

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国际局

(43) 国际公布日
2015年1月29日 (29.01.2015)



(10) 国际公布号
WO 2015/010509 A1

- (51) 国际专利分类号:
G06F 17/30 (2006.01)
- (21) 国际申请号: PCT/CN2014/080179
- (22) 国际申请日: 2014年6月18日 (18.06.2014)
- (25) 申请语言: 中文
- (26) 公布语言: 中文
- (30) 优先权:
201310288821.5 2013年7月3日 (03.07.2013) CN
- (71) 申请人: 深圳市华傲数据技术有限公司 (AUDAQUE DATA TECHNOLOGY CO., LTD.) [CN/CN]; 中国广东省深圳市南山区高新区中区高新中一道9号软件大厦713室, Guangdong 518057 (CN)。
- (72) 发明人: 贾西贝 (JIA, Xibei); 中国广东省深圳市南山区高新区中区高新中一道9号软件大厦713室, Guangdong 518057 (CN)。 王国印 (WANG, Guoyin); 中国广东省深圳市南山区高新区中区高新中一道9号软件大厦713室, Guangdong 518057 (CN)。
- (74) 代理人: 深圳鼎合诚知识产权代理有限公司 (DHC IP ATTORNEYS); 中国广东省深圳市福田区金田路

与福华路交汇处现代商务大厦 2201, Guangdong 518048 (CN)。

(81) 指定国 (除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

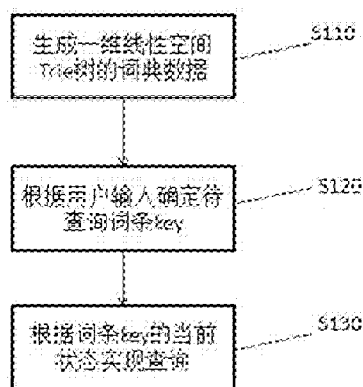
(84) 指定国 (除另有指明, 要求每一种可提供的地区保护): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

— 包括国际检索报告(条约第21条(3))。

(54) Title: ONE-DIMENSIONAL LINER SPACE-BASED METHOD FOR IMPLEMENTING TRIE TREE DICTIONARY SEARCH

(54) 发明名称: 一种基于一维线性空间实现 Trie 树的词典检索方法



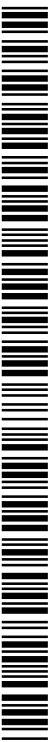
S110 GENERATION OF THE ONE-DIMENSIONAL LINEAR SPACE TRIE TREE DICTIONARY DATA
 S120 DETERMINATION OF THE TO-BE-QUERIED ENTRY KEY ON THE BASIS OF THE USER INPUT
 S130 IMPLEMENTATION OF THE QUERY ON THE BASIS OF THE CURRENT STATE OF THE ENTRY KEY

图 1 / FIG. 1

(57) Abstract: A one-dimensional liner space-based method for implementing trie tree dictionary search: one-dimensional liner space trie tree dictionary data is generated; a to-be-queried entry key is determined on the basis of a user input; and, a query is implemented on the basis of a current state of the entry key. In the trie tree dictionary data that is constructed in a one-dimensional linear space, dictionary loading and search speeds are increased, and rapid retrieval of all prefix terms of an entry is allowed. In addition, trie tree dictionary search implemented on the basis of one-dimensional linear space allows for solving of the problem of a conflict that is caused by insertion of a new state and is found in a process of trie tree construction of a conventional trie tree dictionary data search, thus allowing for prevention of the problem of a movement of a large amount of dictionary data caused by the conflict.

(57) 摘要:

[见续页]



WO 2015/010509 A1

一种基于一维线性空间实现 Trie 树的词典检索方法，通过生成一维线性空间的 Trie 树词典数据；根据用户输入确定待查询词条 key；根据词条 key 的当前状态实现查询。在一维线性空间下构建 Trie 树的词典数据下，提高了词典加载和检索速度，并且能够快速检索到词条的所有前缀词，另外，基于一维线性空间实现 Trie 树的词典检索能够解决传统 Trie 树词典数据检索在 Tire 树的构建过程中存在的因插入新状态而引起的冲突问题，并能避免该冲突导致的大量词典数据的移动问题。

一种基于一维线性空间实现 Trie 树的词典检索方法

技术领域

本发明涉及一种词典检索方法，尤其涉及一种基于一维线性空间实现 Trie 树的词典检索方法。

背景技术

在信息检索和自然语言处理领域，特别是基于词典的技术应用中，词典的规模一般都非常大，拥有成千上万甚至上亿条记录，尤其是搜索引擎中倒排索引词的规模最为庞大。对海量数据词典的查找，当前通常采用索引的数据结构来实现。常用的索引结构包括线性索引表、倒排表、散列（hash）表和搜索树等。对于一个记录的关键字（key）长度为 n ，词典的规模为 N 的情况下（其中 $N \gg n$ ）各索引结构检索的时间复杂度分析如下：

线性索引结构或倒排表一般采用顺序结构存储词典中的记录，对词典中记录的查找一般采用顺序遍历每一个记录，因此，每一次查找的时间复杂度为 $O(N) \times l(n)$ （其中 $l(n)$ 为两条记录的关键字因比较所花费的时间）。此种方法的改进是让词典的每一条记录按关键字（key）有序，在检索的时候用折半查找，其时间复杂度为 $O(\log N) \times l(n)$ 。

在基于词典的自然语言处理应用，如基于词典的中文分词、基于词典的字音转换（汉字转换成拼音）、基于词典的命名实体识别、基于词典的标注（包括词性标注和语义标注等），其中的核心部分是要进行大量的查询操作。为了满足此类应用的要求，就需要一种高效的词典检索方法。现今也有基于二维数组的 Trie 树的词典查询方法，但是基于二维数组的这种查询方法在 Trie 树的构建过程中会存在因插入新状态而引起的冲突，导致要移动大量存在冲突的数据问题。

发明内容

本发明旨在解决上述缺陷之一。

因此，本发明提供一种基于一维线性空间实现 Trie 树的词典检索方法，通过生成一维线性空间的 Trie 树词典数据；根据用户输入确定待查询词条 key；根据词条 key 的当前状态实现查询。在一维线性空间下构建 Trie 树的词典数据下，提高了词典加载和检索速度，并且能够快速检索到词条的所有前缀词，另外，基于一维线性空间实现 Trie 树的词典检索能够解决

传统 Trie 树词典数据检索在 Trie 树的构建过程中存在的因插入新状态而引起的冲突问题，并能避免该冲突导致的大量词典数据的移动问题。

为此，本发明公开了一种基于一维线性空间实现 Trie 树的词典检索方法，该方法包括以下步骤：生成一维线性空间 Trie 树的词典数据；根据用户输入确定待查询词条 key；根据词条 key 的当前状态实现查询。

优选地，词典数据的 key 转化为节点后存储在一维数组中，所述一维数组的值用来标识 base 值是否唯一。

优选地，所述一维线性空间的 Trie 树中将所有终端结点变为非终端结点，在所述终端结点后面增加一个叶子节点，并将叶子节点的 check 值赋上其存储位置。

优选地，所述叶子节点还包括：叶子节点的 base 值用以标识其是否为终端结点。

优选地，所述查询包括以下步骤：把当前结点指向根节点；根据当前输入的字符做出状态转移，获取其直接后继状态的位置；校验当前状态的前驱，确定当前状态由哪一状态转移而来；获取词条 key 对应的 value 值。

优选地，所述查询包括：词条 key 的查询可以获得其所有前缀词的结果。

本发明提供的一种基于一维线性空间实现 Trie 树的词典检索方法，通过生成一维线性空间的 Trie 树词典数据；根据用户输入确定待查询词条 key；根据词条 key 的当前状态实现查询。在一维线性空间下构建 Trie 树的词典数据下，提高了词典加载和检索速度，并且能够快速检索到词条的所有前缀词。同时，在 Trie 树的构建过程中通过调整 base 值，使得其所有直接后继均不会发生冲突，因此避免了数据移动或存储空间分配的回溯问题。

应当理解，以上总体说明和以下详细说明都是说明性和实例性的，旨在提供对所要求的本发明的进一步说明。

附图说明

图1是本发明实施例一种基于一维线性空间实现Trie树的词典检索方法的流程图。

图2是本发明实施例根据词条key的当前状态实现查询的流程框架图。

具体实施方式

为了使本发明的目的、技术方案及优点更加清楚明白，以下结合附图及实施例，对本发明进行进一步的详细说明。应当理解，此处所描述的具体实施例仅仅用于解释本发明，并不用于限定本发明。

本发明实施例提供的一种基于一维线性空间实现 Trie 树的词典检索方法。

如图 1 所示，是本发明实施例一种基于一维线性空间实现 Trie 树的词典检索方法的流程图。

步骤 S110：生成一维线性空间 Trie 树的词典数据。

获取词典数据，要生成一维线性空间 Trie 树的词典数据包括以下具体步骤：

步骤 S111：将所有词条和属性信息以 key 为中心按词典顺序排序，合并拥有相同 key 值的 values，要保证 key 不存在重复。

遍历 _Keys 和 _values 中存储的元素，按词典顺序将 Keys 和 _values 以 key 为中心排序，并合并相同 key 的 value 值，将有序的 key 序列和 value 序列存储到 List<String> keys 和 Collection<String> attributes 中；

该步骤的伪代码如下：

```
array[0] = 1;
Node root_node = new Node();
root_node.left = 0;
root_node.right = keys.size();
root_node.depth = 0;
ArrayList<Node> siblings = new ArrayList<Node>();
fetch(root_node, siblings);
insert(siblings);
return keys.size();
}
```

步骤 S112：定义起始状态，编号为 0，其包含的信息值为[code = 0, depth = 0, start = 0, end = N]，其中 N 为词典的规模，即 key 的数量。

步骤 S113：将起始状态放入双数组第 0 位置，将其 base[0]=1 (array[2*0]=array[0]=1)，并标识 base 为 1 的值已经被占用（保证所有状态的 base 值唯一），check[0]=0 (array[2*0+1]=array[1]=0)

步骤 S114：以起始状态作为当前状态。

步骤 S115：获取当前状态的所有直接后继状态的信息，若直接后继结点列表为空，即当前结点为终端结点“\$”，表示从起始结点到当前结点构成的 key 恰好是词典中的一个完整词条，将当前结点（终端结点）的 base 值赋上当前 key 词典顺序序号的相反数，该路径上执行完毕；否则执行步骤 S116。

该步骤的伪代码如下：

```
int fetch(Node parent, List<Node> siblings) {
// 获取当前结点 parent 的所有直接后继结点，并将结果存储到 siblings 列表内
// 返回当前结点所有直接后继结点的数量，若为 0 表示当前结点为终端结点
int prev = 0;
for (int i = parent.start; i < parent.end; i++) {
    if (keys.get(i).length() < parent.depth) {
        // 若当前 key 已经处理完毕，跳过此 key，此处用于处理终端结点的情况
        // 即终端结点没有后继结点
        continue;
    }
    String tmp = keys.get(i);
    int cur = 0;
    if (keys.get(i).length() != parent.depth) {
        cur = (int) tmp.charAt(parent.depth) + 1;
    }
    if (cur != prev || siblings.isEmpty()) {
        Node tmp_node = new Node();
        tmp_node.depth = parent.depth + 1;
        tmp_node.code = cur;
        tmp_node.start = i;
        if (!siblings.isEmpty()) {
            (siblings.get(siblings.size() - 1)).end = i;
        }
        siblings.add(tmp_node);
    }
    prev = cur;
}
if (!siblings.isEmpty()) {
    (siblings.get(siblings.size() - 1)).end = parent.end;
}
```

```

return siblings.size();
}

```

步骤 S116: 为当前结点寻找一个合适的 base 值, 使得 base 值唯一, 且不会导致所有直接后继结点与现有 Trie 树存储的结点冲突。依次将当前结点的直接后继结点插入 Trie 树中, 并将其 check 值赋上当前结点的 base 值, 再依次把当前结点的直接后继结点作为当前结点, 跳转到步骤 S115。

该步骤的伪代码如下:

```

int insert(List<Node> siblings) {
// 为当前结点寻找一个符合条件的 base 值, 插入其所有直接后继结点
// 并返回当前结点的 base 值
为当前结点寻找一个合适的未被使用的 base 值, 使得 base 值唯一且不为 0, 也不会导致
所有直接后继结点与现有 Trie 树存储的结点冲突, 为了兼容 Trie 树的根结点, base 值从 1 开
始;
used[base] = 1; // 标识此值已被使用
for (int i = 0; i < siblings.size(); i++) {
// 将当前结点的直接后继结点的 check 值赋上当前结点的 base 值, 完成插入操作
array[(((int) base + (siblings.get(i)).code) << 1) + 1] = base;
}
for (int i = 0; i < siblings.size(); i++) {
// 依次把当前结点的直接后继结点作为当前结点
// 递归做获取直接后继结点集合和插入后继结点集合操作
List<Node> new_siblings = new ArrayList<Node>();
if (fetch((siblings.get(i)), new_siblings) == 0) {
// 此时当前结点为终端结点 (直接后继为空)
// 将其 base 值赋上该 key 词典序号的相反数
array[((int)base+(int)(siblings.get(i)).code) << 1] =(int)(-(siblings.get(i)).left - 1);
} else {
int ins = (int) insert(new_siblings);
// 将找到合适的 base: ins 赋给当前结点的 base 值
array[(((int) base + (siblings.get(i)).code) << 1) + 1] = ins;
}
}
}

```

```

}
return base;
}

```

所述一维线性空间的 Trie 树中将所有终端结点变为非终端结点，在所述终端结点后面增加一个叶子节点，并将叶子节点的 check 值赋上其存储位置。

将所有的终端结点变为非终端结点，并在其后面增加一个叶子结点，叶子结点的 check 值赋上自己的存储位置，叶子结点的 base 值赋上从初始结点（0 结点）到当前叶子结点路径上的输入字符组成的完整词条在整个有序词条集中位置的相反数（即叶子结点所处的 key 中在所有按词典排序的词条集中序号的相反数），因此结点的 base 值的符号用于标识是否是终端结点（叶子结点，其 base 值小于 0）。

步骤 S120：根据用户输入待查询词条 key。

Trie 树构建好之后，接下来就是查询用户输入的词条是否存在 Trie 树，即是否是从根结点到叶子结点的一条完整路径。

步骤 S130：根据词条 key 的当前状态实现查询。

如图 2 所示，是根据词条 key 的当前状态实现查询的流程框架图，具体步骤如下：

步骤 S131：把当前结点指向根节点。

步骤 S132：根据当前输入的字符做出状态转移，获取其直接后继状态的位置。

步骤 S133：校验当前状态的前驱，确定当前状态由哪一状态转移而来。

步骤 S134：获取词条 key 对应的 value 值。

在当前状态为 s，输入的字符为 c，下一状态为 t 的条件下，其本方法的查询过程的约束条件修改为：

$check[base[s] + c] = base[s]$ ； $base[s] + c = t$ ；每个状态的 base[s] 值唯一。

若当前状态 s 可以转移到叶子结点 t 中，则其约束条件为：

$base[s] = t$ ； $t = check[t]$ 。

$base[t] < 0$ 且 base[t] 的值为 DFA 的初始结点 0 到当前叶子结点经过的字符组成的词条在所有按词典顺序排序的词条集中位置的相反数。

本发明实施例中可以获取词条 key 的所有前缀词，并可以将检索到的每条结果信息保存在一个对象 TrieResult 中，该表存储的变量作如下描述：Length 表示当前 key 的长度；Index 当前 key 在词典中的存储序号-1，即当前 key 对应属性的存储位置。

本发明实施例一种基于一维线性空间实现 Trie 树的词典检索方法的查询速度 18.3MB/s。

因此，本发明提供一种基于一维线性空间实现 Trie 树的词典检索方法，通过生成一维线性空间的 Trie 树词典数据；根据用户输入确定待查询词条 key；根据词条 key 的当前状态实现查询。在一维线性空间下构建 Trie 树的词典数据下，提高了词典加载和检索速度，并且能够快速检索到词条的所有前缀词，同时，在 Trie 树的构建过程中通过调整 base 值，使得其所有直接后继均不会发生冲突，因此避免了数据移动或存储空间分配的回溯问题。

权利要求书

1. 一种基于一维线性空间实现 Trie 树的词典检索方法，其特征在于，该方法包括：
生成一维线性空间 Trie 树的词典数据；
根据用户输入确定待查询词条 key；
根据词条 key 的当前状态实现查询。
2. 根据权利要求 1 所述的方法，其特征在于，所述词典数据的 key 转化为节点后存储在一维数组中，所述一维数组的值用来标识 base 值是否唯一。
3. 根据权利要求 1 所述的方法，其特征在于，所述一维线性空间 Trie 树中将所有终端结点变为非终端结点，在所述终端结点后面增加一个叶子节点，并将叶子节点的 check 值赋上其存储位置。
4. 根据权利要求 1 或 3 所述的方法，其特征在于，所述叶子节点还包括：叶子节点的 base 值用以标识其是否为终端结点。
5. 根据权利要求 1 所述的方法，其特征在于，所述查询包括以下步骤：
把当前结点指向根节点；
根据当前输入的字符做出状态转移，获取其直接后继状态的位置；
校验当前状态的前驱，确定当前状态由哪一状态转移而来；
获取词条 key 对应的 value 值。
6. 根据权利要求 1 或权利要求 5 所述的方法，其特征在于，所述查询包括：
根据词条 key 的查询可以获得其所有前缀词的结果。

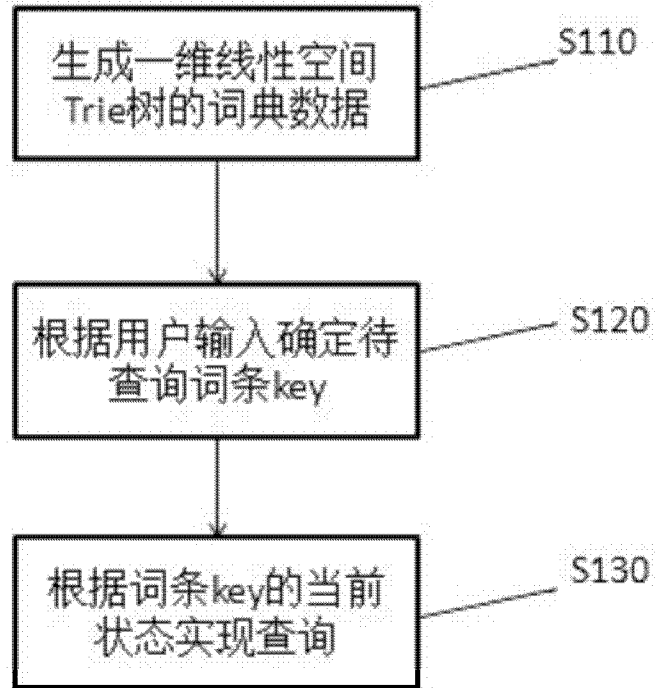


图 1

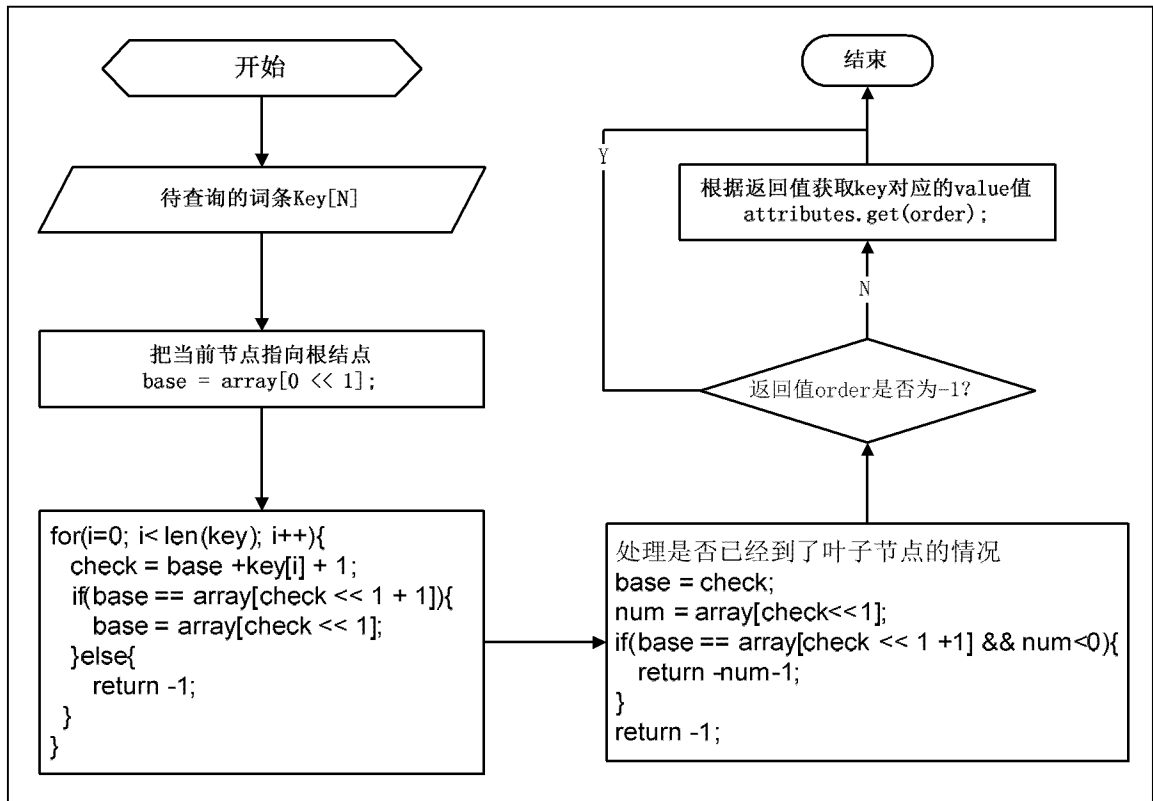


图 2

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2014/080179

A. CLASSIFICATION OF SUBJECT MATTER

G06F 17/30 (2006.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F 17/-

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNPAT, CNKI, WPI, EPODOC: dictionary, entry, tree, retrieve, storage, inquire, search, find, one?dimensional, array, node, state, Trie,
base, check, key, value, vocabulary

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
PX	CN 103365992 A (SHENZHEN HUAAO DATA TECHNOLOGY CO., LTD) 23 October 2013 (23.10.2013) claims 1 to 6	1-6
Y	CN 1786962 A (INST OF COMPUTING TECH CAS) 14 June 2006 (14.06.2006) claims 1 to 6, description, page 3, line 17 to page 4, line 8, page 6, lines 13 to 29, page 7, lines 16 to 27 and figures 1 to 3	1-6
Y	CN 101499094 A (FOCUS TECHNOLOGY CO., LTD) 05 August 2009 (05.08.2009) claims 1, 3 and 5, description, page 8, lines 6 to 9	1-6
A	CN 101398830 A (ALIBABA GROUP CO., LTD) 01 April 2009 (01.04.2009) the whole document	1-6
A	CN 102651026 A (BEIJING BAIDU NETCOM SCI & TEC) 29 August 2012 (29.08.2012) the whole document	1-6

Further documents are listed in the continuation of Box C.

See patent family annex.

<p>* Special categories of cited documents:</p> <p>“A” document defining the general state of the art which is not considered to be of particular relevance</p> <p>“E” earlier application or patent but published on or after the international filing date</p> <p>“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)</p> <p>“O” document referring to an oral disclosure, use, exhibition or other means</p> <p>“P” document published prior to the international filing date but later than the priority date claimed</p>	<p>“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention</p> <p>“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone</p> <p>“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art</p> <p>“&” document member of the same patent family</p>
---	---

Date of the actual completion of the international search 31 August 2014	Date of mailing of the international search report 24 September 2014
---	---

<p>Name and mailing address of the ISA State Intellectual Property Office of the P. R. China No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088, China Facsimile No. (86-10) 62019451</p>	<p>Authorized officer HU, Ni Telephone No. (86-10) 82245952</p>
--	---

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2014/080179

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2008208854 A1 (3618633 CANADA INC.) 28 August 2008 (28.08.2008) the whole document	1-6

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2014/080179

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
CN 103365992 A	23 October 2013	None	
CN 1786962 A	14 June 2006	CN 100444167 C	17 December 2008
CN 101499094 A	05 August 2009	CN 101499094 B	29 September 2010
CN 101398830 A	01 April 2009	CN 101398830 B	27 June 2012
		HK 1128977 A1	16 November 2012
CN 102651026 A	29 August 2012	None	
US 2008208854 A1	28 August 2008	US 7689588 B2	30 March 2010
		CA 2509496 A1	06 December 2006
		CA 2608772 A1	14 December 2006
		WO 2006130947 A1	14 December 2006
		WO 2006130947 A8	27 December 2007
		CA 2608772 C	18 June 2013

国际检索报告

国际申请号

PCT/CN2014/080179

A. 主题的分类 G06F 17/30 (2006.01) i 按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类		
B. 检索领域 检索的最低限度文献(标明分类系统和分类号) G06F 17/- 包含在检索领域中的除最低限度文献以外的检索文献 在国际检索时查阅的电子数据库(数据库的名称, 和使用的检索词(如使用)) CNPAT, CNKI, WPI, EPODOC: 词典, 字典, 词条, 树, 检索, 存储, 查询, 搜寻, 查找, 一维, 数组, 节点, 结点, 状态, dictionary, Trie, tree, search, one?dimensional, base, check, key, value, vocabulary, node		
C. 相关文件		
类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求
PX	CN 103365992 A (深圳市华傲数据技术有限公司) 2013年 10月 23日 (2013 - 10 - 23) 权利要求1-6	1-6
Y	CN 1786962 A (中国科学院计算技术研究所) 2006年 6月 14日 (2006 - 06 - 14) 权利要求1-6、说明书第3页第17行-第4页第8行、第6页第13-29行、第7页第16-27行、附图1-3	1-6
Y	CN 101499094 A (焦点科技股份有限公司) 2009年 8月 05日 (2009 - 08 - 05) 权利要求1、3、5、说明书第8页第6-9行	1-6
A	CN 101398830 A (阿里巴巴集团控股有限公司) 2009年 4月 01日 (2009 - 04 - 01) 全文	1-6
A	CN 102651026 A (百度在线网络技术北京有限公司) 2012年 8月 29日 (2012 - 08 - 29) 全文	1-6
A	US 2008208854 A1 (3618633 CANADA INC.) 2008年 8月 28日 (2008 - 08 - 28) 全文	1-6
<input type="checkbox"/> 其余文件在C栏的续页中列出。 <input checked="" type="checkbox"/> 见同族专利附件。		
* 引用文件的具体类型: “A” 认为不特别相关的表示了现有技术一般状态的文件 “E” 在国际申请日的当天或之后公布的在先申请或专利 “L” 可能对优先权要求构成怀疑的文件, 或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件 (如具体说明的) “O” 涉及口头公开、使用、展览或其他方式公开的文件 “P” 公布日先于国际申请日但迟于所要求的优先权日的文件 “T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了理解发明之理论或原理的在后文件 “X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性 “Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性 “&” 同族专利的文件		
国际检索实际完成的日期 2014年 8月 31日		国际检索报告邮寄日期 2014年 9月 24日
ISA/CN的名称和邮寄地址 中华人民共和国国家知识产权局(ISA/CN) 北京市海淀区蓟门桥西土城路6号 100088 中国 传真号 (86-10)62019451		受权官员 胡妮 电话号码 (86-10)82245952

国际检索报告
关于同族专利的信息

国际申请号

PCT/CN2014/080179

检索报告引用的专利文件			公布日 (年/月/日)	同族专利			公布日 (年/月/日)
CN	103365992	A	2013年 10月 23日	无			
CN	1786962	A	2006年 6月 14日	CN	100444167	C	2008年 12月 17日
CN	101499094	A	2009年 8月 05日	CN	101499094	B	2010年 9月 29日
CN	101398830	A	2009年 4月 01日	CN	101398830	B	2012年 6月 27日
				HK	1128977	A1	2012年 11月 16日
CN	102651026	A	2012年 8月 29日	无			
US	2008208854	A1	2008年 8月 28日	US	7689588	B2	2010年 3月 30日
				CA	2509496	A1	2006年 12月 06日
				CA	2608772	A1	2006年 12月 14日
				WO	2006130947	A1	2006年 12月 14日
				WO	2006130947	A8	2007年 12月 27日
				CA	2608772	C	2013年 6月 18日

表 PCT/ISA/210 (同族专利附件) (2009年7月)