PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification ^{6} :
G06F 17/30

A1

(11) International Publication Number: WO 99/17234

(43) International Publication Date: 8 April 1999 (08.04.99)

(21) International Application Number: PCT/US98/20362

(22) International Filing Date: 29 September 1998 (29.09.98)

(30) Priority Data:

60/060,376	29 September 1997 (29.09.97)	US
09/162,187	28 September 1998 (28.09.98)	US

(71) Applicant: TRIADA, LTD. [US/US]; Suite 311, 315 E. Eisenhower Parkway, Ann Arbor, MI 48108 (US).

(72) Inventors: ZHANG, Tao; Apartment 216, 635 Hidden Valley Drive, Ann Arbor, MI 48105 (US). RAGHAVAN, R., K.; 254 Princeton Street, Canton, MI 48188 (US).

(74) Agents: POSA, John, G. et al.; Gifford, Krass, Groh, Sprinkle, Patmore, Anderson & Citkowski, P.C., Suite 400, 280 N. Old Woodward Avenue, Birmingham, MI 48009 (US).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, HR, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: MULTI-DIMENSIONAL PATTERN ANALYSIS

(57) Abstract

A pattern recognition method applicable to unique associated pattern recognition in a structured information system or structured database is presented. The process may be used to find patterns in a column which are associated with unique data values in another column, or to find the number of unique values in the second column which are paired with the same associated pattern in the first column. The technique is easily extended to more general cases in which both the condition field and the associated pattern field may be two groups of fields.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania
AM	Armenia
AT	Austria
AU	Australia
AZ	Azerbaijan
BA	Bosnia and Herzegovina
BB	Barbados
BE	Belgium
BF	Burkina Faso
BG	Bulgaria
BJ	Benin
BR	Brazil
BY	Belarus
CA	Canada
CF	Central African Republic
CG	Congo
CH	Switzerland
CI	Côte d'Ivoire
CM	Cameroon
CN	China
CU	Cuba
CZ	Czech Republic
DE	Germany
DK	Denmark
EE	Estonia
ES	Spain
FI	Finland
FR	France
GA	Gabon
GB	United Kingdom
GE	Georgia
GH	Ghana
GN	Guinea
GR	Greece
HU	Hungary
IE	Ireland
IL	Israel
IS	Iceland
IT	Italy
JP	Japan
KE	Kenya
KG	Kyrgyzstan
KP	Democratic People's Republic of Korea
KR	Republic of Korea
KZ	Kazakstan
LC	Saint Lucia
LI	Liechtenstein
LK	Sri Lanka
LR	Liberia
LS	Lesotho
LT	Lithuania
LU	Luxembourg
LV	Latvia
MC	Monaco
MD	Republic of Moldova
MG	Madagascar
MK	The former Yugoslav Republic of Macedonia
ML	Mali
MN	Mongolia
MR	Mauritania
MW	Malawi
MX	Mexico
NE	Niger
NL	Netherlands
NO	Norway
NZ	New Zealand
PL	Poland
PT	Portugal
RO	Romania
RU	Russian Federation
SD	Sudan
SE	Sweden
SG	Singapore
SI	Slovenia
SK	Slovakia
SN	Senegal
SZ	Swaziland
TD	Chad
TG	Togo
TJ	Tajikistan
TM	Turkmenistan
TR	Turkey
TT	Trinidad and Tobago
UA	Ukraine
UG	Uganda
US	United States of America
UZ	Uzbekistan
VN	Viet Nam
YU	Yugoslavia
ZW	Zimbabwe

MULTI-DIMENSIONAL PATTERN ANALYSIS

Field of the Invention

The present invention relates generally to computer databases, including structured information systems and associative memory databases and, in particular, to processes for identifying unique patterns among fields within applicable database structures.

Background of the Invention

As set forth in commonly assigned U.S. Patent Nos. 5,245,337, 5,293,164, and 5,592,667, a multi-dimensional approach has been developed for transforming an unstructured information system into a structured information system. This approach addresses the unique properties of multiple information source systems, including database systems, from an information point of view. In particular, this new methodology attempts to unify the two fields of information theory and database by combining the encoding compression theory and the database theory for general data manipulations into a general information manipulation theory with respect to multi-dimensional information space.

Broadly, multiple information sources are described by different information variables, each corresponding to one information source or information stream. Information manipulations are primarily index manipulations which are, in general, more efficient than non-index manipulations of the same number. The only non-index manipulations are carried out at leaf nodes where unique data values are stored. Therefore, the non-index manipulations are minimized. As a further aspect of this approach, a structured information system or database is built by taking into account information relations between different sets of data in the database. Such relations between neighboring nodes are easily analyzed and presented on-line because they are built into the structure. Relations between nodes that are not neighbors are not explicitly built into the existing structure. On-line analysis of these relations requires efficient information manipulations in main memory.

The approach models multiple information sources as different information variables, wherein each variable corresponds to one information source or information stream. In accordance with the methodology, information variables at leaf nodes of an associative memory database structure assume unique data values. The resulting structured database makes it easy to obtain statistical information about the data stored in the database. However, only a limited amount of statistic information can be readily presented once a given tree structure is built. For example, whereas it is trivial to show double patterns formed from two leaf nodes which happen to be the two child nodes of the same double pattern internal node in an existing tree structure, it is non-trivial to show similar double patterns formed from two leaf nodes that have an immediate common ancestor node which is not a double-pattern node in the existing tree structure.

Summary of the Invention

The present invention may be used to solve problems of the type just described in a general way in a structured information system or an associative memory database (AMDB). As one example of many, the process may be used to find patterns in a column which are associated with unique data values in another column, or to find the number of unique values in the second column which are paired with the same associated pattern in the first column. The technique is easily extended, however, to more general cases in which both the condition field and the associated pattern field may be two groups of fields.

In a more particular sense the invention addresses the association of conditional patterns in a field or a single information source with different data values in another field or information source. The first field or information source may be characterized as an associated pattern field, and the second is a condition field. In such a case the invention may be used to determine unique conditional patterns and counts which represent the number of unique data values in the condition field, that are associated with the same unique conditional pattern. The invention finds particular utility in structured information systems, wherein the fields correspond to leaf nodes of a tree structure.

Detailed Description of the Invention

This invention resides in the solution to certain classes of problems that occur in structured information systems, including associative memory databases (AMDBs). One typical, interesting problem adopts the following form: First, assume some patterns exist in a field f_1 or a column, wherein each pattern is associated with a unique data value in a different field f_2, or each pattern is made up of a set of data values in field f_1, all of which pair with the same data value in field f_2. Given this assumption, find out all the unique patterns and counts representative of a unique data value in field f_2 that pairs with the same pattern in f_1.

Although it is difficult to identify such patterns, the solution to this problem is quite interesting in many databases. For example, consider an automobile warranty database containing a vehicle ID field and an option code field, where each field value in the option field corresponds to an option associated with a vehicle identified by its vehicle ID. A vehicle may have many different options, and a complete set of options associated with each vehicle ID is an option pattern or option package. The problem here is to determine all the unique option packages and the number of vehicles that have the same pattern or package. Such pattern recognition may help to identify popular packages and to eliminate those unpopular packages or those with low counts from an assembly line.

Broadly, then, problems of this kind have to do with the association of conditional patterns in a field or a single information source with different data values in another field or information source. The first field or information source may be characterized as an associated pattern field, and the second is a condition field. The goal is to find all the unique conditional patterns and counts which represent the number of unique data values in the condition field, that are associated with the same unique conditional pattern.

To solve the problem, consider two given fields or information sources in an existing tree structure. Assume field a is the condition field and field b is the associated pattern field. In a structured information system, these fields correspond to two leaf nodes, a and b. The difference between the two fields and two leaf nodes is that the two fields are two unstructured information sources which may have redundant information values or data values, and the two leaf nodes are unique information sources which have only unique information values or unique data values.

First, we find the immediate common ancestor node n_a of the two leaf nodes a and b in the existing tree structure. Assume further that the left child node of the common ancestor node n_a is n_1, an ancestor node of node a, but not of node b. Similarly, the right child node n_r of the ancestor node is an ancestor node of node b, but not of node a.

Now, we recall the tokens of node a at node n_1, and recall the tokens of node b at node n_r. This procedure propagates the a tokens to node n_1 and the b tokens to node n_r. The memory tokens of node n_1 are replaced by the recalled tokens of node a and the tokens of node n_r are replaced by the tokens of node b.

Next, we load the memory structure of the ancestor node n_a using the left hashing lists (right hashing lists if the right child node of the common ancestor node is an ancestor node of the condition leaf). In the left hashing structure, a set of lists are built. Each list has a left child token as its list index and stores a set of right child tokens as list elements. Any given list represents pairing between the left child token or the list index and the right child tokens stored in the list.

We replace the list indices by the corresponding tokens of node a and replace the list elements by the corresponding tokens of node b, anticipating some redundant list indices and list elements in the general case. We combine all the lists that have the same list index or that pair with the same token of node a, eliminate redundant elements in each new list, and sort the remaining elements. At this point, we have a set of lists which all have unique list indices. Some lists may store a set of the exact same tokens, although they have different list indices.

Two interesting problems arise here. One is to find out how many unique sets of tokens stored in the lists and what is the number of appearances of each unique set of tokens. To solve this problem, we eliminate identical lists that store the same set of tokens of the leaf node b and keep counting the number of appearances for each unique list. We recall data values of the leaf node b and replace the tokens in each list by the corresponding data values. In this way, we obtain all the unique patterns in field b, associated with unique data values in field a, and the number of appearances for each unique pattern.

EXAMPLES

Assume a database in which one field is user ID identifying a cable TV and another field is option channels specifying optional paid channels associated with each cable TV. Assume there are 100,000,000 records and 20,000,000 different user IDs or cable TVS. The total number of different optional paid channels is 100. On average, each user ID has about 5 paid channels. In the case of conditional pattern recognition, there are some 20,000,000 patterns or packages of optional paid channels. Some patterns may be made of up to 100 optional channels, corresponding to the maximum number of the paid channels each cable TV or user ID can order. Others may have as little as one paid channel.

By virtue of this invention, one might find 10,000 unique patterns of optional channels. On average, each pattern may have counts of 2,000 representing the average number of appearances of each pattern. Some patterns may appear as many times as hundreds of thousands of times or even millions. Others may show up only once. Such patterns of information may help to identify what option packages are popular and what are not. Similarly, one may set phone users who switched phone company to be the condition field and find out what patterns are stored in the hashing lists, associated with each switch specified in the condition field.

It will be appreciated by one of skill in the art that the invention is applicable to a broad range of other problems. To take one further example of many, it may be interesting to find out what is the pattern made up of a set of the left child tokens pairing with the same set of right child tokens or the same right child pattern. This is a pattern-pairing-pattern problem. To solve this problem, one needs to store a set of list indices that correspond to the same set of list elements.

I claim:

1. A method of identifying unique data values in a structured information system having a pattern field and a condition field, comprising the steps of:

determining the immediate common ancestor of the pattern field and the condition field;

recalling the tokens of the two fields;

replacing the field values with the respective recalled tokens;

loading the memory structure of the ancestor node with hashing lists corresponding to the two fields;

building a new set of lists indicative of pairing between list indices and child tokens in accordance with the hashing lists; and

replacing the list indices with the tokens corresponding to the indices.

2. The method of claim 1, further including the steps of:

eliminate redundant elements in each new list; and

sorting the remaining elements.

3. The method of claim 1, wherein the information system is a tree structure, and the method is used to reveal similar, double patterns formed from two leaf nodes that have an immediate common ancestor node which are not a double-pattern node in the existing tree structure.

4. In an information system having two fields or information sources corresponding to leaf nodes a and b of a tree structure, and wherein the left (or right) child node of the common ancestor node n_a is n_1, and an ancestor node of node a, but not of node b, and the right (or left) child node n_r of the ancestor node is an ancestor node of node b, but not of node a, a method of rearranging the structure to perform certain query operations, comprising the steps of:

locating the immediate common ancestor node n_a of the two leaf nodes a and b;

recalling the tokens of node a at node n_1 and recalling the tokens of node b at node n_r, thereby propagating the a tokens to node n_1 and the b tokens to node n_r;

replacing the memory tokens of node n_1 with the recalled tokens of node a and replacing the tokens of node n_r with the tokens of node b;

loading the memory structure of the ancestor node n_a using the left or right hashing lists, as appropriate;

building a set of lists corresponding to the left hashing structure, such that any given list represents pairing between the left child token or the list index and the right child tokens stored in the list, wherein each list has a left child token as its list index and stores a set of right child tokens as list elements;

replacing the list indices by the corresponding tokens of node a and replace the list elements by the corresponding tokens of node b;

combining all the lists that have the same list index or that pair with the same token of node a, thereby eliminate redundant elements in each new list; and

sorting the remaining elements.

5. The method of claim 4, further including the step of eliminating identical lists that store the same set of tokens of the leaf node b while storing the number of appearances for each unique list;

recalling data values of the leaf node b; and

replacing the tokens in each list by the corresponding data values.

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US98/20362

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :G06F 17/30
US CL : 707/2
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)

U.S. : 707/1,2,3,6

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)

APS and DIALOG
search terms: analysis, identify, statistic, pattern, data, database, field, column, mining

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
AP	US 5,809,499 A (WONG et al) 15 September 1998.	1-5N
AE	US 5,832,182 A (ZHANG et al) 3 November 1998.	1-5

□ Further documents are listed in the continuation of Box C. □ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document published on or after the international filing date

"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)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"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

"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

"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

"&" document member of the same patent family

Date of the actual completion of the international search

03 JANUARY 1999

Date of mailing of the international search report

02 FEB 1999

Name and mailing address of the ISA/US
Commissioner of Patents and Trademarks
Box PCT
Washington, D.C. 20231

Facsimile No. (703) 305-3230

Authorized officer
Joni Hill
JACK M. CHOULES

Telephone No. (703) 305-9840