

基于语义的个性化用户兴趣模型^{*})

蒲国林^{1,2} 杨清平^{1,2} 王刚¹ 邱玉辉¹

(西南大学语义网格实验室 重庆 400715)¹ (四川文理学院计算机科学系 达州 635000)²

摘要 探讨了主动服务和个性化服务,给出了以主要特征词描述语义的用户访问历史的表示方法,建立了动态的用户兴趣模型。用户的兴趣模型表示为二层,上层是基于类兴趣的用户模型,下层是基于主要特征词兴趣的类模型。上层描述用户对各个类的兴趣度,下层描述用户对各类中的主要特征词的兴趣度。在建模过程中,通过启发式规则限制模型维度,通过行权重实现最近最新优先。

关键词 主动服务,个性化,访问历史,用户兴趣模型

Personalized Model of User Interests Based on Semantics

PU Guo-lin^{1,2} YANG Qing-ping^{1,2} WANG Gang¹ QIU Yu-hui¹

(Semantic Grid Laboratory, Southwest University, Chongqing 400715, China)¹

(Computer Science Department, Sichuan University of Arts and Science, Dazhou 635000, China)²

Abstract This paper discusses the active services and personalized services, describes semantics with the primary terms, offers the express method of user access histories. The dynamic user interests model is established. User interests model falls into two levels, the top level is a user model based on class interests; the down level is a class model based on primary term interests. The top level describes the class interests of user; the down level describes the primary term interests of classes. In the course of modeling, dimensionality is limited through heuristic rules, and the priorities are given through line-weights.

Keywords Active services, Personalization, Access history, User interests model

1 引言

Internet 的飞速发展和普及使用户对 Internet 应用有了更加智能化、个性化和综合化的要求。主动服务作为一种新的服务计算模式,能够自动适应用户的服务需求,依靠相关的工业标准和 Internet 已有的构件程序,为用户发现、定制和运行能够满足用户需求的构件程序^[1]。个性化是主动服务的主要特征之一。个性化服务是系统在对用户需求和访问历史进行语义处理的基础上,建立用户兴趣模型,并向用户推荐他正需要和最感兴趣的服务。现有的个性化服务系统广泛应用于电子邮件、Web 网页、电子报纸、新闻、新闻组、科技文献、数字图书馆、音乐、电影、电子商务等领域。

用户兴趣模型在个性化推荐服务中起着非常重要的作用,它反映了用户的个人信息和兴趣偏好。本文以主要特征词描述用户语义,构造用户访问历史矩阵,建立用户兴趣的二层语义模型,同时给出了模型降维的启发式规则和模型更新过程。

2 主动服务与个性化服务

2.1 主动服务

服务是自治、开放、自描述、与实现无关的网络构件,具有异步、可组合和与平台无关等特性。Web 服务是当前主要的服务实现技术。W3C 定义 Web 服务是由 URI 标识的软件应用程序,其接口和绑定可以通过 XML 构件进行定义、描述和发现。主动服务是一种 Web 服务,是构架在 XML 和 Internet

技术之上的分布式计算技术;主动服务是一种面向用户的服务,是服务响应本体主动查询用户的信息流,并提交可能的服务供其选择;主动服务能够根据用户的服务需求,从 Internet 或本地网络中搜索、发现、挖掘出能够提供用户服务需求的程序,并组装、编译和执行它们,为用户提供服务^[1]。

主动服务的前提是对用户的服务需求进行语义分析。推荐服务是主动服务的一个重要方面,它是对用户的服务需求的预测。主动推荐是对用户提交的文本进行自然语言处理的结果,是语义的,体现了用户的个性化需求。

主动服务模型^[1]如图 1 所示。

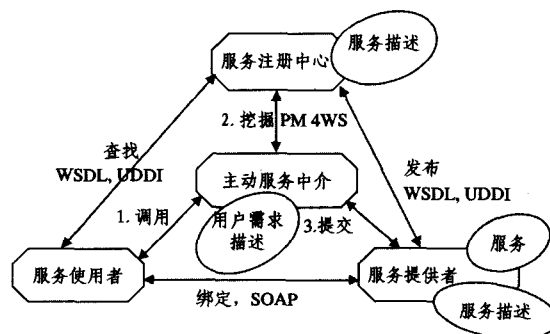


图 1 主动服务模型

2.2 个性化服务

Mizzaro 认为,个性化服务是收集并存储用户的使用信息,分析这些信息得到用户的特定兴趣和需求,然后在合适的

^{*}受西南大学研究生科技创新基金资助(2006011)。蒲国林 博士研究生,主要研究领域为语义网格、服务计算;杨清平 副教授,主要研究领域为人工智能;王刚 博士研究生,主要研究领域为语义网格、服务计算;邱玉辉 教授,博导,主要研究领域为人工智能、语义网格、服务计算。

时间向每一位访问者发送正确的信息。个性化主要体现在系统针对不同的用户提供不同的服务方式和自动搜索并主动向用户推荐他最需要和最感兴趣的服务。个性化服务系统的目的是利用用户兴趣模型提供更有针对性的个性化服务,产生最符合用户需求的服务列表。实现个性化服务的实质是根据服务之间的关联关系、用户的使用偏好等因素选取并推荐服务^[2]。因此,用户兴趣模型是服务推荐的前提,是个性化服务的基础,直接涉及到服务的质量。

3 用户访问历史

3.1 用户信息

为了对各种类型的由 URI 标识的网络资源实现个性化服务推荐,系统必须“知道”用户的个性化需求,这就需要获取、分析用户信息,建立合适的用户兴趣模型。用户信息主要包括:(1)用户基本信息。在用户 Profile 中,可以找到由用户自己填写的用户基本信息,包括用户的 ID、姓名、性别、年龄、民族、学历、职业、e-mail、QQ、电话、住址、工作单位、兴趣爱好等(部分隐私内容对某些用户来说是不愿公开的)。基本信息主要用于区分并识别用户身份,但用户的职业、兴趣、爱好等信息在用户初次使用时能作为系统个性化推荐的依据。(2)用户访问历史。用户访问历史主要包括用户输入和用户浏览的内容、服务器日志等。用户输入是多样化的,声频、视频、文字、图标等都是常用的输入形式。对这些信息进行理解分析都能转换成 XML 格式中的文本。其它信息需要对用户浏览的资源进行分析得到。

3.2 特征词加权的向量空间模型

3.2.1 相关概念

首先给出本文常用的几个基本概念,它们部分地使用了文献^[3]中的定义。

定义 1(文档) 泛指系统中索引以及检索的文本单元。在 Web 应用中也可指网络页面、部份页面或整个网络。

定义 2(文档集) 系统中用于满足用户需要的一组文档。

定义 3(特征词) 即检索词,指文档集中出现的词汇或短语。

定义 4(主要特征词) 指能够代表该文档内容和描述该文档特征的词汇或短语。主要特征词集合是特征词集合的子集。

定义 5(查询条件) 表示由一组特征词表达的用户查询信息。查询条件是用户需求的最直接的反应,因此查询条件中的特征词都是主要特征词。

本文用主要特征词描述用户的语义,即文档的意义只取决于它所包含的单词(主要特征词)。这种忽略句子语义的方法被称为单词包(Bag of Word)方法。本文是主要特征词获取之后的研究工作。

3.2.2 向量空间模型

给定文档集 $D = \{d_1, d_2, \dots, d_N\}$ 和主要特征词集 $F = \{F_1, F_2, \dots, F_n\}$, N 是文档集里的文档总数(查询条件也是文档集里的文档), n 是文档集中主要特征词的个数。则文档 d_j 和查询条件 q_k 表示为主要特征词权重的向量空间模型(Vector Space Model)^[4]:

$$d_j = (\omega_{1,j}, \omega_{2,j}, \dots, \omega_{i,j}, \dots, \omega_{n,j}) \quad (1)$$

$$q_k = (\omega_{1,k}, \omega_{2,k}, \dots, \omega_{i,k}, \dots, \omega_{n,k}) \quad (2)$$

其中, $\omega_{i,j}$ 表示特征词 F_i 在文档 d_j 中的权重, $\omega_{i,k}$ 表示特征

词 F_i 在查询条件 q_k 中的权重。 $\omega_{i,j} = 0$ 表示特征词 F_i 在文档 d_j 中没有出现。

于是把整个文档集看成一个权重矩阵:

$$D = \begin{bmatrix} \omega_{1,1} & \omega_{2,1} & \dots & \omega_{n,1} \\ \omega_{1,2} & \omega_{2,2} & \dots & \omega_{n,2} \\ \vdots & \vdots & \dots & \vdots \\ \omega_{1,N} & \omega_{2,N} & \dots & \omega_{n,N} \end{bmatrix} \quad (3)$$

矩阵的列表示文档集里的文档,矩阵的行表示主要特征词。

权重通常有三种表示方法:词频表示法、0-1 表示法和 TF-IDF^[5] 表示法。TF-IDF 表示考虑了特征词在单个文档中出现的频率和特征词在整个文档集里的分布。具体计算为:

$$\omega_{i,j} = tf_{i,j} \times idf_i \quad (4)$$

其中, $tf_{i,j}$ 是特征词 F_i 在文档 d_j 中出现的原始频率^[6]; idf_i 是倒排文档频率的特征词权重^[7], 即 $idf_i = \log(\frac{N}{n_i})$; N 是文档集里的文档总数; n_i 是文档集里出现特征词 F_i 的文档数。

查询条件文档平均长度只有约 2.3 个词^[8], 所以使用了与文档不同的加权公式计算其权重^[5]:

$$\omega_{i,k} = (0.5 + \frac{0.5tf_{i,k}}{\max_j tf_{i,k}}) \times idf_i \quad (5)$$

3.3 用户访问历史的表示

定义 6(访问文档) 就是用户一次登录访问的所有文档。访问文档是文档集的子集。

与文档集一样,访问文档也可以看成一个权重矩阵,矩阵的列表示访问文档,矩阵的行表示文档集中的主要特征词。

定义 7(访问兴趣) 是用户在一次访问中所形成的对对象的关注程度。在一次访问中,用户对某个特征词的访问兴趣可用该特征词的权重来表示。一次访问形成的所有主要特征词的权重就构成此次访问的访问兴趣向量。访问兴趣向量是对用户一次访问的抽象描述。

设 h^t 表示用户在第 t 次访问中形成的主要特征词访问兴趣向量; I_i^t 表示用户在第 t 次访问中对特征词 F_i 的兴趣度。则:

$$h^t = (I_1^t, I_2^t, \dots, I_i^t, \dots, I_n^t) \quad (6)$$

用户对特征词 F_i 的兴趣度 I_i^t 可由访问文档权重矩阵生成。简单地,在访问文档权重矩阵中对列算术平均即得到此次用户的访问兴趣向量。

定义 8(访问历史) 用户此次访问之前的所有访问就是用户的访问历史。一个用户的访问历史可用访问兴趣矩阵描述:

$$H^{T-1} = \begin{bmatrix} I_1^1 & I_2^1 & \dots & I_n^1 \\ I_1^2 & I_2^2 & \dots & I_n^2 \\ \vdots & \vdots & \dots & \vdots \\ I_1^T & I_2^T & \dots & I_n^T \end{bmatrix} \quad (7)$$

其中, H^{T-1} 是用户第 $T+1$ 次访问时的访问历史,它由用户的前 T 次访问的访问兴趣向量构成。

3.4 访问历史矩阵降维

访问历史矩阵是整个模型建立的基础。由于访问文档只是文档集的一个很小的子集,因此访问历史矩阵中有大量的冗余信息。而且随着时间的流逝,用户的访问兴趣要发生转移。所以对访问历史矩阵的降维可以在不丢失主要信息的情况下大幅度地降低模型的维度。

启发式规则 1 如果两行的相似度达到一个给定的阈值

σ 时,则删除权重较低的一行。可以通过计算两个相量之间夹角的余弦来计算两个文档的相似度。

启发式规则 2 如果矩阵的行数大于给定的阈值 T_X ,则删除首行。

启发式规则 3 如果某主要特征词权重为 0,则删除该行。该规则删除显然未访问的主要特征词列。

启发式规则 4 如果矩阵的列数大于给定的阈值 n_X ,则删除最后连续权重为 0 最多的主要特征词列。

设矩阵 Γ 通过降维后的行数为 T' ,主要特征词数为 n' ,经 $T=T'$ 和 $n=n'$ 重新赋值后,降维后的访问历史矩阵与矩阵 Γ 一致。后文的矩阵 Γ 就是指降维后的矩阵。

4 基于语义的用户兴趣模型

本文把用户兴趣模型分为二层。上层是基于类兴趣的用户模型,下层是基于主要特征词兴趣的类模型。上层描述用户对各个类的兴趣度,下层描述用户对各类中的主要特征词的兴趣度。

4.1 基于类兴趣的用户模型

给定 $\{C_1, C_2, \dots, C_M\}$ 是对降维后的矩阵 Γ 中的 n 个主要特征词的一个分类。其中, M 是类的个数;元素 $C_m (1 \leq m \leq M)$ 是一个具体的类名,它是主要特征词集的子集。这些类是这 n 个主要特征词集的一个划分,且同一个类的主要特征词间具有较高的相关性。即满足:(a)每一个类都有至少一个主要特征词,即不为空;(b)每一个主要特征词都必然属于某一个类;(c)任意一个主要特征词都不能同时属于两个不同的类,即任意两个类的交集为空;(d)任意一个主要特征词与同类的另一个主要特征词间的相关性都大于它与其它任意一个另类的主要特征词间的相关性。

用户对每个类都表现出一定的兴趣度。设 $I_{C_m}^{T+1}$ 表示用户在第 $T+1$ 次访问时对类 C_m 的兴趣度,则在第 $T+1$ 次访问之前用户兴趣模型 UIM^{T+1} 表示为向量:

$$UIM^{T+1} = (I_{C_1}^{T+1}, I_{C_2}^{T+1}, \dots, I_{C_M}^{T+1}) \quad (8)$$

设 $p_{C_m}^t (1 \leq t \leq T)$ 表示用户在第 t 次访问过程中形成的对类 C_m 的兴趣度,则用户访问历史 H^{T+1} 表示为类兴趣度矩阵 H_C^{T+1} :

$$H_C^{T+1} = \begin{bmatrix} p_{C_1}^1 & p_{C_2}^1 & \dots & p_{C_M}^1 \\ p_{C_1}^2 & p_{C_2}^2 & \dots & p_{C_M}^2 \\ \vdots & \vdots & \dots & \vdots \\ p_{C_1}^T & p_{C_2}^T & \dots & p_{C_M}^T \end{bmatrix} \quad (9)$$

其中

$$p_{C_m}^t = \sum_{i=1}^n I_{i,C_m} \quad (10)$$

而 I_{i,C_m} 被定义为:

$$I_{i,C_m} = \begin{cases} I_i & \text{当 } F_i \in C_m \\ 0 & \text{其它} \end{cases}$$

且 $1 \leq t \leq T, 1 \leq m \leq M$ 。

不能通过(9)式的简单求和得到(8)式中的 $I_{C_m}^{T+1}$,因为用户的不同访问兴趣向量对用户兴趣度的贡献不同,距离此次访问越新越近的访问兴趣向量应该赋予较高的权重。因此,对每行要乘上一个与时间有关的单调增加的函数。矩阵的行号 t 也就是访问序号,它的明确意义是用户的第 t 次访问,但它也隐含了时间的意义。即 t 越大,距离此次访问越近。于是问题就转为对不同行要找到一个与行号有关的递增的权重函数 $L(t)$ 。不妨取 $L(t) = \frac{t}{T(T+1)/2}$,它是行号与行号之总

和的比值。于是有:

$$I_{C_m}^{T+1} = \sum_{t=1}^T L(t) p_{C_m}^t = \frac{\sum_{t=1}^T t \sum_{i=1}^n I_{i,C_m}}{T(T+1)/2} \quad (11)$$

4.2 基于特征词兴趣的类模型

仅仅有基于类兴趣的用户模型还不够,因为在个性化推荐时系统不仅要知道用户感兴趣的类,而且还要知道对这些类中的每个主要特征词的兴趣度。基于特征词兴趣的类模型正是对后者的描述。

用户在第 $T+1$ 次访问时对类 C_m 的兴趣度为 $I_{C_m}^{T+1}$,用户对类 C_m 的兴趣度可以分解为基于特征词兴趣的类模型 $CIM_{C_m}^{T+1}$:

$$CIM_{C_m}^{T+1} = (p_{F_1,C_m}^{T+1}, p_{F_2,C_m}^{T+1}, \dots, p_{F_n,C_m}^{T+1}) \quad (12)$$

其中, p_{F_i,C_m}^{T+1} 表示特征词 F_i 在类 C_m 中的权重。

同理不能通过(7)式的简单求和得到(12)式中的 p_{F_i,C_m}^{T+1} 。必须使用与前文一致的行权重函数,所以:

$$p_{F_i,C_m}^{T+1} = \sum_{t=1}^T I_{i,C_m} \times L(t) \quad (13)$$

其中 I_{i,C_m} 与前文定义一致。

5 用户兴趣模型降维与更新

5.1 用户兴趣模型降维

通过对访问历史矩阵的降维,已经把显然的无关信息去掉,但模型中还是有冗余信息。这些信息权重较低,不能形成用户的主要兴趣。合理地删减部份冗余信息,能有效地降低模型的维度。何况用户不可能对所有的类都感兴趣,即使对某类感兴趣,也不是对类中的所有主要特征词感兴趣。正如某用户喜欢看武打电影,但并不是所有的武打电影他都喜欢看,或许他只喜欢看成龙主演的武打电影。

用户兴趣模型主要用于推荐服务,推荐的是用户最需要的信息资源或服务资源。所以在模型中必须保留用户最感兴趣的类和主要特征词,而对权重较小的类和主要特征词则可删除。

启发式规则 5 如果分类数大于给定的阈值 M' 时,则按类的兴趣度大小排序仅保留前 M' 个类。

启发式规则 6 如果某类中的主要特征词个数大于给定的阈值 n'_c 时,则按主要特征词的兴趣度大小排序仅保留前 n'_c 个主要特征词。

在这些启发式规则中,阈值的选择是关键。合理的阈值能减少很多冗余信息,同时不丢失主要信息。

5.2 用户兴趣模型的更新

用户每一次登录时,系统除了提供用户正需要的服务外,还要根据用户的兴趣模型向用户推荐他最可能需要的也就是最感兴趣的服务。

用户注册时,系统根据用户基本信息中的职业、兴趣、爱好等初始化 UIM^1 和 $CIM_{C_m}^1$ 。

用 h_{new} 表示新的访问兴趣向量,则用户兴趣模型更新过程描述如下:

已知模型 UIM^{T+1} 和 $CIM_{C_m}^{T+1}$ 。

输入: h_{new} ; 输出: 新的 UIM^{T+1} 和 $CIM_{C_m}^{T+1}$

a) 系统读取已知模型 UIM^{T+1} 和 $CIM_{C_m}^{T+1}$, 实现服务推荐(按类和主要特征词的权重大小进行推荐);

b) 在用户结束此次访问时,系统获取 h_{new} , 且执行赋值:

$T = T + 1, h^T = h_{new}$;

c) 通过启发式规则 1 到 4 实行访问历史矩阵降维;

- d)用 11 式计算出新模型 UIM^{T+1} ;
- e)用 13 式计算出新模型 CIM_m^{T+1} ;
- f)通过启发式规则 5 和 6 对新模型 UIM^{T+1} 和 CIM_m^{T+1} 降维;
- g)输出并保存 UIM^{T+1} 和 CIM_m^{T+1} , 供下次推荐使用。

结束语 用户兴趣模型是用户兴趣的语义表示,是个性化推荐服务的基础。本文以用户访问历史为基础,构造访问历史的主要特征词权重矩阵,建立动态变化的二层语义用户兴趣模型。在个性化推荐中,该模型能实现二层语义匹配。通过上层匹配,系统知道用户最感兴趣的类;通过下层匹配,系统知道这些类中用户最感兴趣的主要特征词。文中主要是对模型的建立、模型的降维和更新进行了研究。当然,与其它模型一样,也能用该模型对用户按兴趣进行分组,实现协同过滤推荐,为用户发现新的可能感兴趣的资源。

(上接第 128 页)

(2) 在 U 中选择 p_i^* 认为优先属于 W 的 q 个元素,得到 U 的子集 $U_i^{(j)}$;

(3) 依次类推,在 U 中选择 p_i^* 认为优先属于 W 的 sq 个元素,得到 U 的子集 $U_i^{(j)}$,显然, $U_i^{(j)} \subset U_i^{(j)}$; 当 $s * q \geq k$ 时,迭代中止于第 $t(t=s)$ 代;

(4) 求取 U_i 的覆盖频率为

$$m(u_i) = \sum_{s=1}^t \sum_{j=1}^n \chi_{U_i^{(j)}}(u_i) \quad (i=1, 2, \dots, k) \quad (9)$$

其中, $\chi_{U_i^{(j)}}$ 为集合 $U_i^{(j)}$ 的特征函数。

(5) 最后是归一化处理,从而得到各个指标的权重系数,可以得到:

$$W(u_i) = m(u_i) / \sum_{j=1}^k m(u_j) \quad (i=1, 2, \dots, k) \quad (10)$$

2) 加权关联度求取

权重求取之后,就可以进行加权关联度的计算了。求取类别的加权关联度即为求取当前服务器对象对某一负载级别的隶属程度。在这里我们给出隶属程度的定义:

定义 4(隶属程度) 若 w_1, w_2, \dots, w_n 为负载均衡过程的特征权重, k_c 为关联函数矩阵,则

$$K(p_j) = \sum_{i=1}^n w_i K_{p_j c_i} \quad (1 \leq j \leq m) \quad (11)$$

为当前服务器物元对象对负载等级 j 的隶属程度。

4.5 获取负载状态等级

对于集群中的任一服务器对象,计算其对各个负载等级的隶属程度,然后将获得最大隶属程度的类别定为最终的输出结果,这就是负载状态等级的获取。过程如下:

(1) 对于论域中的所有类别,分别计算出服务器对象对其的隶属程度 $K(p_i) (1 \leq i \leq m)$;

(2) 若 $K(p_k) (K(p_k) < 0)$, 则将类别 k 忽略;

(3) 在所有大于零的 $K(p_m)$ 中,求取出:

$$K(p_\theta) = \max(K(p_j)) \quad (j=1, 2, \dots, m, j \neq k) \quad (12)$$

将类别 θ 定为最终负载均衡结果。结果表示为“在 M 台服务器组成的集群中,服务器 $R (1 \leq R \leq M)$ 目前的运行状态处于负载等级 $\theta (1 \leq \theta \leq m)$ ”。

4.6 进行负载均衡控制

到此,我们确定了集群中所有服务器的负载状态等级。前面提过,负载等级的获取是动态负载均衡策略模型的关键。在该模型中,负载等级的数量和分法可以由使用负载均衡的客户自行定制。在我们的设计中,定为“无负载、较轻负载、轻负载、较重负载、重负载、满负载”六种负载等级。在 DNS 负

参考文献

- [1] 张尧学,方存好. 主动服务——概念、结构与实现. 北京,科学出版社,2005(1):3,32,50
- [2] 喻坚,等. 面向服务的计算——原理和应用. 清华大学出版社,2006(12):258-261
- [3] Jurafsky D, Martin J H. 自然语言处理综论. 冯志伟,等译. 北京:电子工业出版社,2005(6):403-406
- [4] Salton G, McGill M J. Introduction to Modern Information Retrieval. McGraw-Hill, New York, 1983
- [5] Salton G, Buckley C. Term Weighting Approaches in Automatic Text Retrieval. Information Processing and Mangement, 1988, 24(5):513-523
- [6] Luhn H P. A statistical approach to the mechanized encoding and searching of literary information. IBM Journal of Research and Development, 1957, 1(4):309-317
- [7] Sparck J K. A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation, 1972, 28(1):11-21
- [8] Silverstein C, Henzinger M, Marais H, et al. Analysis of a very large AltaVista querylog. Tech. rep. 1998-014

载均衡控制中,关键是选择一个负载最小的服务器来响应用户的服务请求。因此,在每一次服务请求中,模型都会在得到所有服务器的负载等级后,按照某种排序对服务器进行任务调度分配,通常是选取集群中当前负载量最小的一个或者几个服务器进行任务分配,从而完成异构服务器群集动态负载均衡控制。

我们通过典型范例,在 VC6.0 和 matlab7.0 上进行实验和画图,证明该模型和采用的算法比一般的 DNS 动态负载均衡算法具有有效性和实用性。

结束语 本文针对 DNS 动态负载均衡一般算法的不足,利用可拓检测技术和模糊数学的原理,构建出一个基于可拓的动态负载均衡模型,它是可拓检测技术和模糊数学的一种全新的应用。利用该方法进行负载状态的确定和负载均衡控制,不仅克服了一般策略的不足,而且易于实现,操作性强,更加重要的是提出了一种通用性强、与传统思维方法不同的负载均衡控制方法,且算法简单,具有很强的健壮性。因此,这种基于可拓检测技术的动态负载均衡算法具有很好的应用前景,而这种用矛盾思想把不可测问题转化成可用可测问题进行表示并解决的思维方式,对工程上的其它难题更是具有指导意义。

参考文献

- [1] 熊智,晏蒲柳,郭成城. Web 集群中文档组织分布的优化策略. 计算机科学,2006,33(12):114-118
- [2] 张普,王青,杨立光. 网络计算机集群负载均衡机制的研究. 计算机工程与设计,2006,27(16):2914-2917
- [3] Cardellini V, Colajanni M. Dynamic loadbalancing on Web-server systems. IEEE Internet Computing, 1999, 3(3):28-39
- [4] Schroeder T, Goddard S, Ramamurthy B. Scalable Web Server Clusterint Technologies. IEEE Network, May/June 2000:38-45
- [5] Yu Yongquan, Peng Haixia, Ye Weigiong. The Principle of Extension Detecting with Extension Sets // International Conference on Computer, Communication and Control Technologies, CCCCT'03. Orlando, Florida, USA, July 2003
- [6] Fu Yong, Wang Hongan, Lu Chenyang, et al. Distributed Utilization Control for Real-Time Clusters with Load Balancing // 27th IEEE International Real-Time Systems Symposium (RTSS'06). 2006:137-146
- [7] 王洪伟,吴家春,蒋馥. 基于可拓集的决策模型研究. 计算机科学,2003,30(8):130-133
- [8] 蔡文,杨春燕,林伟初. 可拓工程方法[M]. 北京:科学出版社,1999:95-101
- [9] 蔡文. 可拓论及其应用. 科学通报,1999,44(17):673-682
- [10] 余永权. 可拓检测技术. 中国工程科学,2001,3(4):89-94
- [11] 余永权,曾碧. 单片机模糊逻辑控制. 北京航空航天大学出版社,1997:21-33