(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 1728147 B
(45)授权公告日 2010.09.08

(21)申请号 200510092244.8

(22)申请日 2005.05.16

(30)优先权数据
10/846,949 2004.05.14 US

(73)专利权人 微软公司
地址 美国华盛顿州

(72)发明人 B·章 G·薛 H-J·曾 马维英 陈正

(74)专利代理机构 上海专利商标事务所有限公司 31100
代理人 李玲

(51)Int.Cl.
G06F 17/30(2006.01)

(56)对比文件

Wen J-R et, al.Query clustering using user logs.ACM transactions on information systems20 1.2002,20(1),62-78<http://portal.acm.org/citation.cfm?id=503108&coll=ACM&dl=ACM&CFID=54162463&CFTOKEN=34995373>.

John A. Tomlin.A new paradigm for ranking pages on the World Wide Web. Proceedings of the 12th international conference on World Wide Web, ACM.2003,350-352<http://portal.acm.org/citation.cfm?id=775202&coll=ACM&dl=ACM&CFID=54162463&CFTOKEN=34995373>.

Taher H. Haveliwala.Topic-sensitivepagerank:Acontext-sensitiverankingalgorithmfor web search. Proceedings of the 11th international conference on World Wide Web, ACM.2002,517-526<http://portal.acm.org/citation.cfm?id=511513&coll=ACM&dl=ACM&CFID=54162463&CFTOKEN=34995373>.

审查员 谭李丽

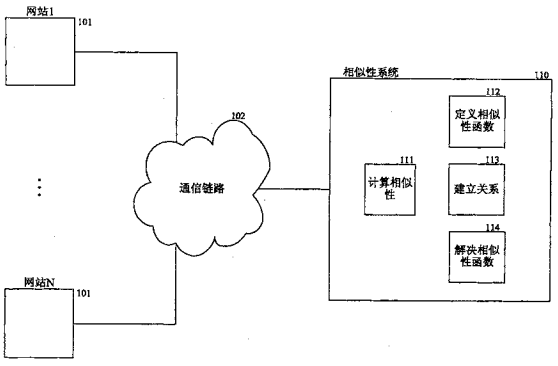
权利要求书 3 页 说明书 7 页 附图 5 页

(54)发明名称

基于异类关系确定目标相似性的方法和系统

(57)摘要

提供一种用于测量目标的相似性的方法和系统，所述测量基于同种类型的目标和不同类型的目标之间的关系以及这些目标与其它目标之间的相似性。在一个实施例中，相似性系统为每一种目标定义内部型和中间型相似性函数。相似性系统可以将某种类型的内部型和中间型相似性函数组合成该类型的一个整体的相似性函数。在定义了相似性函数之后，相似性系统收集目标的属性值，其包括同种类型的目标之间的关系数据，叫作内部型关系，和不同类型的目标之间的关系，叫作中间型关系。在收集了目标的属性值之后，相似性系统通过反复计算目标的相似性来求解内部型和中间型相似性函数，直到相似性收敛于一解值。



CN 1728147 B

1. 一种在计算机系统中产生目标之间相似性测量的方法，每个目标具有多种类型中的一种，单个类型的目标具有内部型关系，一对类型的目标具有中间型关系，所述方法包括：

对于每一类型的目标，

当该每一类型的目标之间的相似性是基于内部型关系时，为每个这样的关系提供内部型相似性函数，其用来测量该每一类型的目标之间的相似性；

当该每一类型的目标之间的相似性是基于与另一类型的目标的中间型关系时，为每个这样的关系提供中间型相似性函数，其基于另一种类型的目标的相似性来测量该每一类型的目标之间的相似性，其中所述中间型相似性函数生成该每一类型的第一目标和第二目标的相似性，所述相似性是其它类型的目标对之间的相似性的加权平均值，其中所述目标对的一个目标与所述第一目标有关系，所述目标对的另一目标与所述第二目标有关系；和

提供相似性函数，其基于该每一类型的任何内部型相似性函数和任何中间型相似性函数来测量该每一类型的目标之间的相似性；以及

对于每个关系，提供定义与该关系有关的目标之间的关系的数据；

基于提供的数据定义的关系来同时求解提供的相似性函数，其中通过基于相似性功能反复计算目标的相似性直到误差测量值收敛来求解所述相似性函数；和

基于所述提供的相似性函数的同时求解来存储相似性。

2. 如权利要求 1 所述的方法，其中基于该每一类型的相似性函数递归地定义该每一类型的内部型相似性函数。

3. 如权利要求 1 所述的方法，其中基于另一种类型的相似性函数递归地定义该每一类型的中间型相似性函数。

4. 如权利要求 1 所述的方法，其中一种类型的相似性函数是该每一类型的内部型和中间型相似性函数的线性组合。

5. 如权利要求 4 所述的方法，其中给每个内部型和中间型相似性函数一个加权值。

6. 如权利要求 5 所述的方法，其中该每一类型的内部型和中间型相似性函数的加权值总和为 1。

7. 如权利要求 1 所述的方法，其中基于从一个反复计算到下一个反复计算的相似性，当误差测量值低于门限误差值时，求解所述相似性函数。

8. 如权利要求 1 所述的方法，其中目标的类型包括网页和查询，并且查询和网页之间的中间型关系基于从查询到网页的点击。

9. 如权利要求 1 所述的方法，其中目标的类型包括网页和查询，并且网页之间的内部型关系基于引入和引出链接，网页和查询之间的中间型关系基于从查询到网页的点击。

10. 一种用于控制计算机系统产生目标之间相似性测量的方法，每个目标具有多种类型中的一种，所述方法包括：

为每一类型的目标提供相似性函数，当为该每一类型的目标定义了内部型相似性时，基于该每一类型的目标之间的内部型相似性来测量该每一类型的目标的相似性，当为该每一类型的目标定义了与另一类型的目标的中间型相似性时，基于另一种类型的目标的相似性来测量该每一类型的目标之间的中间型相似性，其中该每一类型的第一目标和第二目标的相似性是其它类型的目标对之间的相似性的加权平均值，其中所述目标对的一个目标与所述第一目标有关系，所述目标对的另一目标与所述第二目标有关系；

对于每个关系，提供定义与该关系有关的目标之间的关系的数据；

基于提供的数据定义的关系来同时求解提供的相似性函数，其中通过基于相似性功能反复计算目标的相似性直到误差测量值收敛来求解所述相似性函数；和

基于所述求解的相似性函数来存储相似性。

11. 如权利要求 10 所述的方法，其中相似性函数定义一组线性方程式。

12. 如权利要求 10 所述的方法，其中基于不同类型的目标的相似性递归地定义相似性函数。

13. 如权利要求 10 所述的方法，其中基于不同关系的该每一类型目标的相似性递归地定义相似性函数。

14. 如权利要求 10 所述的方法，其中一种类型的相似性函数是该每一类型的内部型和中间型相似性的线性组合。

15. 如权利要求 14 所述的方法，其中给每个内部型和中间型相似性一个加权值。

16. 如权利要求 15 所述的方法，其中该每一类型的内部型和中间型相似性的加权值总和为 1。

17. 如权利要求 10 所述的方法，其中基于从一个反复计算到下一个反复计算的相似性，当误差测量值低于门限误差值时，求解所述相似性函数。

18. 一种用于计算目标相似性分数的计算机系统，每个目标具有多种类型中的一种并且每一类型的目标与另一种类型的目标之间具有中间型关系，所述系统包括：

用于每一类型的目标的元件，其实现相似性函数，所述相似性函数基于该每一类型目标与另一种类型目标之间的中间型相似性来为该每一类型的目标对提供相似性分数，基于其它类型目标对的相似性分数和目标对之间的中间型关系来递归地定义中间型相似性，其中该每一类型的第一目标和第二目标的相似性是其它类型的目标对之间的相似性的加权平均值，其中所述目标对的一个目标与所述第一目标有关系，所述目标对的另一目标与所述第二目标有关系；和

一元件，其基于为一组目标定义的中间型关系通过反复调用实现相似性函数的元件来求解相似性函数，直到相似性分数收敛；和

一用于基于求解的相似性函数来存储相似性的元件。

19. 如权利要求 18 所述的计算机系统，其中相似性函数定义一组线性方程式。

20. 如权利要求 18 所述的计算机系统，其中一种类型具有该每一类型目标之间的内部型关系，相似性函数进一步基于该每一类型目标之间的内部型相似性，所述类型基于目标之间的内部型关系。

21. 如权利要求 20 所述的计算机系统，其中一种类型的内部型相似性基于该每一类型和另一种类型的目标之间的中间型相似性。

22. 如权利要求 20 所述的计算机系统，其中基于不同内部型关系的目标的内部型相似性递归地定义相似性函数。

23. 如权利要求 20 所述的计算机系统，其中一种类型的相似性函数是该每一类型的内部型和中间型相似性函数的线性组合。

24. 如权利要求 23 所述的计算机系统，其中给每个内部型和中间型相似性函数一个加权值。

25. 如权利要求 24 所述的计算机系统，其中内部型和中间型相似性函数的加权值总和为 1。

26. 如权利要求 18 所述的计算机系统，其中基于从一个反复调用到下一个反复调用的相似性，当误差测量值低于门限误差值时，相似性分数收敛。

27. 如权利要求 18 所述的计算机系统，其中基于从一个反复调用到下一个反复调用的相似性，当误差测量值低于门限误差值时，求解所述相似性函数。

基于异类关系确定目标相似性的方法和系统

技术领域

[0001] 所述技术通常涉及确定目标相似性，尤其是基于目标关系来确定所述相似性。

背景技术

[0002] 许多搜索引擎服务，诸如 Google 和 Overture，提供了对接入因特网的信息的搜索。这些搜索引擎服务允许用户搜索其感兴趣的显示页，例如网页。用户发送一个包括搜索项的搜索请求（也叫作“查询”）之后，搜索引擎服务识别出与那些搜索项相关的网页。为了快速识别出相关的网页，搜索引擎服务可以保存关键词到网页的映射。搜索引擎服务可以通过“扫描”网络（也就是万维网）提取每个网页的关键词来生成这种映射关系。为了扫描网络，搜索引擎服务可以使用根网页的列表和识别所有通过这些根网页接入的网页。任何特定网页的关键词都能用各种公知的信息检索技术提取出来，例如识别标题的关键字、网页的元数据中提供的关键字、突出显示的关键字等。搜索引擎服务可以基于每个匹配的相近度、网页普遍性（如 Google 的 PageRank）等来计算相关性分数，所述相关性分数表示每个网页与搜索请求有多相近。然后搜索引擎服务以相关性顺序向用户显示与那些网页的链接。搜索引擎一般地还可以提供在任何文件集合中搜索信息。例如，所述文件集合可能包括所有美国专利、所有联邦法庭判决、公司所有的档案文件等。

[0003] 搜索引擎服务可能需要测量各种目标之间的相似性，诸如网页或查询。例如，搜索引擎服务可以允许交互查询扩展，其需要查询项与其它项之间的相似性计算。如另一个例子中，搜索引擎服务想要将网页分成相似网页的群，以通过网页帮助用户导航。确定目标相似性的典型算法通常使用与目标相关的特征向量，然后计算特征向量之间的距离来作为相似性的表示。例如，网页可具有包括用于计算相似性的关键字、内容等特征。在确定相似性时大多数算法只依靠与目标相关的特征。例如，网页之间的相似性只基于网页的内容。但是少数算法受异类目标特征的影响。例如，一种算法使用击穿数据，其中如果它们包括相同的项或者导致用户选择相同网页，查询就是相似的。因此，这种查询的特征向量包括由用户选择的查询结果网页上的信息。

[0004] 但是当计算一种类型目标之间的相似性时，这些技术就不能考虑相关的其它类型的目标之间的相似性。也就是说，一种类型目标的相似性测量可能与另一种类型目标的相似性测量有关。例如，部分地基于用户选择的或点击的网页之间的相似性，一个查询可能与另一个相似。相反，部分地基于在其结果中返回网页的查询之间的相似性，一种网页可能与另一种网页相似。所以希望能有一种用于测量受异类目标之间关系影响的目标相似性的技术。

发明内容

[0005] 提供了一种基于同种类型和不同种类型的目标关系来测量目标相似性以及这些目标与其它目标的相似性的方法和系统。在一个实施例中，相似性系统为每一种类型的目标定义内部型和中间型相似性函数。相似性系统可以将某种类型的内部型和中间型相似性

函数组合成那种类型的一个整体的相似性函数。在定义了相似性函数之后，相似性系统收集目标的属性值，其包括同种类型的目标之间的关系数据，叫作内部型关系，以及不同种类型的目标之间的关系，叫作中间型关系。在收集了目标的属性值后，相似性系统通过反复计算目标的相似性直到相似性收敛于一个解值来求解内部型和中间型相似性函数。

附图说明

[0006] 图 1 是描述一个实施例中的相似性系统的元件的框图。

[0007] 图 2 是描述一个实施例中定义相似性函数元件的处理流程图。

[0008] 图 3 是描述一个实施例中建立关系元件的处理流程图。

[0009] 图 4 是描述一个实施例中求解相似性函数元件的处理流程图。

[0010] 图 5 是描述一个实施例中计算相似性函数元件的处理流程图。

具体实施方式

[0011] 提供了一种基于同种类型和不同种类型的目标关系来测量目标相似性以及这些目标与其它目标的相似性的方法和系统。在一个实施例中，相似性系统为每一种类型的目标定义内部型和中间型相似性函数。内部型相似性函数测量同种类型的目标之间的相似性。例如，查询之间的内部型相似性函数可以基于查询的搜索项匹配有多相近，基于发送查询的用户的属性。目标之间的内部型相似性也可以依赖于同种类型的其它目标的相似性。例如，如果两个查询每个都与第三个查询高度相似，那么这两个查询就更相似。所述基于其它目标的相似性的目标之间的内部型相似性定义了一种递归函数。中间型相似性函数基于另一种类型的目标属性（包括相似性）来测量一种类型的两个目标之间的相似性。例如，如果用户点击的一个查询结果的网页与用户点击的其它查询结果的网页相似，那么这两个查询会更相似。其它类型的目标的相似性还可以依赖于一种类型的目标的相似性。而且，由于一种类型的目标的相似性可以依赖于其它类型的目标的相似性，并且反之亦然，所以中间型相似性函数在不同类型间是递归的。

[0012] 目标的类型可以基于目标的不同属性有各种相似性定义。例如，网页可以有基于网页内容的内部型相似性和基于网页之间的链接的另一种内部型相似性。相似性系统可以将某种类型的内部型和中间型相似性函数组合成那种类型的一个整体的相似性函数。在一个实施例中，相似性系统通过基于其察觉的精确度为每个内部型和中间型相似性函数加权的线性方程来组合内部型和中间型相似性函数，以表示那种类型的目标的整体相似性。例如，可以给具有高精确度的内部型相似性函数一个高的加权值，给具有低精确度的内部型相似性函数一个低的加权值。

[0013] 在定义了相似性函数后，相似性系统收集目标的属性值，其包括同种类型的目标之间的关系数据，叫作内部型关系，以及不同种类型目标之间的关系，叫作中间型关系。例如，网页可以具有与网页的关键字相对应的非基于关系的属性值。网页还可以具有基于网页之间的引入和引出链接的内部型关系。网页可以具有基于从网页查询结果中点击的带有查询的中间型关系。

[0014] 在收集了目标的属性值后，相似性系统通过反复计算目标的相似性直到相似性收敛于一个解值来求解内部型和中间型相似性函数。相似性系统使用反复手段是因为相似性

函数的递归特性。相似性系统用已初始化的相似性来启动，然后基于初始相似性来为每一种目标计算相似性函数，以给出新的相似性。相似性系统测量新相似性与旧相似性之间的区别来确定相似性是否收敛于一个解值。如果收敛了，新的相似性就代表所述解值。如果没有收敛，相似性系统就重复使新相似性变成旧相似性的过程。因此相似性系统基于另一种目标的相似性和不同种类型的目标之间的关系计算出一种类型的目标的相似性。

[0015] 下面，提供一个在搜索引擎范围内的相似性系统的处理过程的例子。相似性系统将搜索引擎使用的目标（例如网页和查询）和关系（例如引入链接和点击）模拟成定向图 G = (V,E)，其中节点 V 表示搜索引擎的目标，边缘 E 表示目标之间的关系。节点 V 可以分成两个子集 Q = $\{q^1, q^2, \ldots, q^m\}$ 和 P = $\{p^1, p^2, \ldots, p^n\}$，其中 Q 表示查询，P 表示网页。这些网页和查询之间的关系可能包括一引入链接关系 (IL)，一引出链接关系 (OL) 和点击关系 (CT)。对于图中的节点 v，$M_R(v)$ 表示具有关系 R 和节点 v 的相邻节点的集合。例如，$M_{IL}(v)$ 表示网页 v 的引入链接的源点网页的集合。$M_R^i(v)$表示集合中的第 i 个网页。相似性系统用相似性矩阵 S 来表示目标之间的相似性，S[a，b] 表示目标 a 和 b 之间的相似性。

[0016] 相似性系统基于一种原则：一种类型的目标是相似的，是部分地根据另一种类型的相关目标的相似性。如果一种类型的两个目标与另一种类型的相同目标有关系，那么这两个目标范围相似。同样，如果同样类型的两个目标与另一种类型的两个不同但类似的目标有关系，那么这两个目标范围相似。相似性系统用下面的方程来表示这个原则：

[0017] $$S_{O_1}[a,b] = \frac{C}{|M_R(a)||M_R(b)|} \sum_{i=1}^{|M_R(a)|} \sum_{j=1}^{|M_R(b)|} S_{O_2}[M_R^i(a), M_R^j(b)] \qquad (1)$$

[0018] 其中S_{O_1}表示 O_1 类型的目标 a 和 b 之间的相似性，S_{O_2}表示另一类型的目标 i 和 j 之间的相似性，R 表示相似性所依据的中间型关系，C 是加权因子。如果 a 等于 b，那么$S_{O_1}[a,b]$就定为 1，也就是，一个目标与其本身之间的相似性定义为最大的相似性值 1。如果 a 和 b 都与 O_2 中相同的目标 A 有关，那么$S_{O_2}[A,A]$是 1，其对$S_{O_1}[a,b]$给出了最大基值。如果 a 或 b 都没有相邻节点，也就是与 O_2 中的目标没有关系，那么 $|M_R(a)|$ 或 $|M_R(b)|$ 将等于 0。在这种情况下，相似性系统将$S_{O_1}[a,b]$设为 0，防止从 0 分离。作为一个例子，假设 O_1 包含目标 a 和 b，O_2 包含目标 A、B 和 C，并且 a 与 A 和 B 有关，b 与 B 和 C 有关。如果$S_{O_2}[A,B]$是 .7，$S_{O_2}[B,C]$是 .7，并且$S_{O_2}[B,C]$是 .49，加权因子是 .7，那么通过应用方程式 1，$S_{O_1}[a,b]$是 .5（例如 .7/4*(.7+.49+1.0+.7)）。

[0019] 相似性系统基于从内部型相似性函数和中间型相似性函数中导出的相似性的合并来定义整个目标类型的相似性。在一个实施例中，相似性系统使用内部型相似性函数和中间型相似性函数的相似性的线性组合，如下面的方程式所示：

[0020] $S[a,b] = \alpha S_{int\ ra}[a,b] + \beta S_{int\ er}[a,b]$ (2)

[0021] 其中 $S_{int\ ra}$ 和 $S_{int\ er}$ 表示从内部型相似性函数和中间型相似性函数中导出的相似性，α 和 β 是相似性的加权，并且 α + β = 1。通过给 α 和 β 分配不同的值，相似性系统可以将不同相似性函数的基值调整成统一的相似性值。如上所述，方程式 2 可以递归式地定义，因为一个目标的相似性可以基于另一个目标的相似性来定义，这可以基于一个目标的相似性而依次定义。在一个实施例中，相似性系统通过计算相似性来反复求解相似性函数，直到它们收敛（也就是 $||S^i - S^{i-1}|| < \varepsilon$，其中 ε 是误差门限值）。

[0022] 在搜索引擎范围内，相似性系统可以只用查询的内容来定义内部型相似性函数。基于内容的内部型相似性函数可以通过下面的方程式来定义：

[0023] $$S_{QC}[a,b]=\frac{|Keyword(a)\cap Keyword(b)|}{|Keyword(a)\cup Keyword(b)|} \quad (3)$$

[0024] 其中 a 和 b 是查询，S_{QC} 是基于内容的查询的内容相似性矩阵。作为一个例子，当查询 a 和 b 有两个搜索项（或关键字）并且其中一个关键字是共用的时，它们的相似性值将是 .33（也就是 1/3）。相似性系统可以基于对网页的点击关系通过下面的方程式来为查询定义中间型相似性函数：

[0025] $$S_{QCT}[a,b]=\frac{C_{CT}}{|M_{CT}(a)||M_{CT}(b)|}\sum_{i=1}^{|M_{CT}(a)|}\sum_{j=1}^{|M_{CT}(b)|}S_{PCT}[M_{CT}^{i}(a),M_{CT}^{j}(b)] \quad (4)$$

[0026] 其中 S_{QCT} 表示基于点击的查询的相似性矩阵，S_{PCT} 表示基于点击的网页的相似性矩阵，$M_{CT}(a)$ 表示从查询 a 到从查询标记中识别出的网页的点击，C_{CT} 是加权因子。相似性系统将方程式 (3) 和 (4) 组合成一个用于查询的整体的相似性函数，由下面的方程式表示：

[0027] $$S_{Q}[a,b]=\alpha S_{QC}[a,b]+\beta S_{QCT}[a,b] \quad (5)$$

[0028] 其中 S_Q 表示查询的整体的相似性矩阵。

[0029] 相似性系统基于引入链接和引出链接的内部型关系以及引起点击网页的查询的中间型关系来表示网页的相似性。相似性系统基于引入链接关系来定义内部型相似性函数，以反映出当两个网页由相同的网页（或相似的网页）链接时它们是相似的。相似性系统还基于引出链接关系来定义中间型相似性函数，以反映出当两个网页链接到相同的网页（或相似的网页）时，它们是相似的。相似性系统基于引出和引入链接关系通过下面的方程式来为网页表示内部型相似性函数：

[0030] $$S_{OL}[A,B]=\frac{C_{OL}}{|M_{OL}(A)||M_{OL}(B)|}\sum_{i=1}^{|M_{OL}(A)|}\sum_{j=1}^{|M_{OL}(B)|}S_{IL}[M_{OL}^{i}(A),M_{OL}^{j}(B)] \quad (6)$$

[0031] $$S_{IL}[A,B]=\frac{C_{IL}}{|M_{IL}(A)||M_{IL}(B)|}\sum_{i=1}^{|M_{IL}(A)|}\sum_{j=1}^{|M_{IL}(B)|}S_{IL}[M_{IL}^{i}(A),M_{IL}^{j}(B)] \quad (7)$$

[0032] 其中 A 和 B 表示网页，C_{OL} 和 C_{IL} 表示加权因子，S_{OL} 和 S_{IL} 是基于引出和引入链接的相似性矩阵，$M_{OL}(A)$ 表示网页 A 的引出链接的目的地网页，$M_{IL}(A)$ 表示到网页 A 的引入链接的网页源点。相似性系统基于点击关系通过下面的方程式来为网页表示中间型相似性函数：

[0033] $$S_{PCT}[A,B]=\frac{C_{CT}}{|M_{CT}(A)||M_{CT}(B)|}\sum_{i=1}^{|M_{CT}(A)|}\sum_{j=1}^{|M_{CT}(B)|}S_{QCT}[M_{CT}^{i}(A),M_{CT}^{j}(B)] \quad (8)$$

[0034] 其中 $M_{CT}(A)$ 表示用户点击的用来访问网页 A 的查询。由于方程式 8 是根据方程式 4（也就是 S_{QCT}）来定义的，反之亦然，这对方程式定义一种递归功能。相似性系统将网页的整体相似性函数定义成内部型相似性函数和中间型相似性函数的线性组合，其可以由下面的方程式来表示：

[0035] $$S_{P}[A,B]=\alpha' S_{OL}[A,B]+\beta' S_{IL}[A,B]+\gamma' S_{PCT}[A,B] \quad (9)$$

[0036] 其中 S_P 表示网页的相似性矩阵，α'、β' 和 γ' 是加权值，其中 $\alpha'+\beta'+\gamma'=1$。

[0037] 因此相似性系统使用统一的结构来整理异类目标和它们的中间型关系。由于整体相似性函数是递归的，所以相似性系统同时并且反复地求解相似性函数。相似性函数由下面的方程式来表示：

[0038] $$S_{QC}[a,b]=\frac{Keyword(a)\cap Keyword(b)}{Keyword(a)\cup Keyword(b)}$$

[0039] $$S_{QCT}[a,b]=\frac{C_{CT}}{|M_{CT}(a)||M_{CT}(b)|}\sum_{i=1}^{|M_{CT}(a)||M_{CT}(b)|}\sum_{j=1}^{}S_P[M_{CT}^{i}(a),M_{CT}^{j}(b)]$$

[0040] $S_Q[a,b] = \alpha S_{QC}[a,b]+\beta S_{QCT}[a,b]$

[0041] $$S_{OL}[A,B]=\frac{C_{PC}}{|M_{OL}(A)||M_{OL}(B)|}\sum_{i=1}^{|M_{OL}(A)||M_{OL}(B)|}\sum_{j=1}^{}S_P[M_{OL}^{i}(A),M_{OL}^{j}(B)] \qquad (10)$$

[0042] $$S_{IL}[A,B]=\frac{C_{PR}}{|M_{IL}(A)||M_{IL}(B)|}\sum_{i=1}^{|M_{IL}(A)||M_{IL}(B)|}\sum_{j=1}^{}S_P[M_{IL}^{i}(A),M_{IL}^{j}(B)]$$

[0043] $$S_{PCT}[A,B]=\frac{C_{CT}}{|M_{CT}(A)||M_{CT}(B)|}\sum_{i=1}^{|M_{CT}(A)||M_{CT}(B)|}\sum_{j=1}^{}S_Q[M_{CT}^{i}(A),M_{CT}^{j}(B)]$$

[0044] $S_P[A,B] = \alpha' S_{OL}[A,B]+\beta' S_{IL}[A,B]+\gamma' S_{PCT}[A,B]$

[0045] 在方程式 10 中可以看到，任何两个查询之间的中间型相似性受网页的相似性的影响，内部型也是一样。由于网页的中间型相似性受查询的的相似性的影响，内部型也是一样，所以方程式 10 定义了递归的关系。因此，网页和查询的相似性相互影响并且收敛到一个稳定的状态。

[0046] 图 1 是描述一个实施例中的相似性系统的元件的框图。网站 101 经由通信链路 102 链接到相似性系统 110。相似性系统包括计算相似性元件 111、定义相似性函数元件 112、建立关系元件 113 和求解相似性函数元件 114。计算相似性元件基于中间型关系其它类型的目标的相似性来计算目标之间的相似性。计算相似性元件调用定义相似性函数元件、建立关系元件和求解相似性函数元件。定义相似性函数元件可以与用户交互来定义目标的类型、目标之间的关系和每种类型的目标的各种相似性函数。建立关系元件基于收集的数据生成关系数据。例如收集的数据可以包括查询、查询结果的网页和查询标记。求解相似性函数元件反复计算定义的相似性函数以生成更新的相似性矩阵，直到相似性矩阵收敛于一解值。

[0047] 能实现相似性系统的计算设备可以包括中央处理单元、存储器、输入装置（例如键盘和点击装置）、输出装置（例如显示装置）和存储装置（例如硬盘驱动器）。存储器和存储装置是计算机可读介质，其包含实现相似性系统的指令。另外，数据结构和消息结构可以经由数据传输介质来存储或发送，例如通信链路上的一个信号。可以使用各种通信链路，例如因特网、局域网、广域网或点对点呼叫连接。

[0048] 相似性系统可以在各种操作环境下实现。适合使用的各种公知的计算系统、环境和配置包括个人计算机、服务器计算机、手持或膝上型设备、多处理器系统、微处理器系统、可编程用户电子装置、网络 PC、小型机、大型计算机、包括任何上述系统或装置的分布式计算环境等。

[0049] 可以结合由一个或多个计算机或其它装置来执行的计算机可执行指令来描述相

似性系统，例如程序模块。通常，程序模块包括执行特殊任务或实现特殊抽象数据类型的例行程序、程序、目标、分量、数据结构等。一般来说，程序模块的参数可以按需要在各种实施方式中进行组合和分布。

[0050] 图 2 是描述一个实施例中定义相似性函数元件的处理流程图。在方框 201-209 中，所述元件循环选择目标的每种类型并且为那种类型的目标定义内部型和中间型相似性函数。在一个实施例中，所述元件可以与用户交互来定义目标之间的内部型和中间型关系。所述元件还可以定义不是递归地基于目标之间的相似性的相似性函数，例如基于查询的搜索项的相似性。在方框 201 中，所述元件选择目标的下一种类型。在决定方框 202 中，如果所有类型的目标都已经被选择，那么所述元件就返回，否则所述元件就继续到方框 203。在方框 203 中，所述元件为选定的类型选择下一个内部型关系。在决定方框 204 中，如果所有的内部型关系都已经被选择，那么所述元件就继续到方框 206，否则所述元件继续到方框 205。在方框 205 中，所述元件为选定类型和关系定义内部型相似性函数。所述元件然后循环到方框 203 来选择下一个内部型关系。在方框 206 中，所述元件为选定的类型选择下一个中间型关系。在决定方框 207 中，如果所有的中间型关系都已经被选择，那么所述元件就继续到方框 209，否则所述元件继续到方框 208。在方框 208 中，所述元件为选定的类型和关系定义中间型相似性函数。所述元件然后循环到方框 206 来选择下一个中间型关系。在方框 209 中，所述元件通过组合定义的内部型相似性函数和中间型相似性函数来为选定的类型定义整体的相似性函数。所述元件可以将加权因子应用到每一个组合的相似性函数中去。所述元件然后循环到方框 201 来选择下一种类型的目标。

[0051] 图 3 是描述一个实施例中建立关系元件的处理流程图。所述元件处理收集的数据并生成关系数据。在方框 301-308 中，所述元件循环选择每种类型的目标并为那种类型的目标生成关系数据。在方框 301 中，所述元件选择下一种目标。在决定方框 302 中，如果所有类型都已经被选择，那么所述元件就返回，否则所述元件继续到方框 303。在方框 303 中，所述元件为选定的类型选择下一个内部型关系。在决定方框 304 中，如果所有的内部型关系都已经被选择，那么所述元件就继续到方框 306，否则所述元件继续到方框 305。在方框 305 中，所述元件为选定的类型和选定的内部型关系设置关系数据的元素。所述元件然后循环到方框 303 来选择下一个内部型关系。在方框 306 中，所述元件为选定的类型选择下一个中间型关系。在决定方框 307 中，如果所有的中间型关系都已经被选择，那么所述元件就循环到方框 301 来选择下一种类型的目标，否则所述元件继续到方框 308。在方框 308 中，所述元件为选定的类型和选定的中间型关系设置关系数据的元素。所述元件然后循环到方框 306 为选定的类型选择下一个中间型关系。

[0052] 图 4 是描述一个实施例中求解相似性函数元件的处理流程图。在方框 401 中，所述元件初始化相似性矩阵。例如，所述元件可以将对角线的相似性值设置成一个表示最大相似性的值并且将其它相似性值设置为随机数。在方框 402 中，所述元件将误差值设置成一个非常大的数字以便执行至少一个重复过程。在方框 403-408 中，所述元件循环多次重复计算整体相似性函数以更新相似性矩阵，直到相似性值收敛于一解值。在方框 403 中，所述元件选择下一个重复过程，在决定方框 404 中，如果该类型的相似性误差值的总和小于门限误差值，那么所述解值收敛，所述元件返回，否则所述元件继续到方框 405。在方框 405 中所述元件选择下一种类型的目标。在决定方框 406 中，如果所有类型都已经被选择，那么

所述元件就继续到方框 408，否则所述元件继续到方框 407。在方框 407 中，所述元件为选定的类型计算相似性函数来为选定的类型更新相似性矩阵，然后循环到方框 405 来选择下一种类型。在方框 408 中，所述元件为选定的类型计算这个重复过程的相似性值与前一个重复过程的相似性值之间的误差。所述元件然后循环到方框 403 来开始下一个重复过程。

[0053] 图 5 是描述一个实施例中计算相似性函数元件的处理流程图。所述元件传送一种类型的目标并且为该类型更新相似性矩阵。在方框 501 中，所述元件为所传送的类型选择下一个内部型相似性函数。在决定方框 502 中，如果所有的内部型相似形功能都已经被选择，那么所述元件就继续到方框 504，否则所述元件就继续到方框 503。在方框 503 中，所述元件为所传送的类型的每一个目标计算一个新的相似性值。然后所述元件循环到方框 501 来选择下一个内部型相似性函数。在方框 504 中，所述元件为所传送的类型选择下一个中间型相似性函数。在决定方框 505 中，如果所有的中间型相似性函数都已经被选择，那么所述元件就继续到方框 507，否则所述元件就继续到方框 506。在方框 506 中，所述元件用选定的中间型相似性函数为所传送的类型的每一个目标计算新的相似性值。然后所述元件循环到方框 504 来选择下一个中间型相似性函数。在方框 507 中，所述元件使用加权来组合矩阵以生成当前重复过程的所传送的类型的整体相似性。然后所述元件返回。

[0054] 本领域的熟练技术人员会明白，尽管这里为了说明的目的而描述了相似性系统的特定实施例，但是在不脱离本发明精神和范围的情况下可以作各种修改。因此，除了附加的权利要求之外，对本发明不作限制。

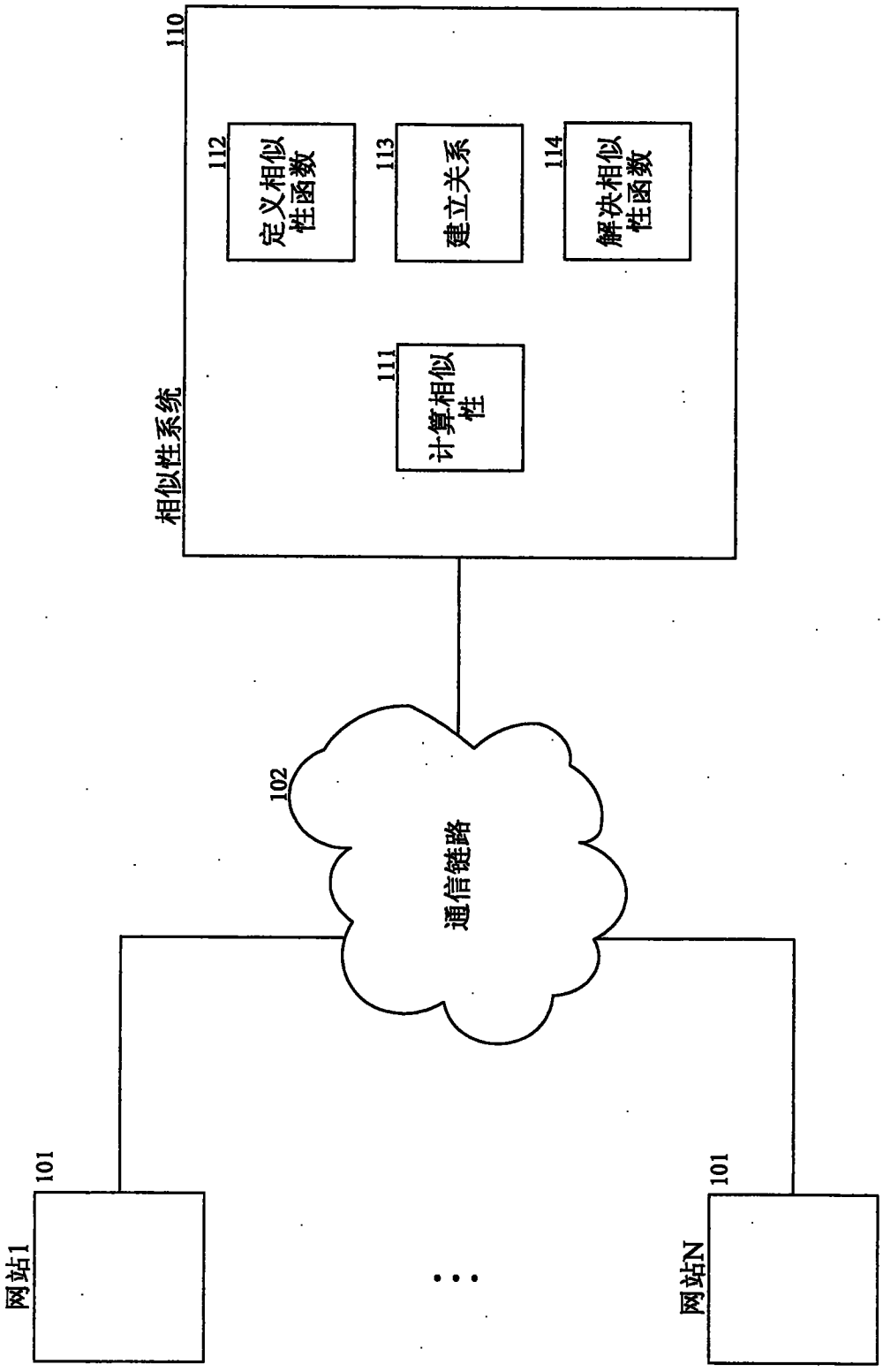

110
112
定义相似性函数
113
建立关系
114
解决相似性函数
111
计算相似性
相似性系统
102
通信链路
101
网站1
...
101
网站N

图 1

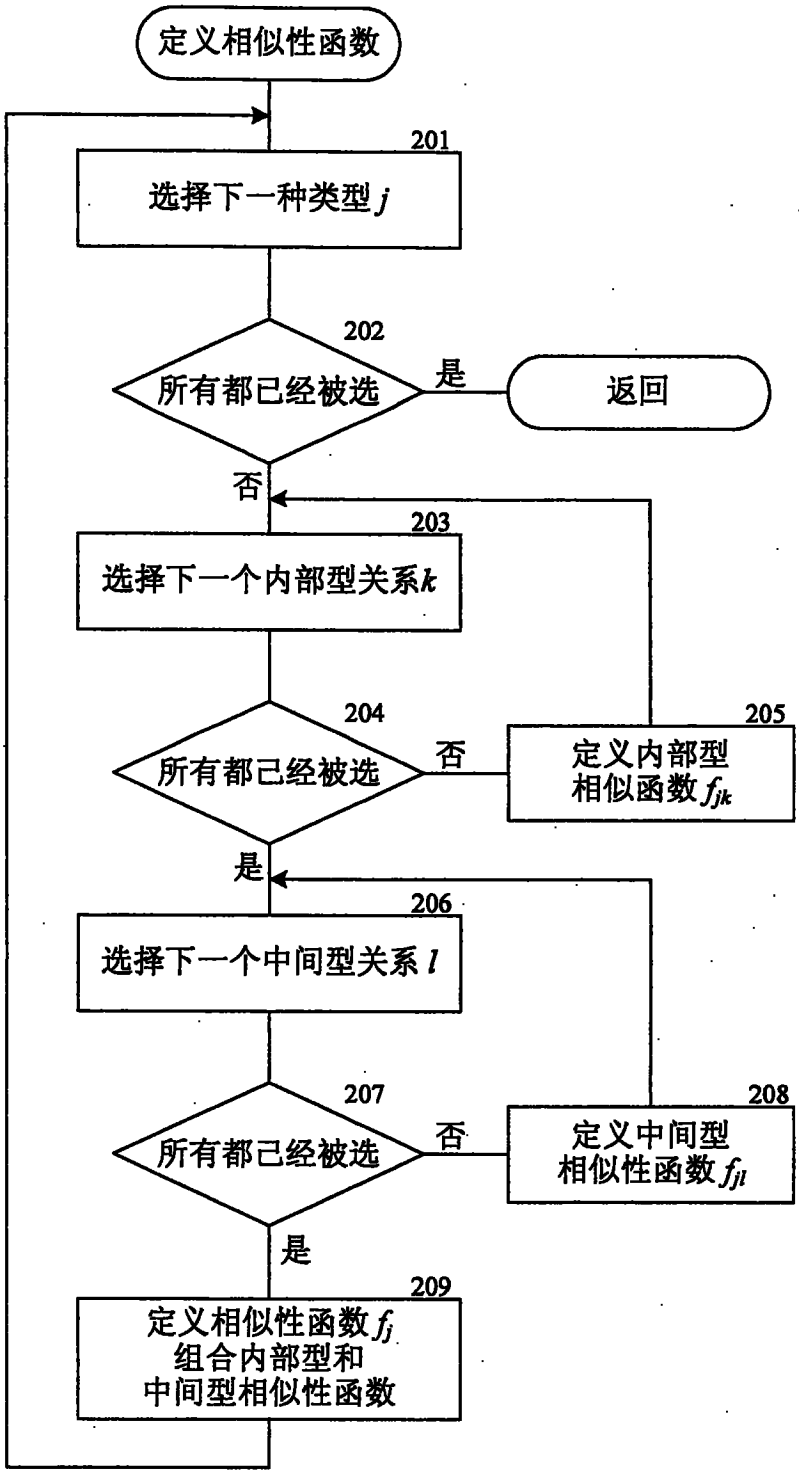

定义相似性函数
201
选择下一种类型 j
202
所有都已经被选
是
返回
否
203
选择下一个内部型关系k
204
所有都已经被选
否
205
定义内部型
相似函数 f_{jk}
是
206
选择下一个中间型关系 l
207
所有都已经被选
否
208
定义中间型
相似性函数 f_{jl}
是
209
定义相似性函数 f_j
组合内部型和
中间型相似性函数

图 2

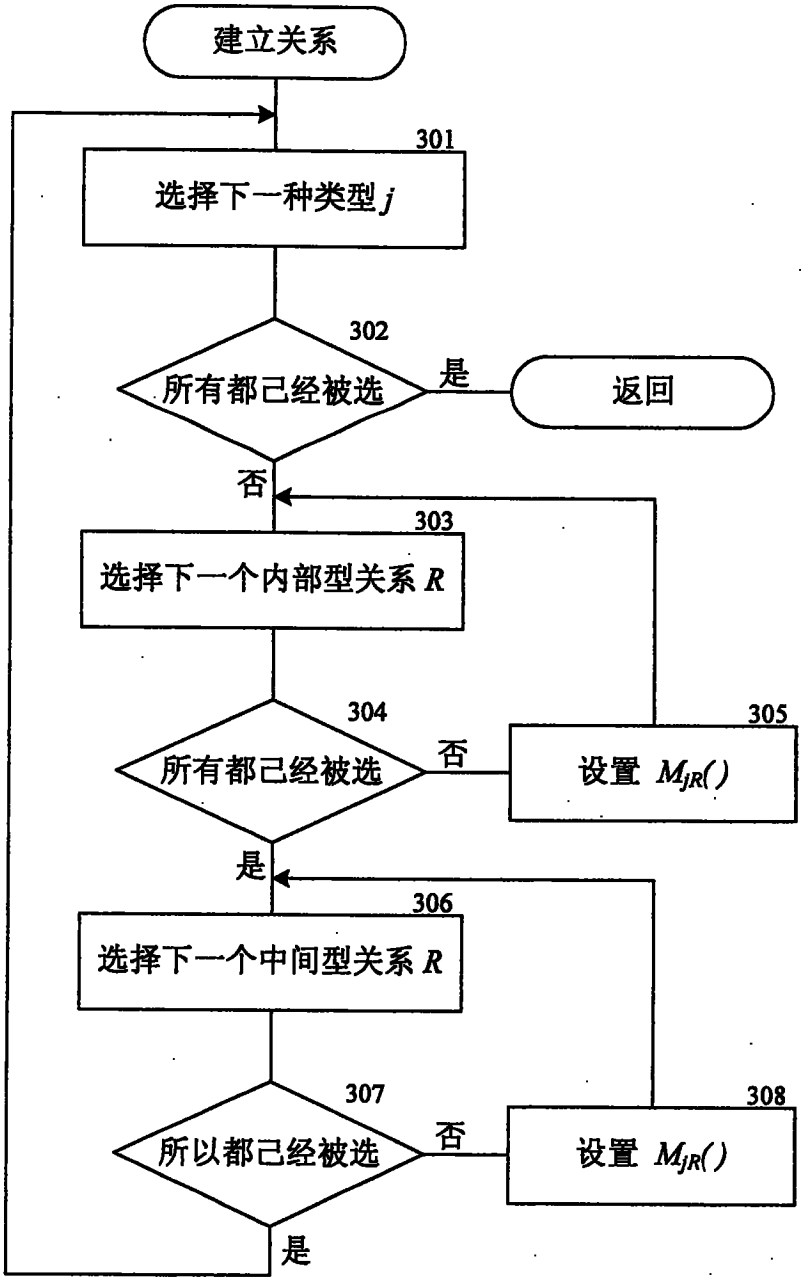

建立关系
301
选择下一种类型 j
302
所有都已经被选
是
返回
否
303
选择下一个内部型关系 R
304
所有都已经被选
否
305
设置 $M_{jR}()$
是
306
选择下一个中间型关系 R
307
所以都已经被选
否
308
设置 $M_{jR}()$
是

图 3

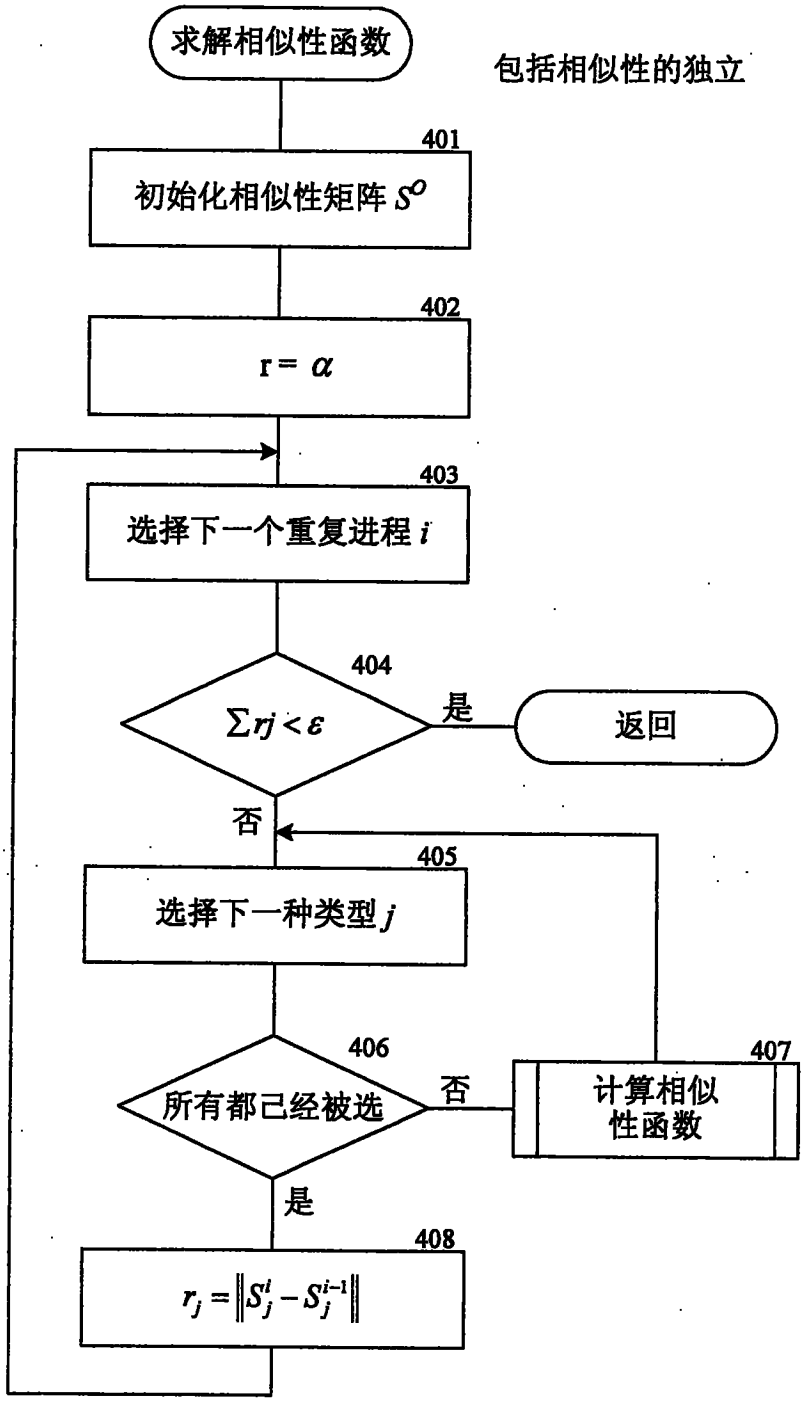

求解相似性函数
包括相似性的独立
401
初始化相似性矩阵 S^0
402
r = α
403
选择下一个重复进程 i
404
$\Sigma rj < \varepsilon$
是
返回
否
405
选择下一种类型 j
406
所有都已经被选
否
407
计算相似性函数
是
408
$r_j = \|S_j^i - S_j^{i-1}\|$

图 4

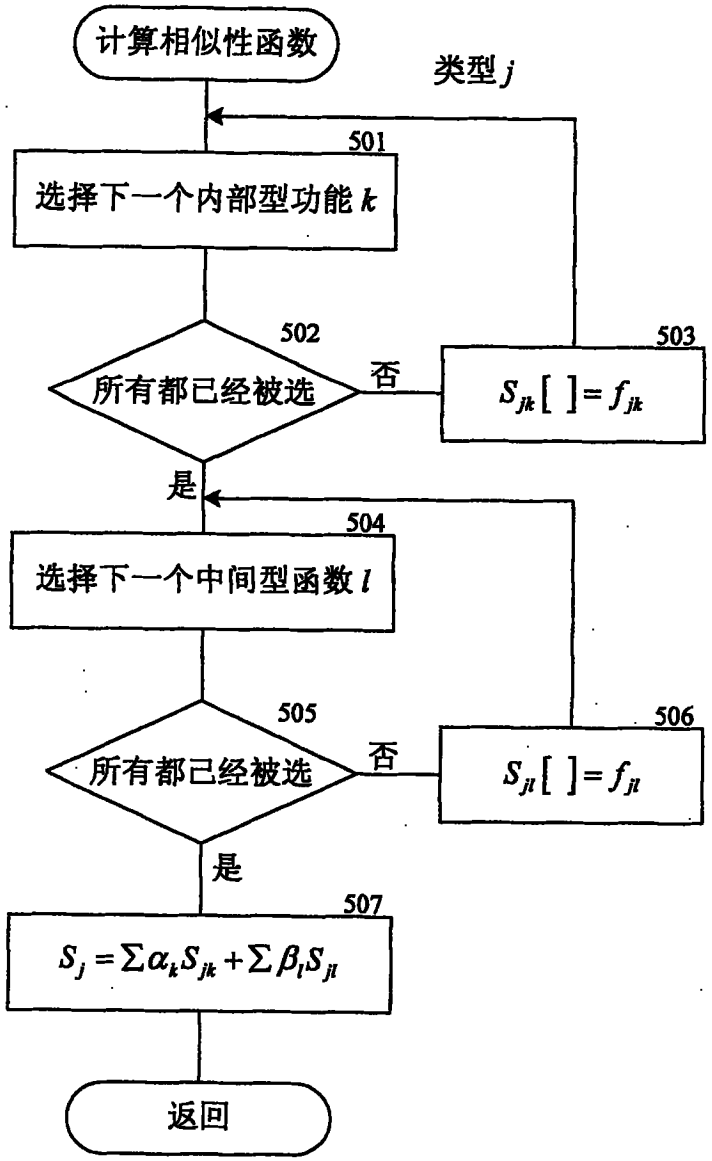

计算相似性函数
类型 j
501
选择下一个内部型功能 k
502
所有都已经被选
否
503
$S_{jk}[\ ]=f_{jk}$
是
504
选择下一个中间型函数 l
505
所有都已经被选
否
506
$S_{jl}[\ ]=f_{jl}$
是
507
$S_j=\sum\alpha_k S_{jk}+\sum\beta_l S_{jl}$
返回

图 5