(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 103177123 A
(43)申请公布日 2013.06.26

(21)申请号 201310129209.3

(22)申请日 2013.04.15

(71)申请人 昆明理工大学
地址 650093 云南省昆明市五华区学府路253号

(72)发明人 刘秉国 刘明 彭金辉 刘晨辉 张利波 何广军

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

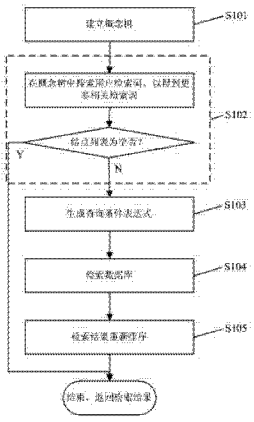
权利要求书1页 说明书6页 附图2页

(54)发明名称

一种提高数据库检索信息相关度的方法

(57)摘要

本发明提供了一种提高数据库检索信息相关度的方法，该方法包括如下步骤：(A)建立概念树；(B)在概念树中搜索用户检索词，以得到更多相关检索词；(C)生成查询条件表达式；(D)检索数据库；(E)检索结果重新排序。本发明的有益效果是：1)提高了关系型数据库信息检索的相关度，节省了用户查找信息的时间；2)结合概念树，进一步量化相关度，引入相关度参数，为用户检索信息时，在精确与模糊之间提供了更多的选项，更加方便了用户的使用；3)利用检索词及概念之间的关系动态计算权值，再利用动态权值重新进行排序，进一步提升用户体验；4)整个检索在单条语句中完成，有效减少了与数据库系统的交互，提高了执行效率。



CN 103177123 A

1. 一种提高数据库检索信息相关度的方法，其特征在于，包括如下步骤：

(A)建立概念树：利用所属领域的概念间内在的关联因素建立概念树，所述概念树分为多层，第一层为根结点，除根结点外，概念树中的每一结点至少包括一个或一个以上的数据库表中检索字段的值以及权值；

(B)在概念树中搜索用户检索词，以得到更多相关检索词：当获取用户输入的检索词后，在所述概念树中按照一定的策略搜索检索字段的值，如果存在该结点检索字段的值与检索词相匹配，则按规则将该结点及其相关结点插入一个结点列表中，完成搜索后，返回该结点列表；

(C)生成查询条件表达式：若返回的结点列表不为空，则顺序遍历结点列表中的结点，将检索字段的值与检索词相匹配的字段按“字段名 = 字段值”组成表达式，多个表达式之间用逻辑连词“OR”进行连接，当遍历完整个结点列表后，生成一个完整的 SQL 查询条件表达式，执行下一步骤(D)，若返回的结点列表为空，则返回空的最终的检索结果；

(D)检索数据库：根据生成的查询条件表达式，进一步生成完整的 SQL 查询语句，提交给数据库进行检索并返回结果集；

(E)检索结果重新排序：对结果集在内存中按结点列表中结点的权值重新进行排序，并返回最终的检索结果，清空结点列表。

2. 根据权利要求 1 所述的提高数据库检索信息相关度的方法，其特征在于，步骤(B)中，若概念树为 n 层，为提高检索信息的相关度，所述搜索策略为：先搜索第 n 层，即先搜索叶结点，若搜索完第 n 层，有检索字段的值与检索词相匹配的结点，按规则插入结点列表，并返回结点列表，结束搜索；若没有搜索到，则继续搜索第 n-1 层，以此类推，直到搜索到第 1 层的根结点，则表明搜索失败，返回空的结点列表。

3. 根据权利要求 1 所述的提高数据库检索信息相关度的方法，其特征在于，步骤(B)中，为量化相关度，依据概念树的层数，引入相关度参数，在所述结点列表中插入搜索到的结点及其相关结点的规则是通过相关度参数来决定的：

相关度参数 r，概念树的层数为 n，相关度参数 r 的取值范围为：$1<r\leqslant n$，当在第 i 层搜索到结点，其相关度参数 $r\geqslant i$，则将该结点及其子孙结点插入所述结点列表中，对叶结点，仅插入该叶结点到所述结点列表中；若 $r<i$，则将该结点在第 i 层的父结点及其父结点的所有子孙结点插入所述结点列表中。

4. 根据权利要求 1 所述的提高数据库检索信息相关度的方法，其特征在于，在步骤(B)中，所述概念树的结点的权值是在每次搜索到结点后，插入结点列表时应用权值计算的算法来动态计算的，对检索到的结点的权值要增加权值，而对其相关结点的权值则要降低。

一种提高数据库检索信息相关度的方法

技术领域

[0001] 本发明涉及一种提高数据库检索信息相关度的方法，属数据库检索技术领域。

背景技术

[0002] 在许多专业领域数据库系统中，往往存在大量的专业术语或专业名词的记录，如在冶金领域的矿物介电常数数据库所涉及的矿物名称的记录非常多，而在大量记录的数据库中，如何根据用户的带有专业术语的检索词有效的检索出更多相关的信息也是各种专业领域数据库系统的一个重要研究方向。

[0003] 通常在数据库中检索矿物介电常数记录时，会根据矿物的名称来进行检索。如检索“硫铁矿”，采用常规的数据库检索方法，可以应用 SQL（Structured Query Language)语句来检索数据库的表(Table）中的某个字段(Field）的值为某一指定的值，例如：“SELECT 矿物名称，介电常数 FROM 矿物介电常数表 WHERE 矿物名称 = ‘钛铁矿’”这样的语句来检索数据库。这种检索方式往往检索到单一的信息，而无法得到与“钛铁矿”相关联的矿物，如“锰铁矿”、“硫铁矿”的介电常数信息。这种方式下，用户往往需要多次输入检索词才能检索到所需要的信息，而且每次的显示结果都是单一的，不能将一些用户需要的信息整合到一起显示，以便用户对比研究。目前很多检索系统是通过提供高级检索方式，由用户输入多个检索词以构造检索语句来实现，而这种方式也需要用户录入较多的检索词，很不方便用户的使用。

[0004] 为了得到更多的信息，对单一检索词，通常的做法就是将 SQL 语句的条件表达式改为模糊检索方式，并将检索词进行拆分，如：对“硫铁矿”可以拆分成条件表达式“矿物名称 LIKE ‘%硫%’”、“矿物名称 LIKE ‘%铁%’” 和“矿物名称 LIKE ‘%矿%’”，然后将这些条件表达式构造为检索语句在数据库中进行检索，最后将检索结果通过 UNION 连词合并起来。这种检索方式又将会检索出大量的与用户的期望不相关的信息，用户需要花费大量的时间来自行筛选和判断，也就是说，信息量很大，但相关度却很低。另一种方法就是采用分词技术来解析检索词，如将“硫铁矿”解析为“硫铁矿”和“铁矿”，而将“锰铁矿”解析为“锰铁矿”和“铁矿”，然后再进行模糊检索。但是很多专业术语并没有统一的规律可循，如“钛精矿”，如果解析为“钛精矿”和“精矿”显然不合适。同时，使用模糊检索方式，数据库系统在检索时将会扫描文本性字段，如果过多使用模糊检索将会导致系统的检索性能下降。

发明内容

[0005] 针对上述问题，本发明提供了一种提高数据库检索信息相关度的方法，包括如下步骤：

(A)建立概念树：利用所属领域的概念间内在的关联因素建立概念树，所述概念树分为多层，第一层为根结点，除根结点外，概念树中的每一结点至少包括一个或一个以上的数据库表中检索字段的值以及权值；

(B)在概念树中搜索用户检索词，以得到更多相关检索词：当获取用户输入的检索词

后，在所述概念树中按照一定的策略搜索检索字段的值，如果存在该结点检索字段的值与检索词相匹配，则按规则将该结点及其相关结点插入一个结点列表中，完成搜索后，返回该结点列表；

（C）生成查询条件表达式：若返回的结点列表不为空，则顺序遍历结点列表中的结点，将检索字段的值与检索词相匹配的字段按“字段名 = 字段值”组成表达式，多个表达式之间用逻辑连词“OR”进行连接。当遍历完整个结点列表后，生成一个完整的 SQL 查询条件表达式，执行下一步骤（D），若返回的结点列表为空，则返回空的最终的检索结果；

（D）检索数据库：根据生成的查询条件表达式，进一步生成完整的 SQL 查询语句，提交给数据库进行检索并返回结果集；

（E）检索结果重新排序：对结果集在内存中按结点列表中结点的权值重新进行排序，并返回最终的检索结果，清空结点列表。

[0006] 步骤（B）中，若概念树为 n 层，为提高检索信息的相关度，所述搜索策略为：先搜索第 n 层，即先搜索叶结点，若搜索完第 n 层，有检索字段的值与检索词相匹配的结点，按规则插入结点列表，并返回结点列表，结束搜索；若没有搜索到，则继续搜索第 n-1 层，以此类推，直到搜索到第 1 层的根结点，则表明搜索失败，返回空的结点列表。

[0007] 步骤（B）中，为量化相关度，依据概念树的层数，引入相关度参数，在所述结点列表中插入搜索到的结点及其相关结点的规则是通过相关度参数来决定的：

相关度参数 r，概念树的层数为 n，相关度参数 r 的取值范围为：$1<r\leqslant n$，当在第 i 层搜索到结点，其相关度参数 $r\geqslant i$，则将该结点及其子孙结点插入所述结点列表中，对叶结点，仅插入该叶结点到所述结点列表中；若 $r<i$，则将该结点在第 i 层的父结点及其父结点的所有子孙结点插入所述结点列表中。

[0008] 步骤（B）中，所述概念树的结点的权值是在每次搜索到结点后，插入结点列表时应用权值计算的算法来动态计算的，对检索到的结点的权值要增加权值，而对其相关结点的权值则要降低。

[0009] 本发明的有益效果是：

1）本发明的基本思路是利用领域概念之间的某种内在的关联关系建立概念树，将用户检索词使用概念树制导，得到更多的与用户检索词相关联的检索词，利用“OR”逻辑连接词生成条件表达式，进而提高传统关系型数据库检索的相关性，实际应用表明，本发明大幅提高了传统关系数据库信息检索的相关度，节省了用户查找信息的时间；

2）结合概念树，进一步量化相关度，引入相关度参数，为用户检索信息时，在精确与模糊之间提供了更多的选项，更加方便了用户的使用。

[0010] 3）利用检索词及概念之间的关系动态计算权值，再利用动态权值重新进行排序，使得系统能够更好的“感知”用户，进一步提升用户体验；

4）整个检索仅在单条查询语句中完成，而且不包括任何子查询，便于数据库系统执行查询优化的同时，还有效减少了与数据库系统的交互和减少了数据库系统的负担，提高了整个系统的执行效率；

5）提高检索信息的相关度的同时，避免了使用 Like 操作符，提高了数据库系统的检索速度。

[0011] 附图说明：

图 1 是本发明的一种提高数据库检索信息相关度的方法的主要流程图；

图 2 是本发明的实施例的矿物概念树的示意图。

[0012]

具体实施方式：

为了使技术人员对本发明的目的、优点更加明白，下面结合附图对本发明做进一步说明。

[0013] 如图 1 所示，为本发明所述方法的主要流程。该方法包括：

步骤 S101：建立概念树，利用所属领域的概念间内在关联因素建立概念树，所述概念树分为多层，第一层为根结点，除根结点外，概念树中的每一结点至少包括权值以及一个或一个以上的数据库表中被检索的字段的值；

概念树构造为：第一层为根结点，第二层为金属元素，第三层为化合物，第四层为矿物，金属元素结点的子结点为含该金属元素的化合物，化合物结点的子结点为含该化合物的矿物，除根结点外，每个结点的信息除包括矿物名称的字段值外，还包括分子式、品位和权值，矿物结点的品位为该结点的矿物所含其父结点的化合物的含量，为百分比，其余结点的品位均设置为 1。

[0014] 步骤 S102：在概念树中搜索与用户检索词相关联的检索词，当获取用户输入的检索词后，在所述概念树中搜索字段值，如果存在该结点的字段值与检索词相匹配，则按规则将该结点及其相关结点插入一个结点列表中，继续执行搜索，直到搜索完整个概念树，并返回结点列表；

若所述概念树为 n 层，为提高检索信息的相关度，所述搜索策略为：先搜索第 n 层，即先搜索叶结点，若搜索完第 n 层，有检索字段的值与检索词相匹配的结点，按规则插入结点列表，并返回结点列表，结束搜索；若没有搜索到，则继续搜索第 n-1 层，以此类推，直到搜索到第 1 层的根结点，则表明搜索失败，返回空的结点列表。

[0015] 为量化相关度，依据概念树的层数，引入相关度参数，在所述结点列表中插入搜索到的结点及其相关结点的规则是通过相关度参数来决定的：

相关度参数 r，概念树的层数为 n，相关度参数 r 的取值范围为：$1<r\leqslant n$，当在第 i 层搜索到结点，其相关度参数 $r\geqslant i$，则将该结点及其子孙结点插入所述结点列表中，对叶结点，仅插入该叶结点到所述结点列表中；若 $r<i$，则将该结点在第 i 层的父结点及其父结点的所有子孙结点插入所述结点列表中。

[0016] 概念树的结点的权值是在每次搜索到结点后，插入结点列表时应用权值计算的算法来动态计算的，结点的权值是这样来计算的：当被检索到的结点的品位为 P，其权值 W，需要提高权值，则 W=P+1；而对其子结点、兄弟结点或父结点，需要降低权值，当某一结点的品位为 P_i，则该结点对应的权值为 $W_i=P_i\times P$。

[0017] 步骤 S103：生成查询条件表达式，若返回的结点列表不为空，则顺序遍历结点列表中的结点，将结点中字段值匹配检索词字段值按：“字段名 = 字段值”，生成表达式，多个表达式之间用逻辑连词“OR”进行连接。当遍历完整个结点列表后，生成一个完整的 SQL 查询条件表达式，执行下一步骤 S104，若返回的结点列表为空，则返回空的最终的检索结果；

步骤 S104：检索数据库，根据生成的查询条件表达式，进一步生成完整的 SQL 查询语

句，提交给数据库进行检索并返回结果集；

步骤 S105：检索结果重新排序，对结果集在内存中按结点列表中结点的权值重新进行排序，并返回最终的检索结果，清空结点列表。

[0018] 下面结合具体实例，从开发及应用的角度，对本发明再做详细说明。

[0019] 如表 1 所示，为本实施例的数据库中的矿物介电常数表，主要包括：ID、矿物名称、介电常数、介电损耗和品位等字段，ID 为主键。表 2 为矿物成份表，包括：ID、化合物、中文名、百分比、品位标记等字段，其中，ID 与化合物为矿物成分表的联合主键。矿物成份表的 ID 是矿物介电常数表的外键。

ID	矿物名称	介电常数	介电损耗	品位
……	……	……	……	……
10011	氧化铁矿	1.5921	0.235	30%
10012	高钛渣	10.163	0.559	40%
10013	菱铁矿	2.097	0.025	35%
10014	二氧化硅	2.3771	0.16449	95%
10015	钛铁矿	6.2327	0.332	37%
10016	磁铁矿	33.7	0.053	56%
……	……	……	……	……

[0020] 表 1

ID	化合物	中文名	百分比	品位标记
10011	Fe_2O_3	三氧化二铁	30%	√
10011	杂质	杂质	70%	
……	……	……	……	……
10016	Fe_3O_4	四氧化三铁	56%	√
……	……	……	……	……

表 2

概念树的建立可以通过相关软件来编辑建立，也可以通过程序方式来自动建立，计算机根据被检索的字段的记录在概念上的某个内在关联因素可以进一步建立概念树，在冶金领域，矿物间的一个主要关联因素就是化学元素、化合物和矿物成份，本实施例中，名采用“矿物名称”作为检索字段，并采用了一种自动建立概念树的方法：

a）建立金属元素列表；

b）从矿物成份表及矿物介电常数表中，通过关联查询得到化合物及矿物列表；

c）进一步的，先建立概念树的根结点；

d）由金属元素表生成第二层金属元素结点，其检索字段“矿物名称”的值为：铁、钛等金属元素中文名，对应分子式为 Fe、Ti，品位均设置为 1，权值默认设为 1；

e）根据化合物与矿物列表，判断化合物中是否包含第二层的金属元素，进一步生成第三层的化合物结点，如：三氧化二铁 Fe_2O_3 中，含有铁 Fe，则三氧化二铁 Fe_2O_3 为其子结点，其检索字段“矿物名称”的值为：三氧化二铁、四氧化三铁等化合物的中文名，对应分子式为 Fe_2O_3、Fe_3O_4 等，品位均设置为 1，权值默认设为 1；

f）根据化合物及矿物列表，判断第三层的化合物是否包括矿物，进而得到第四层的矿

物结点，如磁铁矿中含化合物Fe_3O_4，则磁铁矿为四氧化三铁Fe_3O_4的子结点，其检索字段“矿物名称”的值为：“磁铁矿”等矿物的中文名，对应分子式为其主要化合物的分子式Fe_3O_4，品位为其所含化合物Fe_3O_4的含量56%，权值默认设为1；

经过以上过程，可以得到如图2所示的矿物概念树。

[0021] 实际应用中，相关度参数r可设置为3。用户输入检索词，如“黄铁矿”，计算机首先在概念树的第四层搜索矿物结点，当搜索到检索字段“矿物名称”的值为“黄铁矿”的结点，相关度参数r设置为3，而检索到的结点的层数i为4，满足r<i，则对该结点及其兄弟结点和父结点计算权值并插入结点列表中，直到搜索完第四层的所有结点，此时结点列表中有三个结点，分别是“黄铁矿”“白铁矿”和“二硫化铁”，结束搜索。

[0022] 下一步就是根据结点列表生成查询条件表达式，计算机根据上述结点列表中的三个结点和检索字段名“矿物名称”进一步生成查询条件表达式为：“矿物名称 = ‘黄铁矿’ OR 矿物名称 = ‘白铁矿’ OR 矿物名称 = ‘二硫化铁’”。

[0023] 接下来计算机将事先设定的SELEC语句：“SELECT 矿物名称，品位，介电常数，介电损耗 FROM 矿物介电常数表”和查询表达式装配起来，生成完整的查询表达式：

“SELECT 矿物名称，品位，介电常数，介电损耗 FROM 矿物介电常数表 WHERE 矿物名称 = ‘黄铁矿’ OR 矿物名称 = ‘白铁矿’ OR 矿物名称 = ‘二硫化铁’”该SQL语句提交给数据库系统在数据库中检索。

[0024] 当用户输入检索词为“黄铁矿”，计算机在向结点列表中插入结点时，根据权值计算规则计算权值，“黄铁矿”权值为1+0.422=1.422，“白铁矿”权值为0.422*0.412=0.174，经过数据库系统的检索，得到了检索结果集，计算机根据权值按降序排列并将最终的检索结果展示给用户，得到如表3所示的检索结果：

矿物名称	品位	介电常数	介电损耗
黄铁矿	42.20%	33.11	0.062
白铁矿	41.20%	32.12	0.064

表3

当用户输入检索词为“白铁矿”，则“白铁矿”权值为1+0.412=1.412，“黄铁矿”权值为0.422*0.412=0.174，计算机对检索结果集根据权值按降序排列，则得到表4所示的检索结果：

矿物名称	品位	介电常数	介电损耗
白铁矿	41.20%	32.12	0.064
黄铁矿	42.20%	33.11	0.062

表4

也就是说，系统实现了根据用户的检索词来动态排序，使得系统能够更好的“感知”用户，进一步提升了用户体验。

[0025] 实际应用中，“钛铁矿”既含铁，也含钛，因此，钛铁矿在概念树中出现了多处，又如多数“镁钛矿”中也含有一定的$FeTiO_3$，在本实施例中，“镁钛矿”也归入化合物$FeTiO_3$的子

结点，“镁钛矿”就。当用户输入检索词“钛铁矿”，相关度参数设置为 r=3，计算机在概念树的第四层搜索矿物结点，直到搜索完所有的矿物结点，将得到两处“钛铁矿”，在插入结点列表时，同名的矿物仅插入一个结点，则最后生成的查询语句为：

“SELECT 矿物名称，品位，介电常数，介电损耗 FROM 矿物介电常数表 WHERE 矿物名称 = ‘钛铁矿’ OR 矿物名称 = ‘镁钛矿’ OR 矿物名称 = ‘$FeTiO_3$’”。

[0026] 通过检索，可以得到“钛铁矿”及与之相关的“镁钛矿”的介电常数的信息，将检索结果展示给用户。

[0027] 如果相关度参数设置为 r=2，满足 r<i，如图 2 所示的概念树中，“钛铁矿”的第 2 层的父结点有“钛”和“铁”，按规则，计算机将会把“钛”和“铁”的所有子结点，包括化合物结点和矿物结点均插入结点列表中，最后生成的 SQL 语句为：

“SELECT 矿物名称，品位，介电常数，介电损耗 FROM 矿物介电常数表 WHERE 矿物名称 = ‘钛’ OR 矿物名称 = ‘二氧化钛’ OR 矿物名称 = ‘$FeTiO_3$’ OR 矿物名称 = ‘钛铁矿’ OR 矿物名称 = ‘镁钛矿’ OR 矿物名称 = ‘高钛渣’OR 矿物名称 = ‘钛精矿’ OR 矿物名称 = ‘铁’ OR “矿物名称 = ‘黄铁矿’ OR 矿物名称 = ‘白铁矿’ OR 矿物名称 = ‘二硫化铁’……”。

[0028] 其检索结果中包括“钛”和“铁”的所有矿物，也就是说，降低了相关度，检索信息模糊化，也得到了更多的检索信息。

如果相关度参数设置为 r=4，满足 r ≥ i，且为叶结点，则生成的 SQL 语句为：

“SELECT 矿物名称，品位，介电常数，介电损耗 FROM 矿物介电常数表 WHERE 矿物名称 = ‘钛铁矿’”。

[0029] 其检索结果仅得到“钛铁矿”的检索信息，提高了相关度，检索到的信息更精确，而信息量也减少。

[0030] 实际应用中，当用户输入检索词“钛”，计算机按照搜索规则，如图 2 所示，在第四层的矿物结点中搜索不到与“钛”精确匹配的矿物名称，则搜索第三层的化合物结点，到第二层的金属元素结点中搜索到与“钛”精确匹配的结点，若相关度参数 r 设置为 3，而检索到的结点的层数 i 为 2，满足 r ≥ i，按则将该结点及其子孙结点放入结点列表中，分别是“钛”、“二氧化钛”、“$FeTiO_3$”、“钛铁矿”、“镁钛矿”、“高钛渣”和“钛精矿”。最后生成的查询语句为：

“SELECT 矿物名称，品位，介电常数，介电损耗 FROM 矿物介电常数表 WHERE 矿物名称 = ‘钛’ OR 矿物名称 = ‘二氧化钛’ OR 矿物名称 = ‘FeTiO3’ OR 矿物名称 = ‘钛铁矿’ OR 矿物名称 = ‘镁钛矿’ OR 矿物名称 = ‘高钛渣’ OR 矿物名称 = ‘钛精矿’”

该 SQL 语句提交给数据库系统检索，可以得到“钛”及与之相关的各种矿物的介电常数的信息。在计算权值时，第二层金属元素和第三层化合物其品位值无实际意义，仅为计算排序的权值使用。第二层的金属“钛”的品位值设 1，第三层的化合物“二氧化钛”、“$FeTiO_3$”的品位值取 1，也可以取“钛”化合物中所占的比例，第四层的矿物结点的品位值则为该矿物实际的品位值，根据权值计算规则计算出其权值，并将结果集在内存中排序后得到最终的检索结果。检索结果展示给用户。

[0031] 本专利是通过具体实施过程进行说明的，在不脱离本专利范围的情况下，还可以对本专利进行各种变换及等同代替，因此，本专利不局限于所公开的具体实施过程，而应当包括落入本专利权利要求范围内的全部实施方案。

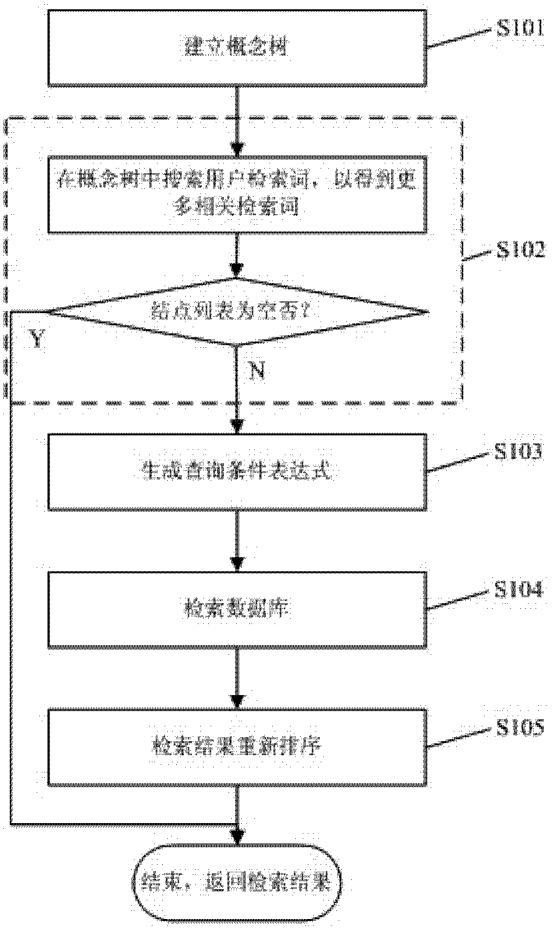

建立概念树
S101
在概念树中搜索用户检索词，以得到更多相关检索词
S102
结点列表为空否？
Y
N
生成查询条件表达式
S103
检索数据库
S104
检索结果重新排序
S105
结束，返回检索结果

图 1

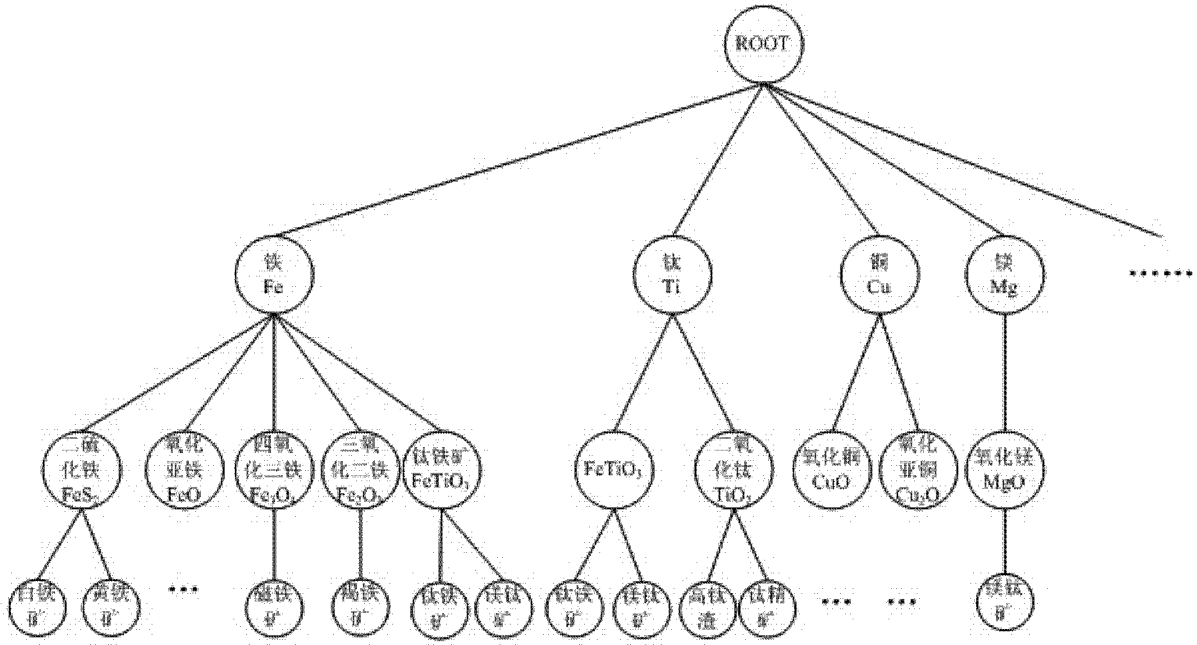

ROOT
铁
Fe
钛
Ti
铜
Cu
镁
Mg
二硫化铁 FeS_2
氧化亚铁 FeO
四氧化三铁 Fe_3O_4
三氧化二铁 Fe_2O_3
钛铁矿 $FeTiO_3$
$FeTiO_3$
二氧化钛 TiO_2
氧化铜 CuO
氧化亚铜 Cu_2O
氧化镁 MgO
白铁矿
黄铁矿
磁铁矿
褐铁矿
钛铁矿
铁钛矿
钛铁矿
铁钛矿
高钛渣
钛精矿
镁钛矿

图 2