Journal of Applied Sciences 4 (3): 344-353, 2004
ISSN 1607-8926

Improving Business Efficacy with Revised Graph-based Mining Association Rules

K.Q. Yan and [1]S.C. Wang
Department of Business Administration, [1]Department of Information Management,
Chaoyang University of Technology,
168 Gifeng E. Rd., Wufeng, Taichung County, Taiwan, 413, Republic of China

Abstract: Due to the internationalization of the domestic business environment nowadays, competitions that every company has to survive have come not merely from the challenges of other local companies but from everywhere around the world. With the revised graph-based mining association rules we offer, all we need is one database scan process. After the scan, the bit vector corresponding to each product can be established and any association rule can then be arbitrarily picked out for rule mining. This way, exhaustive, repetitive scans over the whole database will no longer be necessary. Our revised graph-based mining association rules are quite so flexible that they can be adaptively used for distinct association rule mining in distinct fields.

Key words: Data mining, association rule, graph-based mining association rule, knowledge management, information technology management

INTRODUCTION

For retailers and wholesalers who are facing more and more competitions and challenges, the market has become tougher and tougher. It is absolutely no easy job for them to create their unique value and stand out among all those competitors when too many companies are selling the same things with the prices dropping unreasonably low and the customers scattering and wandering about. Most retailers and wholesalers in recent years have equipped themselves with lots and lots of information processing technologies so as to simplify the collecting and storing of their business details and to enable themselves to put together and analyze their sales records and everything by exploiting the mathematical power of their computer systems. For example, the well-celebrated bar code technique has been used far and wide to speed up business flows and so are the world-famous POS (point of sales) systems used to record every trade every customer has made with the company so that stock control can be properly done[1]. After all, the ability to manage and gain quick access to any and every wanted detailed trade record with a customer is one the key to the survival of a business.

Therefore, for retailers and wholesalers of all sizes everywhere, it is, we can say, the most important thing to keep track of and to learn from their past dealing and selling records and their customer information and dig down into the heart of those whole packs of data and uncover the implicit consumption behavior patterns and the psychology underlying those patterns. By properly responding to such tacit customer knowledge, companies can easily and reasonably increase their business and of course their profits, which will turn the companies into more powerful, more competitive survivors[2]. However, how exactly can we "dig out" those so-called useful "customer consumption behavior patterns" from tons and tons of files of all kinds? Well, the answer to this question, we think, is knowledge management. In the field of knowledge management, data mining is the most celebrated technique that deals with knowledge elicitation processes[3-6]. Data mining can also be alternatively called "knowledge discovery." The whole point of knowledge discovery is to extract such valuable and yet seemingly abstract things as knowledge rules, constraints and regularities that can be of great help in the future from large quantities of raw data in past company records[1].

The most famous application of association rules in the field of data mining is the market basket analysis[7]. Through the use of association rules, trade records in association-rule-based databases can be drawn out for analyses and comparisons to derive the logical relationships between customers and the products they have bought. In 2001, Yen and Chen[8] proposed their innovative method-graph-based association rules. Their method is a breakthrough capable of extracting and

Corresponding Author: Amy, S.C. Wang, Department of Information Management, Chaoyang University of Technology, 168 Gifeng E. Rd., Wufeng, Taichung County, Taiwan, 413, Republic of China Tel: 886-4-2332-3000 Ext: 3131 Fax: 886-4-2374-2337 or 886-4-2374-2309 E-mail: scwang@mail.cyut.edu.tw

analyzing the association patterns that exist in primitive association rules, generalized association rules and multi-level association rules and then putting together some dominant association rules that match the users' expectations. Unfortunately, when such a method is facing cases of multi-level association rule mining, it needs to do exhaustive scans to the whole database repeatedly until it finds appropriate association rules. This of course has an influence on the efficacy.

In this study, we would like to offer our "revised graph-based mining association rules." As the name suggests, our revised graph-based mining association rules are in fact just a step further from their original form. With our new method, all we need is just one database scan; after it builds the bit vector corresponding to each and every product listed in the database, we can arbitrarily pick out any association rule at all to do association rule mining and no further whole-database scans are needed any more. This sure will make multi-level association rule mining much easier and more efficient.

Association rules: In the field of data mining, the most celebrated method of all is the association rules mining technique applied to the so-called market basket analysis. Basically, association rules are used to find out customers' consumption behavior patterns from among trade records. For example, in the following association rule expression (1):

Buys (x, "diapers")?Buys (x, "beers") [5%, 60%]...(1)

The variable "x" stands for the customer. In this example, the customer x has a support rate of 5% and a confidence rate of 60%, where

Support (5%) = number of records with diapers and beers appearing at the same time/number of all records

Confidence (60%) = number of records with diapers and beers appearing at the same time/number of all records with diapers

Association rule expression (1) reveals the following two facts: (a) According to all the records in the database, 5% of the customers are likely to buy diapers and beers at the same time; (b) Of all the diaper buyers, 60% are likely to also buy beers. In this expression, the strength of the association between "diapers" and "beers" is evaluated by two probability values: support and confidence. In the example above, the support value is 5% and the confidence value is 60%.

The Apriori algorithm is the earliest algorithm developed for association rules mining[7]. It works by picking out a candidate set from all the trade records, filtering out those products whose support values are smaller than the preset minimum support to form the so-called "large k itemset," and then doing self join on the large k itemset to form the k+1 itemset, which is the candidate set in the next round. Therefore, this process repeats and repeats until there can never be another large k+n itemset any longer.

The bottleneck of the Apriori algorithm is that when the large 1 itemset does self join, it produces a candidate 2 itemset with a large number of elements in it. Besides that, after the production of every candidate set, the database has to be scanned all over in order to figure out how many times the products in the candidate set appear in the database[7]. As a result, to save the Apriori algorithm from its weakness of high computation cost, many researchers have developed methods to improve it.

To improve the Apriori algorithm, two ways have been tried: First, the candidate set production method remains the same, while some sophisticated skills are employed to prevent the algorithm from forming candidate 2 itemsets with large numbers of elements, for example, DHP (Direct Hash with efficient Pruning for fast data mining)[9]. The other way is not to produce any candidate set at all but to change the data structure-for example, turns it into a tree or graph-to store the frequencies of all the product combinations there are in the trade record database. The strength of this method is that only one higher cost database scan is needed at the very beginning; after that, any association rules in the reformed database can be easily acquired without having to scan the whole database again. Methods of this kind get rid of the weakness of repeated database scans by simply reducing the number of database scans done. Yen and Chen's general graph-based association rules mining technique[8] is a method of this kind.

Categorization of association rules: The categorization of association rules can depend on whether they are qualitative or quantitative, whether they have attributes of different dimensions, or whether they come from different conceptual levels.

Depending on whether they are qualitative or quantitative, association rules can be either Boolean associations or not.

Boolean associations: These association rules take into consideration only whether or not the products were sold; in other words, these rules are not concerned about the numbers of the items sold. For example, in expression (1) above, the exact numbers of diapers and beers sold are ignored; to put it another way, whether the customer bought a dozen of beers or just one can does not make any difference to this association rule.

Quantitative associations: Some data have non-binary attributes, for example, age, income, numbers of items

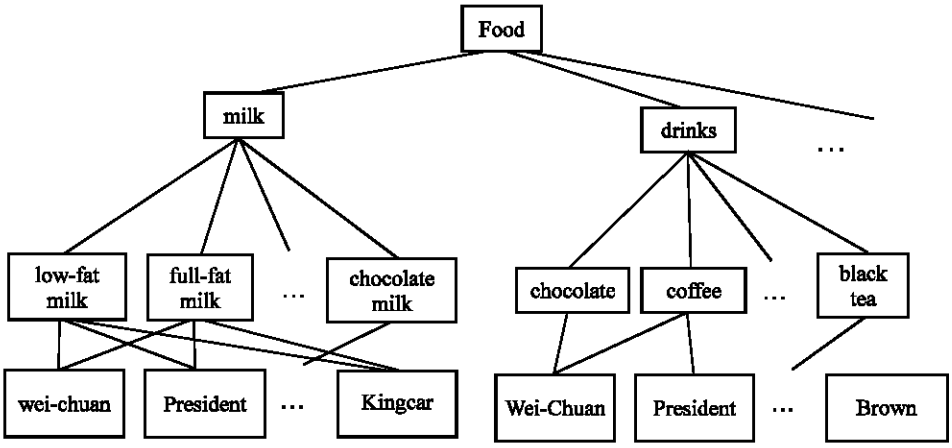

Fig. 1: A conceptual hierarchy

bought, etc. When these data have to be clearly identified, we need to use quantitative association rules instead of Boolean associations[10].

If we classify association rules by their dimensional attributes, then they can be either single dimensional associations or multiple dimensional associations.

Single dimensional associations: All the factors of the association rule extracted belong to the same dimension. For example, Buys (x, "diapers") ^ Buys (x, "milk") ? Buys (x, "beers") [5%, 60%]. In this rule, all the associated factors are products bought; that is, they belong to the same dimension.

Multiple dimensional associations: The factors of the association rule extracted are from more than one dimension. For instance, Age (x, "30-40") ^ Income (x, "60K-90K") ? Buys (x, "Gold Credit Card") [5%, 60%]. In this rule, the associated factors are age, income and buys-each comes from a different dimension.

Han and Fu[11] as well as Srikant and Agrawal[12] have introduced the concept of hierarchical structure into the field of association rules mining. They have enabled this data mining technique to reach out much farther and wider and have also made association rules come in much handier when practical decisions are to be made. Fig. 1 shows an example conceptual hierarchy.

A conceptual hierarchy is basically a set of elements from different levels. In a conceptual hierarchy, the higher-level nodes (sometimes called parent nodes) usually have some lower-level nodes (child nodes) to themselves. Child nodes have more concisely defined meanings and clear-cut boundaries, while their parent nodes have broader meanings and more blurred boundaries and the higher the level, the wider the range. For example, in Fig. 1, low-fat milk, full-fat milk and chocolate milk all belong to the category "milk." Low-fat milk is a child node with a quite narrowed down, quite concisely defined meaning: "milk that has a low percentage of fat;" on the other hand, the parent node "milk" is comparatively broader in meaning and contains such child nodes as low-fat milk, full-fat milk, chocolate milk and so on.

According to the number of levels in the conceptual hierarchy, association rules can be classified as either single level rules or multiple level rules.

Single level association analysis: The rules extracted involve only factors on one single conceptual level. In[8,11,12] such association rules are referred to as primitive associations. For example, in this rule "Buys (x, "diapers") ^ Buys (x, "milk") ? Buys (x, "beers") [5%, 60%]," diapers, beers and milk all belong to the same conceptual level "products." As for the brands of the diapers and the beers that are usually bought by the same customers at the same time, well, this information is not revealed in the single level association rule here.

Multiple level association analysis: In contrast to single level association rules, the mining of multiple level association rules requires taking into consideration both multiple attributes of the items analyzed and the levels the attributes belong to. For example, in multiple level association rules, the attributes that are absent from single level association rules, like the brand, size, kind, etc., can be shown and analyzed in multiple level association rules. In other words, in a multiple level association rule, such information as "those customers who buy brand A low-fat milk tend to buy brand B chocolate-flavored cookies" can be revealed. That is, the node "milk" can have branches classifying its child nodes classified by such attributes as brand and kind and so can

the node "cookies" have child nodes classified by such attributes as brand as flavor.

Multiple level association rules are basically developed with a view to rolling up or drilling down the classifying concepts, making it possible to reveal the correlations between products across different conceptual levels. For example, a jacket is a short kind of coat and a coat is a piece of clothing. A user might want to infer the rule "Buys (x, "coat") ? Buys (x, "shoes")." As far as categorization is concerned, the association rules on this conceptual level can be connected to such rules as "Buys (x, "clothing") ? Buys (x, "shoes")" that are one level higher, as well as connected to such rules as Buys (x, "jacket") ? Buys (x, "shoes")" that are one level lower. This kind of rule may either be established or not, for the category "coat" covers more than just jackets. Therefore, generalization[10] and specialization[11] are done to the conceptual levels of the trade data so that the strong rules extracted out can have power that is more decisive when crucial commercial strategies are to be settled on.

The most famous methods of multiple level association rules mining include the generalized association rules proposed by Srikant and Agrawal[10] and the multiple level association rules by Han and Fu[11]. Srikant and Agrawal's method[10] does the so-called "roll-up" from the lower levels up to the higher levels, while Han and Fu's method[11] follow the opposite direction: doing the so-called "drill-down" and going from the higher level concepts down to the lower level concepts.

Graph-based association rules mining: Yen and Chen[8] proposed their graph-based association rule mining technique to improve the shortcoming of the scheme by Apriori[7] that the database needs to be repeatedly scanned. With yen and Chen's scheme, the database only needs to be scanned once; after setting up the association graph, the system can do scans directly to the graph and search for all the hidden association rules in it.

Yen and Chen's method is basically composed of five stages: (1) serial number generation phase, (2) Large 1 itemset generation phase, (3) association graph set-up phase, (4) association pattern generation phase and (5) association rule generation phase.

For example, let us look at the trade records in Table 1[11]. Suppose the lowest threshold of the association rule's support value is 2. Because the focus of this study is on the revision of the methodology, we shall skip the generation process of association rule in the fifth phase. Generally speaking, in the practical business environment, the identification of strong association rules involves so much more domain knowledge than in other environments that the final decision still depends heavily on expert knowledge, which means total automation is currently still way out of reach. In other words, association rules with high support and confidence values do not necessarily mean absolute accuracy and practical value. Therefore, we shall emphasize only phases (1) through (4).

Table 1: Trade records

TID	Items
T1	A, C, E, F
T2	A, E, G, H
T3	B, D, F, I
T4	A, C
T5	A, D, E, F, J
T6	E, H, K
T7	H, I, K, L

Phase (1) serial number generation: To every product added to the trade record table, one and only one unrepeated integer serial code can be given in arbitrary order. In this example, products A through L are given one and only one serial number each, namely serial numbers 1 through 12. After the execution of this procedure, we get the following results:

A: 1	B: 2	C: 3
D: 4	E: 5	F: 6
G: 7	H: 8	I: 9
J: 10	K: 11	L: 12

Phase (2) large 1 itemset generation: In this phase, all the trade records in the database must be scanned and the number of times of appearance must be figured out for every product. The support values must be checked to see if they exceed the threshold value. In addition, the bit vector table must be set up for every product item that has got its serial number. The length of the bit vector equals the number of the trade records in the database; the bit position in the bit vector table stands for the serial order by which the product item appear in the trade records. For example, suppose the bit vector of product A is 1101100, in which the bit value "1" appears at the first, the second, the fourth and the fifth bit position, standing for the fact that this product A appears in the first, the second, the fourth and the fifth trade record in the database. On the other hand, the bit value "0" shows up at the third, the sixth and the seventh bit position. That means this product A does not show up in the third, the sixth and the seventh trade record in the database. Following this method, we can decide the bit vectors for all the other product items. After this step, we have the following bit vectors: 1101100 for product A, 0010100 for product D, 1100110 for product E, 1010100 for product F,

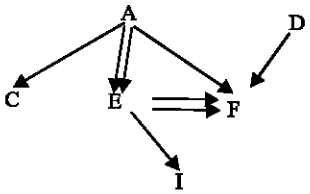

Fig. 2: Association graph

0100011 for product H, 0010001 for product I and 0000011 for product K. Totally, there are seven products whose support values are greater than or equal to the threshold (Count the number of digits "1" in the bit vector of each product. If the number is greater than or equal to 2, then the support value passes the threshold). In this case, product B (with the bit vector 0010000), product C (1001000), product G (010000), product J (0000100) and product L (0000001) fail to pass the threshold, so we simply ignore them. To figure out the number of trade records in which any two products whose support values are greater than or equal to the threshold show up together, we can do the "AND" operation on the two products. In the resulting bit vector, the number of digits "1" that show up stands for the number of trades where both of the products were sold. For example, the number of times where products A and D appear together (namely the support value of A and D together) is: 1101100 AND 0010100 = 0000100. The list below shows all the established cases where a pair of products has a united support value greater than or equal to the threshold value 2 (Large 2 itemset/ Frequent 2 itemset).

A AND C	=	1101100	AND	1001000	=	1001000
A AND E	=	1101100	AND	1100110	=	1100100
A AND F	=	1101100	AND	1010100	=	1000100
D AND F	=	0010100	AND	1010100	=	0010100
E AND F	=	1100110	AND	1010100	=	1000100
E AND I	=	1100110	AND	0010001	=	0100010

Phase (3) association graph construction: After the above phase, based upon the product items whose support values are greater than or equal to the threshold value, we can do the "AND" operation on each pair of products, find the Large 2 itemsets whose bit vectors are greater than or equal to the threshold value and construct the association graph according to the product serial numbers we derive in the first phase (I AND I+1). For example, after doing the AND operation on products A (1101100) and E (1110110), we get this bit vector 1100100. Since it exceeds the support threshold 2, we build a directional connection between (1) and (5). Doing likewise to all the other products, we can get the complete association graph (Fig. 2).

Phase (4) association pattern generation: In this phase, our job is basically to find product combinations from the association graph that are composed of two or more than two product items. In other words, we have to find the product item sets each of which appears in at least two of the trade records in the database with the support value greater than or equal to the threshold value (Note: If such a product item set has three items, then it is called a Large 3 itemset. By the same token, a Large 4 itemset is such an item set that includes four items). For example, in the association graph (Fig. 2) we can find two consecutive association lines that point from product A to product E and from product E to product F, respectively. That means the item set (A, E) has a support value greater than or equal to the threshold value and so does the item set (E, F). In other words, we have a reason to expect association rules to be established for these two item sets. This way, if we want to make sure whether the item set of three (A, E, F) is a Large 3 itemset, all we have to do is compute (A AND E) AND (E AND F). In this example, (A AND E) AND (E AND F) = (1101100 AND 1110110) AND (1110110 AND 1000100) = 1100100 AND 1000100 = 1000100. Since the number of times the digit "1" appears is equal to the threshold value 2, we know that the item set (A, E, F) is a Large 3 itemset. In Fig. 2, the Large 3 itemset is indicated by the dotted directional line.

Revised graph-based mining association rules: Yen and Chen[8] have extended the graph-based association rule mining method into the field of multiple level association rule mining as well as generalized association rule mining. However, when this graph-based method is applied to the mining of multiple level association rules, it has to repeatedly give serial numbers to the nodes on every conceptual hierarchical level before it can find the Large 1 itemsets, set up the association graph, generate the association pattern and finally extract the multiple level association rules desired.

Among the phases, phase (1), which is in charge of serial number production and phase (2), which takes care of Large 1 itemset generation, are the most resource- and time-consuming. Why? Because the hardest work of assigning one unrepeated serial number to each and every product item seen in the trade records, executing the exhaustive scan all through the database, recording the order in which all the product items appear in the trade records and then giving every product item its bit vector whose length is equal to the number of trade records there are in the database, must be completed in these two phases. In case the conceptual hierarchy is composed of three hierarchical levels, then that means all the above processes have to be executed three times if all the multiple level association rules are to be extracted.

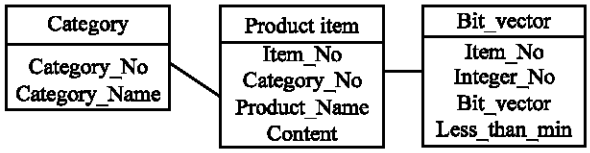

Fig. 3: Connections between the category, the product information and the bit vector table

```
INSERT INTO concept_bit_vector
SELECT A.item_no, A.integer_no,
A.bit_vector, A.less_than_min,
B.product_brand, B.content, C.category_name
FROM tb_bit_vector A,
bas_item_master B,
bas_item_catgory C
WHERE (A.item_no = B.item_no)
AND (B.catgory_no = C.catgory_no)
ORDER BY C.category,
B.content, B.product_brand,
A.integer_no
```

Fig. 4: Using the SQL command to SELECT data sets

As we mentioned earlier, in this study, our focus is on the revision of the graph-based association rule mining technique. Our new "revised graph-based association rule mining method" needs only one database scan to determine the bit vector for each of the product item even when it is dealing with multiple level association rule mining. After that, association rule mining of any conceptual level and any number can be done without doing repeated database scans for every conceptual level.

Let us take the trade records (Table 1) for example. With the illustration of how the hierarchical levels are constructed in Fig. 1, we can use our revised graph-based association rule mining method and see how it can complete the mining of association rules on all the levels after only one database scan. In the conceptual hierarchy, the first conceptual level has the following nodes: milk, drinks, cookies, etc. and the minimum support value is 4. On the second conceptual level, the concept "milk" has categories like low-fat milk, full-fat milk and chocolate milk and the minimum support value here is 3. Then, on the third conceptual level (brand), the elements are Wei-chuan, President, King Car and so on and the minimum support value is also 3.

The reason for setting lower threshold values for lower levels is that lower level concepts tend to appear fewer times in the trade records than higher level concepts. In fact, if we choose threshold values properly and wisely, we can avoid the embarrassment of producing too many useless association rules or ignoring valuable association patterns[11].

Table 2: Integer numbers for product items

Item	Integer_No	Item	Integer_No
A	1	G	7
B	2	H	8
C	3	I	9
D	4	J	10
E	5	K	11
F	6	L	12

Table 3: Bit vector table (tb_bit_vector)

Item	Integer_No	Bit_vector
A	1	1101100
B	2	0010000
C	3	1001000
D	4	0010100
E	5	1100110
F	6	1010100
G	7	0100000
H	8	0100011
I	9	0010001
J	10	0000100
K	11	0000011
L	12	0000001

Way to improve the efficiency

Serial number assignment phase: To each product item that appears in the database of trade records, we assign an integer number unique to itself in arbitrary order. In this example, we give integers 1 through 12 to product items A through L in that order. The results are shown in Table 2.

Large 1 itemset generation phase: The database is scanned once so that the bit vector of every product item can be established and stored in the table "tb_bit_vector" (Table 3). The reason why we keep this bit vector table is that this way we can project the established bit vectors onto the conceptual levels every product item relates to. By doing so, we can spare all the repeated scans through the database.

Repeated use of the bit vector table: The categorization of the product items in the conceptual hierarchy (Fig. 1). If we can do multiple level association rule mining to the same trade records, then we can derive broader association rules and get information that is more useful. According to the hierarchical structure in Fig. 1, we can go from top to bottom and observe that the category "milk" is conceptually divided according to the content into such product items as low-fat milk, full-fat milk and chocolate milk. Besides that, each product item can possibly come from more than one brand. In this association-based data structure, we pick out the product item master table (bas_item_master), which keeps track of every detail about the product items and the product item category table (bas_item_catgory), which records the categorization of the product items, as our two conceptual

levels to work on. The reason is that the "bas_item_master" table includes the lowest level concept-brands-while the "bas_item_category" table has the concept that belong to the second level-categories.

Then, we do the "join" operation and put together the bit vector table (tb_bit_vector in Table 3), the product item master table (bas_item_master) and the product item category table (bas_item_catgory) (Fig. 3). For instance, in Fig. 4, we use the SQL command to select desired data sets and put them into a new table concept_bit_vector. After the connections are made, we can proceed with further multiple level association rule analyses.

Joining the tb_bit_vector table with the product item category table, we get the attributes of the items about their categories; joining the tb_bit_vector table with the product item master table, we get the information as to the contents and brands of the product items. When dealing with multiple level association rule analysis, Yen and Chen's graph-based association rule mining technique[8] has to execute one serial number assigning procedure for every conceptual level so as to derive the Large 1 itemsets and the association rules on that level. In contrast, our revised graph-based association rule mining technique needs only one serial number assignment and one Large 1 itemset generation process to accomplish the task.

First conceptual level: First of all, we analyze the association rules on the first conceptual level: category. Using the SQL command (SELECT DISTINCT category FROM concept_bit_vector), we come to the conclusion that there are such independent attributes as milk, instant noodles, drinks, cookies, candy and buns on the first conceptual level (Table 4). For every category, if it has more than one bit vector under it in the concept_bit_vector table, then we do the OR operation to the bit vectors of all the product items under this category and take the result as the support value of this category; otherwise, when the category has only one bit vector under it, then the support value of this category is equal to the number of the "1" digits in the bit vector.

$$\forall \text{concept value} \begin{cases} \text{logic or all bit vector, if count} > 1 \\ \text{bit vector, if count} <= 1 \end{cases}$$

For example, product items that belong to the category "milk" include items A, B, C and D. Doing the OR operation to the bit vectors of items A, B, C and D, we get the result that the support value of the category "milk" is 1101100 OR 0010000 OR 1001000 OR 0010100 = 1111100. The total number of "1" digits in 1111100 is 5, meaning that the support value of the category "milk" is 5. Since the value is greater than the preset threshold value 4, the category "milk" is a Large 1 itemset on this conceptual level. On the other hand, the category "instant noodles" has only one record; in other words, there is no need for any OR operation. The support value of this category is the original bit vector, namely 1.

Table 4: Bit vector table plus category plus content plus brand (concept_bit_vector)

Item	Integer No	Bit vector	Category	Content	Brand
A	1	1101100	milk	chocolate	Wei -chuan
B	2	0010000	milk	chocolate	President
C	3	1001000	milk	coffee	Wei -chuan
D	4	0010100	milk	coffee	President
L	12	0000001	instant noodles	chocolate	Hei -song
E	5	1100110	drinks	chocolate	Wei -chuan
F	6	1010100	drinks	coffee	Wei -chuan
G	7	0100000	drinks	coffee	President
H	8	0100011	cookies	coffee	Hei -song
K	11	0000011	candy	coffee	King Car
I	9	0010001	buns	chocolate	Wei -chuan
J	10	0000100	buns	chocolate	Hei -song

Table 5: Support values of items on the first conceptual level

Category	Bit vector	Support value
milk	1101100 OR 0010000 OR 1001000 OR 0010100 = 1111100	5
drinks	1100110 OR 1010100 OR 0100000 = 1110110	5
cookies	0100011	3
candy	0000011	2
buns	0010001 OR 0000100 = 0010101	3

Table 6: Bit vector table for items in categories whose support values are greater than or equal to the threshold 4

Item	Integer No	Bit vector	Category	Content	Brand
A	1	1101100	milk	chocolate	Wei- chuan
B	2	0010000	milk	chocolate	President
C	3	1001000	milk	coffee	Wei- chuan
D	4	0010100	milk	coffee	President
E	5	1100110	drinks	chocolate	Wei- chuan
F	6	1010100	drinks	coffee	Wei- chuan
G	7	0100000	drinks	coffee	President

This way, we can figure out the support values of all the items (categories) on the first conceptual level and organize them into the Table 5.

Among the categories, only "milk" and "drinks" have greater support values than the threshold value 4, so we keep the two and ignore the others. Then, we check the item set "milk and drinks" and count the number of times they both appear in a trade record to see if this item set has a support value greater than the preset threshold 4, namely to see if it is a Large 2 itemset. We can simply do the AND operation to the bit vectors of the two and get the following result:

$$\text{milk AND drinks} = 1111100 \text{ AND } 1110110 = 1110100$$

As a result, the support value of "milk and drinks" is 4, which is equal to the preset threshold. Therefore, as far as the first conceptual level is concerned, "milk and drinks" is a Large 2 itemset. Finally, we get the graph of the first conceptual level in the hierarchy (Fig.5).

Fig. 5: Graph of the first conceptual level

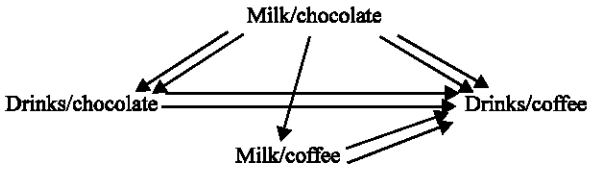

Fig. 6: Association graph of the second conceptual level

Second conceptual level: Since we have got the graph of the first conceptual level, we can simplify Table 5 into Table 6 on the right. The rationale here is that, if a parent node does not appear often in the database, then its child nodes must be even less often seen. For example, in our case, the support value of the category "cookies" is lower than the threshold value. Therefore, we can be sure that the product item "chocolate cookies," which is a child node of "cookies," can in no way be a frequently seen item.

In the same way, we can combine categories with contents and get on the second conceptual level. After computing the support values on the two conceptual levels, we can derive Table 7 from Table 6.

As Table 7 shows, in our example, we have four Large 1 itemsets: milk/chocolate, milk/coffee, drinks/chocolate and drinks/coffee, whose support values are either greater than or equal to the preset threshold 3 for the second conceptual level. The following operations are for the support values of the combinations of any two Large 1 itemsets.

milk/chocolate AND milk/coffee = 1111100 AND 1011100 = 1011100
milk/chocolate AND drinks/chocolate = 1111100 AND 1100110 = 1100100
milk/chocolate AND drinks/coffee = 1111100 AND 1110100 = 1110100
milk/coffee AND drinks/coffee = 1011100 AND 1110100 = 1010100
drinks/chocolate AND drinks/coffee = 1100110 AND 1110100 = 1100100

From the above results, we learn that these combinations "milk/chocolate and milk/coffee," "milk/chocolate and drinks/chocolate," "milk/chocolate and drinks/coffee," "drinks/chocolate and drinks/coffee," and "milk/coffee and drinks/coffee" satisfy the requirement of the minimum support on the second conceptual level. According to the results, we derive the association graph of the second conceptual level (Fig. 6).

Then, based on the association graph in Fig. 6, we can execute the following operations and find that

Table 7: Support values on the second conceptual level

Category	Bit_vector	Support
milk/chocolate	1101100 OR 0010000 = 1111100	5
milk/coffee	1001000 OR 0010100 = 1011100	4
drinks/chocolate	1100110	4
drinks/coffee	1010100 OR 0100000 = 1110100	4

Table 8: Support values on the third conceptual level

Category	Bit_vector	Support
milk/chocolate/Wei-chuan	1101100	4
milk/chocolate/President	0010000	1
milk/coffee/Wei-chuan	1001000	2
milk/coffee/President	0010100	2
drinks/chocolate/Wei-chuan	1100110	4
drinks/coffee/Wei-chuan	1010100	3
drinks/coffee/President	0100000	1

"milk/chocolate, milk/coffee and drinks/coffee" and "milk/chocolate, drinks/chocolate and drinks/coffee" are Large 3 itemsets. In the association graph, Large 3 itemsets are represented by dotted directional lines.

milk/chocolate AND milk/coffee AND drinks/coffee = 1011100 AND 1010100 = 1010100
milk/chocolate AND drinks/chocolate AND drinks/coffee = 1100100 AND 1100100 = 1100100

Third conceptual level: The third conceptual level is "category/content/brand." Table 8 shows the support values on the third conceptual level.

According to the results in Table 8, the Large 1 itemsets, whose support values are either greater than or equal to the preset threshold value 3 for the second conceptual level, are "milk/chocolate/Wei-chuan," "drinks/chocolate/Wei-chuan," and "drinks/coffee/Wei-chuan."

milk/chocolate/Wei-chuan AND drinks/chocolate/Wei-chuan = 1101100 AND 1100110= 1100100

According to the above result, we derive the association graph on the third conceptual level (Fig. 7).

In the above example of our revised method of multiple level association rule analysis, observing the concept on the first conceptual level-category, we learn the pattern that customers tend to buy drinks at the same time they purchase milk (Fig. 5).

Buys (x, "milk") ? Buys (x, "drinks")

However, such a message is too weak for marketing project designers to develop aggressive and powerful plans from. They need stronger, more complete, more concrete association patterns. Therefore, we follow the lead of the hierarchy down to the lower levels-the contents and the brands-to find the association rules that govern the relationships between milk and drinks and their contents as well as between the contents and the

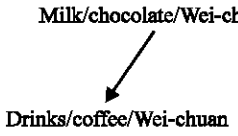

Fig. 7: Association graph on the third conceptual level

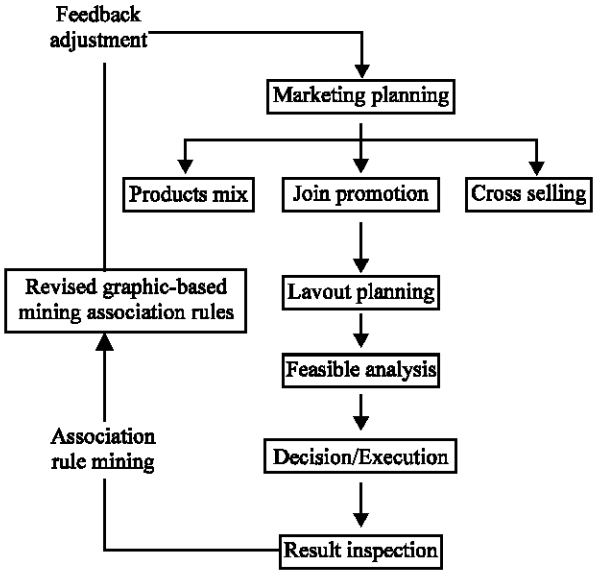

Fig. 8: Information feedback mechanism of a dynamically learning business organization

brands. On the conceptual level of contents, we find the following association rule (Fig. 6):

Buys (x, "chocolate milk") ? Buys (x, "coffee drinks")
Buys(x, "chocolate milk") AND Buys (x, "chocolate drinks") ? Buys (x, "coffee drinks")
Buys (x, "chocolate milk") AND Buys (x, "coffee milk") ? Buys (x, "coffee drinks")
For the categories "milk" and "drinks," on the conceptual levels of "contents" and "brands," we find that Wei-chuan
chocolate milk and Wei-chuan coffee drinks tend to be bought at the same time (Fig. 7).
Buys (x, "Wei-chuan chocolate milk") ? Buys (x, "Wei-chuan coffee drinks")

These consumption behavior patterns involve association rules across different conceptual levels and therefore can be strong reference for the design of marketing plans.

In a fully electronic organization of knowledge management, when facing stronger and stronger competitions, the current knowledge must be continuously transformed into future competitive power through organizational learning, because only by doing so can a business carry on. In fact, our revised graph-based, multiple-level association rule mining technique is right the best feedback mechanism for an organization of knowledge management. By exploring the trade records in the database, we can get more feedback than just how good the sales are or what the best/worst seller is. By means of our revised multiple level association rule mining technique, we can find the connections between product items, between categories, between product contents and even connections across conceptual levels of brands, contents and categories. Therefore, with our help, organizations of knowledge management will no longer have to rely solely on traditional empirical experiences to speculate about changes of consumer preferences. They can use our multiple level association system to gather any feedback information they need at any time from their sale records in order to design multi-product, multi-brand marketing promotion plans and consumption streamlines. Then, after executing their plans, businesses can use our multiple level association system again to examine the feedback. This way, they can form a "promotion plan-practice-result inspection-feedback adjustment" circle (Fig. 8). Keeping running in this circle, any business can become a self-learning organization of knowledge management capable of dynamically adjusting its marketing strategies to adapt itself to the changes of customer behaviors and thereby strengthening its competitiveness.

Due to the internationalization of the domestic business environment nowadays, competitions that every company has to survive have come not merely from the challenges of other local companies but from everywhere around the world. Under such circumstances, facing unfavorable factors such as customers of low degrees of loyalty and swift changes of industrial structures, business managerial levels seem to have no choice at all but to take full advantage of state-of-the-art information processing techniques when crucial decisions are to be made for their companies to offer real-time responses and satisfy the market. In order for organized knowledge to contribute the most to business interest so that companies can stay highly competitive, optimized, standardized and flexible knowledge databases must be developed. Through organizing properly sized, workable knowledge database structures, companies can promote various projects as to the development of their information-related internal affairs, nurturing intelligent, innovative, knowledge-friendly organization culture. With a view to exploring how to digitally turn knowledge into real value for business organizations, in this study, we shall focus on the establishment of revised graph-based mining association rules.

With the revised graph-based mining association rules we offer, all we need is one database scan process. After the scan, the bit vector corresponding to each product can be established and any association rule can then be arbitrarily picked out for rule mining. This way, exhaustive, repetitive scans over the whole database will no longer be necessary. Our revised graph-based mining association rules are quite so flexible that they can be adaptively used for distinct association rule mining in distinct fields. In this study, the association rules will be applied to the development of electronic businesses whose core is knowledge management. By dealing with the knowledge management and procedures as a whole, business organizations can expect to see improved managing efficiency and smoother working procedures. In addition, details as to product design and production, service, as well as delivery can also be much more effectively planned and efficiently controlled.

As for the constraints, our revised graph-based association rule mining technique, just like Yen and Chen's method, has an efficiency bottleneck to break through. Instead, the temporary results in the middle of the procedure have to be stored in the auxiliary memory that has a slower speed. This or course has quite an influence on the efficiency and efficacy of our revised method.

In the future, we shall focus on shortening the length of the bit vector and lowering the complexity of the association graph. For example, in order to make the number of trade records smaller, we can try dividing the database into several sub-bases before we start our revised graph-based association rule mining. Finally, we can put the sub-base graphs together into a whole. This way, we might be able to improve the efficiency and efficacy of our revised graph-based association rule mining technique.

REFERENCES

1. Chen, M.S., J. Han and S. Yu, 1996. Data mining: an overview from a database perspective. IEEE Transactions on Knowledge and Data Engineering, 8: 866-883.
2. Spiegler, I., 2000. Knowledge management: A new idea or recycled concept? Communications of the AIS., 3: 112-137.
3. Liebowitz, J., 1999. Knowledge management handbook. CRC Press.
4. Liebowitz, J., 2001. Knowledge management: Learning from knowledge engineering. CRC Press.
5. Turban, Mclean and Wetherbe, 1999. Information technology for management- making connection for strategic advantage. 2nd Edn., Wiley.
6. Turowski, K., 1999. A virtual electronic call center solution for mass customization. Proceedings of the 32nd Hawaii International Conference on System Sciences, pp: 35-42.
7. Han, J. and M. Kamber, 2001. Data mining: Concepts and techniques. Academic Press.
8. Yen, S.J. and L.P. Chen, 2001. A graph-based approach for discovering various types of association rules. IEEE Transactions on Knowledge and Data Engineering, 13: 839-845.
9. Park, J.S., M.S. Chen and P.S. Yu, 1995. An effective hash based algorithm for mining association rules. Proceedings of ACM SIGMOD., pp: 175-186.
10. Srikant, R. and R. Agrawal, 1995. Mining generalized association rules. Proceedings of International Conference on Very Large Databases, pp: 407-419.
11. Han, J. and Y. Fu, 1999. Mining multiple-level association rules in large databases. IEEE Transactions on Knowledge and Data Engineering, 11: 798-805.
12. Srikant, R. and R. Agrawal, 1996. Mining quantitative association rules in large relational tables. Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, pp: 1-12.