C_d$ ， R_{onto} 是本体中概念间的关系。

3.2 本体语义相似度计算

根据上一小节提出的方法，可以构建出一系列的系统用户本体，构成用户本体的概念词或专业术语在表述意义上基本是一致的，但在数量和概念结构方面存在差异。本文拟通过比较本体所包含的属性概念词及结构，综合计算用户本体之间的语义相似度，找到与目标用户本体相似的用户本体集。

3.2.1 属性相似度计算

基于属性的语义相似度计算原则：事物之间的关联程度与其所具有的公共属性数相关，对于两个被比较概念词而言，公共属性项越多，相似度越大。基于此原则，两本体的属性相似度计算表示为：

$$Sim_{property}(O_1, O_2) = \frac{|P_1 \cap P_2|}{|P_1 \cap P_2| + \theta |P_1 - P_2| + (1 - \theta) |P_2 - P_1|} \quad (1)$$

其中， $0 < \theta < 1$ ， P_1 和 P_2 分别表示本体 O_1 和 O_2 的属性集合。 $|P_1 \cap P_2|$ 表示集合 P_1 和 P_2 的交集的元素个数， $|P_1 - P_2|$ 表示属于集合 P_1 而不属于集合 P_2 的元素的个数， $|P_2 - P_1|$ 表示属于集合 P_2 而不属于集合 P_1 的元素的个数。 θ 的计算方法如下：

$$\theta = \frac{\text{Max}\{\text{depth}(O_1), \text{depth}(O_2)\}}{\text{depth}(O_1) + \text{depth}(O_2)}$$

式中， $\text{depth}(O_i)$ 表示本体的所有概念路径的最大值。

3.2.2 结构相似度计算

本文根据文献[17]所提出的利用公共子序列(或子串)对概念相似度计算的相关研究，提出利用公共子序列元素进行本体的结构相似度计算。公共子序列，即两个本体中共有的概念序列，在公共子序列中不仅概念节点是一致的，而且概念节点之间的关联顺序也是一致的，在一定程度上反映出了本体的相似性。利用公共子序列计算结构相似度，可以根据其数量、层级、长度 3 个因子分别计算，再进行综合汇总。

(1) 公共子序列数量

在两个本体中，相同属性所处的位置不一定相同，仅仅计算相同属性的数量还不足以精确地表现出两个本体的相似度。在多个本体中，拥有更多公共子序列的本体，具有更高的相似度。设 $CS(O_1, O_2)$ 表示两个本体的公共子序列，则：

$$CS(O_1, O_2) = \{ \langle C, \leq \rangle \mid C \in \{ O_1 \cap O_2 \} \}$$

定义 $CSnum(O_1, O_2)$ 为公共子序列的数量因子，其计算函数如下：

$$CSnum(O_1, O_2) = \frac{2 \times NCS(O_1, O_2)}{S(O_1) + S(O_2)} \quad (2)$$

其中， $S(O_i)$ 表示本体 O_i 的子序列数量。 $NCS(O_1, O_2)$ 表示两个本体的公共子序列数量。

(2) 公共子序列层级

认为在公共属性的层次结构中，低层级部分概念应该比高层级的概念具有更高的相似性。因此，如果公共子序列在待比较的本体中所处的位置越低，则待比较的本体之间的相似度越高。通过从子序列的深度 D_c 中减去最小公共子序列的深度，得到公共子序列层级系数 $CS_{pec}(O_1, O_2)$ 。

$$CS_{pec}(O_1, O_2) = D_c - \text{depth}(LCS(O_1, O_2)) \quad (3)$$

其中， $\text{depth}(c) = |path(c, root)|$ ，即概念 c 到本体根节点的距离； $LCS(O_1, O_2)$ 表示最小公共子序列。

(3) 公共子序列长度

两个子序列共享的信息越多，则它们越相似。即两个本体中，若所具有的公共子序列的长度越长，则其相似度越高。定义最长公共子序列的长度因子的计算函数如下：

$$CS_{len}(O_1, O_2) = \text{Max}(\text{length}(CS(O_1, O_2))) \quad (4)$$

3.2.3 相似度综合计算

较多文献对相似度的综合计算方法做了阐述，文献[18]提出的计算综合相似度的公式如下：

$$Sim(O_1, O_2) = \sum (W_i \times Sim_i(O_1, O_2)), W_i \in (0, 1) \quad (5)$$

另有文献提出综合相似度的计算公式为：

$$Sim(O_1, O_2) = \prod (Sim_i(O_1, O_2)^{W_i}), W_i \in (0, 1) \quad (6)$$

结合式(2)一式(4)，考虑公共子序列因子之间的相关程度，按文献[17]的思路，根据式(6)得到基于公共子序列因子的本体相似度度量定义如下：

$$Sim_{CS}(O_1, O_2) = \log((CSnum(O_1, O_2) - 1)^a \times (CS_{pec}(O_1, O_2)^\beta + k) \times (CS_{len}(O_1, O_2)^\xi + l)) \quad (7)$$

式(7)用 \log 函数表示公共子序列数量相似度计算结果，利用了 \log 函数的形态特点。当 $CSnum(O_1, O_2) = 1$ 时，两个本体的相似度最大。 $a > 0, \beta > 0, \xi > 0$ 是各因子的权值， k, l 是一个常量。为确保该函数是正数且组合式是非线性的， k, l 必须大于或等于 1。

结合式(1)计算得到的相似度 $Sim_{property}(O_1, O_2)$, 通过式(7)计算得到的相似度 $Sim_{CS}(O_1, O_2)$, 再根据式(5)将其综合计算, 得到用户本体的相似度计算函数:

$$Sim(O_1, O_2) = f_1 \times Sim_{property}(O_1, O_2) + f_2 \times Sim_{CS}(O_1, O_2) \quad (8)$$

其中, f_1 、 f_2 是两类相似度所占的权值。

3.2.4 相似度初始权值的设定

本文采用 sigmoid 函数来设定各相似度计算函数的初始权值。sigmoid 函数^[19]是一种常用的阈值函数, 具有光滑、单调性、对称性等很多良好的性质。很多自然过程包括复杂系统的学习曲线都呈现出 sigmoid 函数的特征, 其函数形式如下:

$$f(x) = \frac{1 - e^{-\varphi x}}{1 + e^{-\varphi x}} = \frac{2}{1 + e^{-\varphi x}} - 1, \varphi > 0 \quad (9)$$

sigmoid 函数具有较小的初始值, 加速增长, 加速度逐渐减小, 最后逐渐稳定, 其形态比较符合人类的情感分布。其中, φ 是根据具体应用而设定的常量, x 表示由各因子计算得到的相似度值, $0 < x < 1$, $f(x)$ 为各个方法相应的初始权值。式(8)中两类相似度所占的权值 f_1 、 f_2 便可以使用 sigmoid 函数获得。

3.3 基于本体的协同过滤推荐算法(BOCFR)描述

本文提出的基于本体的协同过滤推荐算法(Collaborative Filtering Recommendation Algorithm Based on Ontology, BOCFR)使用语义相似度计算模式计算用户的相似度。首先利用社交网络、项目评分等信息构建用户本体, 本体的构建采用了形式概念分析的方法, 即先根据属性关系生成概念格, 再进一步生成语义本体, 每个用户对应其本体的实例; 其次将目标用户本体与其他用户本体进行语义相似度计算, 根据计算结果获取推荐用户本体集, 及进一步获取其实例集, 即推荐用户集; 最后根据推荐用户的相似度和偏好评分实现推荐, 推荐算法如下:

$$\bar{P}_{u,i} = \bar{r}_u + \frac{\sum_{v=1}^n sim_{u,v}(r_{v,i} - r_v)}{\sum_{v=1}^n ||sim_{u,v}||} \quad (10)$$

设 $\bar{P}_{u,i}$ 表示用户 u 对项目 i 的预测评分值, \bar{r}_u 表示用户 u 对所有项目评分的平均值, $sim_{u,v}$ 表示用户 u 和用户 v 的相似度, $r_{v,i}$ 表示用户 v 对项目 i 的评分值, r_v 表示用户 v 对所有项目评分的平均值。

BOCFR 的实现过程如下:

输入: 用户基本信息集、用户社交信息集、项目-评分信息集

输出: 目标用户 u 项目预测评分 $\bar{P}_{u,i}$

- (1) 用 FCA 本体工具, 基于输入信息生成用户本体 O_i 。
- (2) 用式(1)计算属性相似度 $Sim_{property}$ 。
- (3) 用式(2)~式(4)分别计算出公共子序列因子 CSnum、CSpec、CSlen。
- (4) 用式(7)计算结构相似度 Sim_{CS} , 并结合属性相似度 $Sim_{property}$, 按式(8)得到用户本体的最终相似度值 Sim , 其权值 f_1 、 f_2 由式(9)确定。
- (5) 用式(10)计算目标用户 u 对项目 i 的预测评分值 $\bar{P}_{u,i}$ 。

4 实验结果及分析

4.1 实验数据集

利用本文提出的 BOCFR 方法在 MovieLens 数据集上进

行了验证实验。在推荐系统研究领域中, MovieLens 数据集应用得比较广泛。该数据集包含了 943 个用户对 1682 部电影的真实评分, 标注达到 100000 个, 评分采用五分制。我们分别用 314、628 和 943 个用户, 构建 3 种规模的评价矩阵, 每个评价矩阵留出 80% 的评价信息作为训练集, 20% 作为测试集。

4.2 衡量标准

本文采用通常使用的平均绝对偏差(Mean Absolute Error, MAE)作为准确性度量标准。MAE 通过计算预测的用户评分与实际的用户评分之间的偏差(即用户对资源评分的实际值与预测值之差的绝对值的加权平均值)来度量预测的准确性, MAE 越小, 推荐结果的准确度越高。MAE 计算公式如下:

$$MAE = \frac{1}{n} \sum_{u,i} |r_{u,i} - \hat{r}_{u,i}|$$

其中, 算法预测结果为 $\hat{r}_{u,i}$, 而对应测试集中实际项目集合为 $r_{u,i}$, n 表示集合大小。

4.3 实验结果分析

将通过本文方法生成的推荐用户集与传统的基于用户的协同过滤推荐算法(UCF)、结合社交与标签信息的协同过滤推荐算法(CFR-ST)^[4], 以及文献[20]提出的改进相似性度量的协同过滤推荐算法(Improve-Sim)所生成的推荐用户集进行比较。结果如图 1 所示。

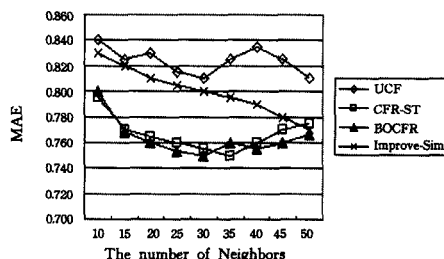


图 1 几种推荐方法下的 MAE 值

由图 1 可知, 本文所提出的基于本体语义相似度的协同过滤推荐算法(BOCFR)与其他算法相比, 推荐效果更好, 可以有效提高推荐质量。

结束语 随着互联网应用的发展与社交网络的普及, 将更多的网络信息运用到个性化推荐中是推荐方法研究的发展趋势, 而如何将多来源的信息融汇贯通地利用起来是研究的重点。本文通过构建用户本体, 利用本体自身的语义特点, 将多源信息组合成为一个能够较全面反映用户特征的有机整体, 再计算用户本体的相似度从而得到相似用户, 这在一定程度上提高了推荐质量, 为 Web2.0 下的个性化推荐研究提供了一个较好的解决方案, 并对利用本体演化进行推荐的推理预测提供了参考。

参考文献

- [1] Juan B. Collaborative filtering recommendation algorithm based on semantic similarity of item[C]// 2012 IEEE Fifth International Conference on Advanced Computational Intelligence(ICA-CI). IEEE, 2012: 452-454
- [2] Chen D, Chiang Y. Combining Personal Ontology and Collaborative Filtering to Design a Document Recommendation System [J]. Journal of Service Science & Management, 2009, 2(4): 322-328

(下转第 225 页)

- puters & Industrial Engineering, 2005, 48(2): 409-425
- [12] Zhang G, Shao X, Li P, et al. An effective hybrid particle swarm optimization algorithm for multi-objective flexible job-shop scheduling problem [J]. Computers & Industrial Engineering, 2009, 56(4): 1309-1318
- [13] Xing L, Chen Y, Yang K. An efficient search method for multi-objective flexible job shop scheduling problems [J]. Journal of Intelligent Manufacturing, 2009, 20(3): 283-293
- [14] Kacem I, Hammadi S, Borne P. Pareto-optimality approach for flexible job-shop scheduling problems [J]. Hybridization of evolutionary algorithms and fuzzy logic Mathematics and Computers in Simulation, 2002, 60(3): 245-276
- [15] Wang X, Gao L, Zhang C, et al. A multi-objective genetic algorithm based on immune and entropy principle for flexible job-shop scheduling problem [J]. The International Journal of Advanced Manufacturing Technology, 2010, 51(5): 757-767
- [16] Li J, Pan Q, Xie S. An effective shuffled frog-leaping algorithm for multi-objective flexible job shop scheduling problems [J]. Applied Mathematics and Computation, 2012, 218(18): 9353-9371
- [17] Chiang T, Lin H. Flexible job shop scheduling using a multi-objective memetic algorithm [M]// Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence. Springer Berlin Heidelberg, 2012: 49-56
- [18] Chiang T, Lin H. A simple and effective evolutionary algorithm for multiobjective flexible job shop scheduling [J]. International Journal of Production Economics, 2013, 141(1): 87-98
- [19] Gong M G, Jiao L C, Du H, et al. Multi-objective immune algorithm with nondominated neighbor-based selection [J]. Evolutionary Computation, 2008, 16(2): 225-255
- [20] Cheng R. A tutorial survey of job-shop scheduling problems using genetic algorithms, part I: Representation [J]. Computers & Industrial Engineering, 1996, 30(4): 983-997
- [21] Balas E. Machine scheduling via disjunctive graphs; an implicit enumeration algorithm [J]. Operations Research, 1969, 17(6): 941-957
- [22] Zitzler E, Thiele E, Laumanns M, et al. Performance assessment of multiobjective optimizers; An analysis and review [J]. IEEE Transactions on Evolutionary Computation, 2003, 7(2): 117-132

(上接第 207 页)

- [3] 吴正洋, 汤庸, 洪少文. 社交网络中虚拟团队的语义描述及本体构建[J]. 计算机应用与软件, 2014(11): 6-10
Wu Zheng-yang, Tang Yong, Hong Shao-wen. Semantic Description and Ontology Construction for Virtual Teams in Social Networking[J]. Computer Applications and Software, 2014(11): 6-10
- [4] 于洪, 李俊华. 结合社交与标签信息的协同过滤推荐算法[J]. 小型微型计算机系统, 2013, 34(11): 2467-2471
Yu Hong, Li Jun-hua. Collaborative Filtering Recommendation Algorithm Using Social and Tag Information[J]. Journal of Chinese Computer Systems, 2013, 34(11): 2467-2471
- [5] 陈孝文. 基于社交网络的协同过滤推荐算法研究[D]. 广州: 华南理工大学, 2013
Chen Xiao-wen. Research of Collaborative Filtering Recommendation Algorithm Base On Social Network [D]. Guangzhou: South China University of Technology, 2013
- [6] 贺超波, 汤庸, 傅成州. 融合社交网络信息的协同过滤方法[J]. 暨南大学学报(自然科学版), 2013, 34(3): 243-252
He Chao-bo, Tang Yong, Fu Cheng-zhou. Collaborative filtering method with social network information[J]. Journal of Jinan University(Natural Science), 2013, 34(3): 243-252
- [7] Li Y, Bandar Z A, McLean D. An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources[J]. IEEE Transactions on Knowledge and Data Engineering, 2003, 15(4): 871-882
- [8] Budanitsky A, Hirst G. Evaluating wordnet-based measures of lexical semantic relatedness [J]. Computational Linguistics, 2006, 32(1): 13-47
- [9] Chen Jun-peng, Liu Juan. Combining ConceptNet and WordNet for Word Sense Disambiguation[C]// International Joint Conference on Natural Language Processing, 2011: 686-694
- [10] Yu Weng, Qiu Li-rong. TECS: A Web Text Extraction Tool Based on Semantic Similarity Calculation[C]// 2011 4th International Conference on Intelligent Networks and Intelligent Systems(ICINIS), 2011: 25-228
- [11] Rada R, Mili H, Bicknell E, et al. Development and Application of a Metric on Semantic Nets[J]. IEEE Transactions on Systems, Man, and Cybernetics, 1989, 19(1): 17-30
- [12] Harispe S, Ranwez S, Janaqi S, et al. The Semantic Measures Library: Assessing Semantic Similarity from Knowledge Representation Analysis[M]// Natural Language Processing and Information Systems. Springer, 2014: 254-257
- [13] Mohammad S, Hirst G. Distributional Measures of Semantic Distance: A Survey[J]. Eprint Arxiv: 12031858v1
- [14] wikipedia Distributional_semantics [OL]http://en.wikipedia.org/wiki/Distributional_semantics
- [15] Richardson R F, Smeaton A, Murphy J. Using WordNet as a Knowledge Base for Measuring Semantic Similarity Between Words[C]// Proceedings of AICS Conference, 1994
- [16] 刘宏哲, 须德. 基于本体的语义相似度和相关度计算研究综述[J]. 计算机科学, 2012, 39(2): 8-13
Liu Hong-zhe, Xu De. Ontology Based Semantic Similarity and Relatedness Measures Review[J]. Computer Science, 2012, 39(2): 8-13
- [17] Batet M, Sánchez D, Valls A, et al. Semantic similarity estimation from multiple ontologies[J]. Applied Intelligence, 2013, 38(1): 29-44
- [18] 张忠平, 田淑霞, 刘洪强. 一种综合的本体相似度计算方法[J]. 计算机科学, 2008, 35(12): 142-145
Zhang Zhong-ping, Tian Shu-xia, Liu Hong-qiang. Composite Approach for Ontology Similarity Computation[J]. Computer Science, 2008, 35(12): 142-145
- [19] 方耀宁, 郭云飞, 扈红超, 等. 一种基于 Sigmoid 函数的改进协同过滤推荐算法[J]. 计算机应用研究, 2013, 30(6): 1688-1691
Fang Yao-ning, Guo Yun-fei, Hu Hong-chao, et al. Improved collaborative filtering recommender algorithm based on Sigmoid function[J]. Application Research of Computers, 2013, 30(6): 1688-1691
- [20] 文俊浩, 舒珊. 一种改进相似性度量的协同过滤推荐算法[J]. 计算机科学, 2014, 41(5): 68-71
Wen Jun-hao, Shu Shan. Improved Collaborative Filtering Recommendation Algorithm of Similarity Measure [J]. Computer Science, 2014, 41(5): 68-71