

# Principles of Searching for a Variety of Types of Associative Rules in OLAP-cubes

Hlib Horban, Ihor Kandyba, Mykhailo Dvoretzkyi, Anzhela Boiko

*Petro Mohyla Black Sea National University, 68-Desantnykiv St 10, Mykolaiv, 54003, Ukraine*

## Abstract

The classification of association dependencies which can take place among multidimensional data is presented in the article. The representation of templates of inter-dimensional association rules is considered. Generation methods of inter-dimensional and intra-dimensional association rules are presented. Formulas for calculating objective and subjective characteristics of significance of these association rules types are presented.

## Keywords <sup>1</sup>

OLAP, Data Mining, multidimensional data, association rule, support, confidence, lift, leverage, template, dimension, measure, attribute, combination, set, difference.

## 1. Introduction

Technologies of on-line Analytical process (OLAP) [1, 2] and data processing [3, 4] are typically employed in trendy data analysis systems and in decision support systems, that alter additional or less effective knowledge analysis. OLAP technology permits conducting user-defined operation like consolidation, detalization, data slice, cube rotation et al. At the identical time data processing investigates some cumulated hidden knowledge that was unknown before that and will be enough helpful within the data analytics process, upon that knowledge is taken from data sheets pre-spawned likewise by means that of database management systems (DBMS). One of the foremost common tasks of Data Mining is association, that represents detection of regularities between connected objects, an example of which can be the rule that event Y follows event X [5]. X is named a condition or an antecedent, and Y is named a consequent. Rules of that sort are known as association rules. **(Slide 2)**

Data Mining strategies and algorithms [6], together with association rule mining likewise, are chiefly supported processing bestowed in tabular type, wherever sets of analyzed knowledge are settled either in one column or in one line, so that they add one dimension. But such knowledge regularities could happen even in three-dimensional data [7]. If to think about a three-dimensional cube rather than relational table data, then an item set for association rule mining may be bestowed as a collection of attribute values for every dimension, likewise as sets of values within the plurality of dimensions. If strategies and algorithms for association rule mining in relational data are sufficiently researched in publications, then for three-dimensional knowledge such strategies and algorithms don't seem to be nevertheless sufficiently investigated.

In [8] the problem of integration OLAP technology with data processing strategies is taken into account. Specifically, there have been tries to increase the perform of OLAP and share the distributed OLAP server with data processing infrastructure, that resulted in detection of association rules that were present in cubes which were known as Association Rule Cubes. In [9] a shot was created to summarize data in a cube and to increase OLAP operators by algorithms for association rules mining.

---

*IT&I-2020: Information technology and interactions, December 02-03, 2020, Kyiv, Ukraine*


EMAIL: hlib.horban@chmnu.edu.ua (A. 1); jeo2145@gmail.com (A. 2); m.dvoretzkiy@gmail.com (A. 3); angboyko1996@gmail.com (A. 4)

ORCID: 0000-0002-6512-3576 (A. 1); 0000-0002-8589-4028 (A. 2); 0000-0001-5913-6859 (A. 3); 0000-0002-3449-0453 (A. 4)

© 2020 Copyright for this paper by its authors.

Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CC BY

 CEUR Workshop Proceedings (CEUR-WS.org)

A model had conjointly been planned that solved the matter of association rule mining, that conjointly extended SQL language by the operator known as META RULE [10, 11].

A planned technique of association rule mining in information repositories are often observed, that is predicated on the organization of multidimensional data and is capable to extract association rules from many dimensions at completely different levels of abstraction [12], further as generalized version of association rule mining in OLAP cubes, known as Cubegrades [13]. Finally, attention ought to be paid to the approach of directed method of association rule mining in data cubes [14] and software system supported it, referred to as OLEMAR (Online setting for Mining of Association Rules) [15, 19, 20]. But in above-mentioned works the formalized equipment for sleuthing associations in multidimensional data in most cases is taken into account for relative and multidimensional models (ROLAP and MOLAP) or for databases in a specific subject. Therefore, it's fascinating to develop a lot of general tool for mining of inter-dimensional association rules in multidimensional data.

Data analysis poses new tasks in database technology. Their combination will lead to the second generation of database systems, which will allow the creation and management of knowledge bases in the same way as in classical business applications. It would be advisable to combine database, OLAP and Data Mining technologies in a single information system. Such a system would increase the level of intelligence by integrating the aforementioned information technology models. Analysis of the sources of information on the research on the integration of OLAP and Data Mining, implemented in relational DBMS, allows us to conclude that the solution to this problem is far from being complete. Therefore, research aimed at the analysis and integration of these models and technologies is of great interest. This will increase the range of tasks of decision support systems created as part of intelligent information systems.

The aim of the research is to increase the level of intelligence of information systems by creating an instrumental software system for the automated design of multidimensional databases, methods of forming associative rules and their implementation as part of the system, which allows for a much lower cost of designing information-analytical systems.

## 2. The Research

The main elements of OLAP cubes are dimensions and measures. Dimension could be a values sequence some of the parameters to be analyzed. Samples of dimensions is time, geographic location, etc. Typically, dimensions contain extra data that permits users to investigate actual knowledge. Values that are obtained at the intersection of cube dimensions and represent quantifying facts are referred to as measures. Samples of them could also be sales volumes, product balances, etc. [21, 22]. Therefore, flat system is depicted as a hypercube (usually a cube could be a figure containing 3 dimensions, however during this case the quantity of dimensions could also be larger), whose edges are dimensions, and cells are measures. The structure of three-dimensional hypercube is conferred in Fig. 1.

Mathematically the hypercube is suitable to represent by following sets:

D – a set of hypercube dimensions for a specific subject area:  

$$D = \{D_1, D_2, \dots, D_i, \dots, D_n\},$$

where  $D_i$  –  $i^{th}$ -dimension,  $n$  – the quantity of dimensions;

A – a set of attributes (values of elements) of hypercube dimensions:  

$$A = A_1 \cup A_2 \cup \dots \cup A_i \cup \dots \cup A_n,$$

where  $A_i$  – a set of attributes of dimension  $D_i$ , that successively are often diagrammatic as:

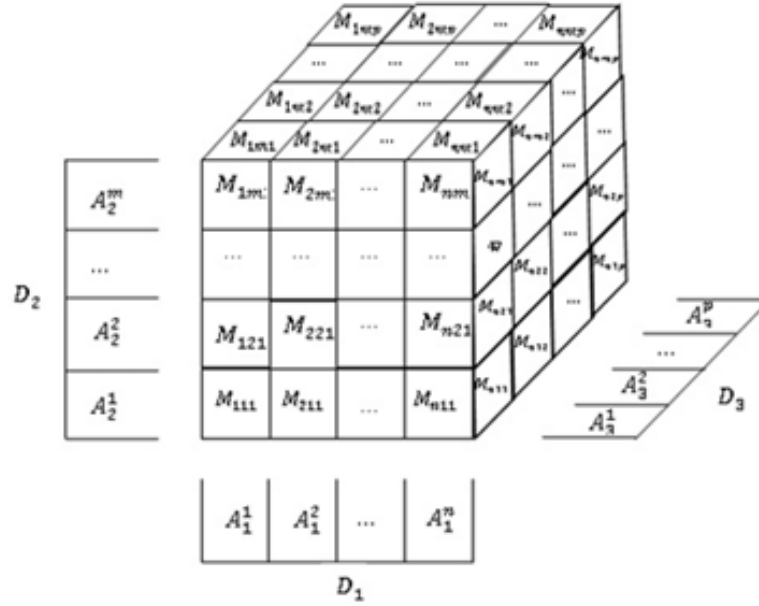
$$A_i = \{A_i^1, A_i^2, \dots, A_i^k, \dots, A_i^m\},$$

where  $k$  – attribute of  $i^{th}$ -dimension,  $m$  – the quantity of attributes in  $i^{th}$ -dimension;

M – a set of values of hypercube measures:

$$M = \{M_{I_1, I_2, \dots, I_i, \dots, I_n}^1, \dots, M_{I_1, I_2, \dots, I_i, \dots, I_n}^l, \dots, M_{I_1, I_2, \dots, I_i, \dots, I_n}^z\},$$

where  $I_i$  – attribute index of  $i^{th}$ -dimension,  $n$  – the quantity of dimensions,  $M_{I_1, I_2, \dots, I_i, \dots, I_n}^l$  – l-measure for the cube cell with  $I_1, I_2, \dots, I_i, \dots, I_n$  index,  $z$  – the quantity of hypercube measures.



**Figure 1.** OLAP multidimensional structure

If to contemplate OLAP cube rather than relational data, then associate item set of association rules will represent a collection of values (attributes) of every dimension. Association rules that arise in multidimensional data will be classified by the subsequent sorts [7, 23]:

1. Inter-dimensional association rules – rules between attributes of different dimensions:

$$(A_i^x \in D_i) \wedge \dots \wedge (A_j^y \in D_j) \rightarrow A_k^z \in D_k,$$

where  $I, J, K$  – corresponding indices of dimensions included into the association rule;  $I, J, K = 1 \dots n$ ;  $n$  – the quantity of dimensions in OLAP cube,  $D_I$  –  $I^{\text{th}}$ -dimension,  $x, y, z$  – corresponding indexes of dimension attributes,  $x, y, z = 1 \dots m_i$ ;  $m_i$  – the quantity of attributes of  $I^{\text{th}}$ -dimension;  $A_i^x$  – corresponding attribute of  $I^{\text{th}}$ -dimension.

2. Intra-dimensional association rules:

$$(A_i^x \in D_i) \wedge \dots \wedge (A_i^y \in D_i) \rightarrow (A_i^z \in D_i) \wedge \dots \wedge (A_i^v \in D_i),$$

where  $I = 1 \dots n$ ,  $n$  – the quantity of dimensions in the cube,  $x, y, z, v$  – certain attribute values of  $I^{\text{th}}$ -dimension,  $x, y, z, v = 1 \dots m_i$ ,  $m_i$  – the total quantity of attributes of  $I^{\text{th}}$ -dimension.

3. Hybrid association rules – dependencies between dimensions, but some operands can be attributes of the same dimension:

$$(A_i^x \in D_i) \wedge \dots \wedge (A_j^y \in D_j) \rightarrow (A_j^v \in D_j) \wedge \dots \wedge (A_k^z \in D_k),$$

In the higher than conferred attributes and belong to the identical dimension of OLAP cube that has  $J$  index [24, 25].

Hybrid association rules may be known as repetition association rules in distinction to different rules thought of, that essentially represent association rules while not repetitions.

Since the dimensions of contemporary databases will reach sufficiently massive volumes (up to gigabytes and terabytes), association rule mining needs economical algorithms that are ascendible and permit to seek out solutions of the task at an inexpensive time. One of such algorithmic rules is Apriori algorithm, 1<sup>st</sup> planned by Srikant and Agraval [16]. Within the original, it's been developed for relational databases and permits generating frequent data sets from group action tables.

The Apriori algorithm uses an iterative approach. Within the start of the algorithm there are one-element frequent data sets that are denoted as  $L_1$ . Within the next step  $L_1$  is employed to seek out frequent two-element sets, from that set  $L_2$  is made, that successively is employed to seek out three-element sets  $L_3$ , and so on, till all doable frequent  $k$ -element sets  $L_k$  are found. So as to extend the potency of generation of frequent data sets, questionable nonmonotonic property is employed, that relies on the subsequent observation: if some data set  $I$  isn't frequent, i.e.  $Supp\{I\} < MinSupp$ , then on

addition of a specific object  $i$  there to, the ensuing new data set additionally won't be frequent:  $Supp\{I \cup \{i\}\} < MinSupp$ .

Using this property, frequent  $k$ -element data sets  $L_k$  can be obtained by combining frequent  $(k-1)$ -element sets. Moreover, for some  $k$ -element set  $l_k$  to be included into  $L_k$  frequent sets, all of its  $(k-1)$ -element subsets also need to be frequent. If at least one of them is not a frequent set,  $l_k$  should be excluded from plurality of frequent item sets<sup>27,28</sup>. This observation facilitates creation of plurality of candidates for  $k$ -element data sets  $C_k$ , which will be superset  $L_k$  that is obtained by withdrawal from  $C_k$  infrequent data sets, and it is the result of checking values of support for each candidate  $c_k$ , ( $c_k \in C_k$ ). On the basis of the nonmonotonic property set  $C_k$  is generated in two steps. In the first step the candidate is generated by combining components of plurality of frequent sets  $L_{k-1}$ , where two members can be combined if they have  $k-2$  common elements, i.e.:

$$L_{k-1} \cup L_{k-1} = \{A \cup B \mid A, B \subset L_{k-1}, |A \cap B| = k - 2\}.$$

In the second step elements, that embody rare  $(k-1)$ -element data sets are deleted from set  $C_k$ . Application of higher than algorithm is additionally potential on dimensional data, which can any facilitate detection of regularities at totally different levels of abstraction. But it's quite natural that for various styles of association rules in OLAP cubes this algorithmic rule can have the suitable varieties.

### 3. Inter-dimensional association rules mining

This type of association rules represents the relationships between attributes of various dimensions. Such rules don't essentially need to embody attributes from all existing dimensions. For instance, the subsequent inter-dimensional association rules might exist:

$$A_I^x \in D_I \rightarrow A_J^y \in D_J.$$

In the general case the minimum variety of dimensions in an association rule is two, and therefore the most variety is that the actual quantity of dimensions in a cube. I.e.  $k=2..n$ , wherever  $k$  is the variety of dimensions in an association rule,  $n$  is the total variety of dimensions in a cube.

If to put the sign of implication between operands elsewhere, then absolutely other association rules will be obtained, which, on the contrary, could have one quantity within the antecedent and several other in the subsequent:

$$A_I^x \in D_I \rightarrow (A_J^y \in D_J) \wedge \dots \wedge (A_K^z \in D_K),$$

and if the quantity of dimensions is quite 3, then the subsequent association rules are often obtained:

$$(A_I^x \in D_I) \wedge \dots \wedge (A_J^y \in D_J) \rightarrow (A_H^v \in D_H) \wedge \dots \wedge (A_K^z \in D_K),$$

in which there are several operands in both components of the association rule.

To facilitate understanding of inter-dimensional association rule mining method, it's better to represent their supposed templates which will dissent from the foundations by the actual fact that solely corresponding dimension is indicated in them rather than specific attribute of a specific dimension, which may be written generally kind as follows:

$$D_i \wedge \dots \wedge D_j \rightarrow D_k \wedge \dots \wedge D_l,$$

where  $D_i, D_j, D_k, D_l \subset D$  – bound dimensions from the set of dimensions,  $i, j, k, l = 1..n$  – bound quantity of cube dimensions,  $n$  – the entire variety of cube dimensions.

It is possible to find a sufficiently large number of association rules templates from the cube, and it would be appropriate to ask how many of them can be found and how to generate all possible templates in order to find association rules already based on them with specific dimension values. Difficulties in respondent this question are often given by the actual fact that for multidimensional cube the entire variety of dimensions will be unknown before. To unravel it, we have a tendency to should take into account all potential templates which will arise between dimensions. It is evident for three-dimensional cube that there are following templates of association rules between them.

Rules between two dimensions:

$$D_1 \rightarrow D_2; \quad D_2 \rightarrow D_1; \quad D_1 \rightarrow D_3; \\ D_3 \rightarrow D_1; \quad D_2 \rightarrow D_3; \quad D_3 \rightarrow D_2.$$

Rules between three dimensions in 3-dimensional cube:

$$D_1 \wedge D_2 \rightarrow D_3; \quad D_3 \rightarrow D_1 \wedge D_2; \quad D_1 \wedge D_3 \rightarrow D_2;$$

$$D_2 \rightarrow D_1 \wedge D_3; \quad D_2 \wedge D_3 \rightarrow D_1; \quad D_1 \rightarrow D_2 \wedge D_3;$$

The larger quantity of dimensions during a cube, the harder is to seek out all potential templates of association rules between them. The look for interdimensional associative rules is mentioned well in papers [15, 16]. Currently we are going to show however the characteristics of the importance of associative rules are calculated.

Denote the measurement value in a particular cube cell as  $M_{\underbrace{i_1, i_2, \dots, i_t, \dots, i_n}_n}$ , where  $i_t$  – an index of  $t$ -cube dimension.

Possible values that can be equal to  $i_t$ :

$0 < i_t \leq n$ , if the  $t^{\text{th}}$  cube dimension contains a mounted value;

$i_t = 0$ , if by the  $t^{\text{th}}$  dimension aggregation takes place.

Based on the above, final cube value are denoted as  $M_{\underbrace{0, 0, \dots, 0}_n}$ .

Denote  $i = \langle i_1, i_2, \dots, i_t, \dots, i_n \rangle$  and  $ALL = \langle \underbrace{0, 0, \dots, 0}_n \rangle$ . Then support of multidimensional set from cube are adequate to the ratio of similar value of the cube measure in the cell having this set to the entire cube value:

$$Sup(i) = \frac{M_i}{M_{ALL}}.$$

In order to feature a particular set to frequent sets, it's necessary that the value of its support is larger or adequate to the user-specified value of the minimum support:

$$Sup_i \geq MinSup.$$

The algorithmic rule for generating all frequent item sets ought to embody generation of  $k$ -element sets, wherever  $k = 1..n$ . The only is generation of single-element sets, since in its implementation it's necessary to travel through all values of attribute of every dimension one by one from one another while not combining them. Generation of frequent single-element sets will be executed exactly as many times as the quantity of combinations with one element  $C_N^1$ , two-element ones  $C_N^2$ , etc. During this case generation of frequent sets with the quantity of components that's larger than one uses lists of sets obtained within the previous step. A standard list of all frequent item sets in multidimensional data forms the idea for generation of inter-dimensional association rules.

Support for the association rule is up to support of the frequent item beset that it's fashioned. This statement becomes apparent because of the very fact that a particular item set forever includes each an antecedent and a ensuant of an associate rule.

Confidence of the association rule multidimensional data can be presented as the ratio of the cube meaning for values of dimensions, denoted along within the antecedent and also the ensuant, to the collective cube meaning by dimensions, that are denoted solely within the antecedent. Now let's present formulas for data calculating of characteristics of the association rule significance for the final case.

Let *AntDim* set embody dimensions that within the condition of a particular rule have specific values, i.e. they're not collective:

$$AntDim = \{D_i, \dots, D_j, \dots, D_k\},$$

where  $D_i$  – dimension, that has index  $i$ ;  $i, j, k = 1..n$ ,  $n$  – the whole variety of dimensions.

Similarly to the set delineated on top of, it's additionally doable to explain *ConsDim* set, which is able to embody dimensions that have specific values as a consequence of the association rule:

$$ConsDim = \{D_l, \dots, D_m, \dots, D_p\},$$

where  $l, m, p = 1..n$ .

One and also the same dimension cannot at the same time be enclosed in each sets, as a result of it can not be enclosed to the antecedent and resultant of the inter-dimensional association rule at the identical time:

$$AntDim \cap ConsDim = \emptyset.$$

The above sets give information only about dimensions included in relevant parts of the association rule. That is only an association rule template can be formed with their help. So as to

create inter-dimensional association rule itself, we'd like sets that contain specific values of corresponding dimensions. We tend to decision them hymenopteran and Cons severally.

Ant set will have the following form:

$$Ant = \{val_{ant_1}, val_{ant_2}, \dots, val_{ant_k}, \dots, val_{ant_n}\},$$

where  $val_{ant_k}$  – value of  $k$ -dimension, which can take the following values:  $val_{ant_k} = x, 1 \leq x \leq t_k$ , if  $k$ -dimension contains fixed value ( $t_k$  – the quantity of values in  $k$ -dimension);  $val_{ant_k} = 0$ , if aggregation is carried out by  $k$ - dimension.

Cons set has similar content:

$$Cons = \{val_{cons_1}, val_{cons_2}, \dots, val_{cons_k}, \dots, val_{cons_n}\}.$$

Similar to support of an item set calculating in multidimensional data, let's denote the ordered set of values of corresponding dimension of a certain cube cell as  $i: i = \langle i_1, i_2, \dots, i_t, \dots, i_n \rangle$ .

Then the formula for calculating the association rule confidence in multidimensional data in the general case takes the following form:

$$Conf(i) = \frac{M_i}{M_{Ant}}.$$

In its turn, if the ordered set of dimension values in the cube cell, in which its full aggregate is located, to mark as ALL:  $ALL = \langle \underbrace{0, 0, \dots, 0}_n \rangle$ , then formulas for calculating subjective characteristics

of the association rule significance for the general case will be equal to:

lift of the inter-dimensional association rule:

$$Lift(R) = \frac{Conf(R)}{Supp(Cons)} = \frac{M_i \cdot M_{ALL}}{M_{Ant} \cdot M_{Cons}};$$

leverage of the inter-dimensional association rule:

$$Lev(R) = Supp(R) - Supp(Ant) \cdot Supp(Cons) = \frac{M_{All} \cdot M_i - M_{Ant} \cdot M_{Cons}}{M_{All}^2}.$$

Generation of all possible inter-dimensional association rules is based on the obtained general list of frequent item sets in OLAP cube.

#### 4. Intra-dimensional association rules mining

Association rules within one dimension can be found in every dimension which is a part of the cube. At the same time one of attributes of the certain dimension may belong to the antecedent, and the rest – to the consequent.

Let's represent the set of corresponding indexes of attributes like *Ant* and *Cons*:

$$Ant = \{x, \dots, y\};$$

$$Cons = \{z, \dots, v\}.$$

The minimum number of dimension attributes in the rule will be 2 when exactly one attribute at a time will be in the antecedent and in the consequent:  $A_I^x \rightarrow A_I^y$ .

In its turn the maximum number of attributes can be their total number in the dimension. To obtain rigorous association rule, it is needed that there would be at least one attribute in the antecedent and consequent, that in this case represents their number:

$$A_I^x \rightarrow A_I^y \wedge \dots \wedge A_I^v;$$

$$A_I^x \wedge \dots \wedge A_I^y \rightarrow A_I^v.$$

The majority of obtained association rules will have the following form:

$$(A_I^x \in D_I) \wedge \dots \wedge (A_I^y \in D_I) \rightarrow (A_I^v \in D_I) \wedge \dots \wedge (A_K^z \in D_K),$$

when there will be more than one attribute of dimension both in the antecedent and consequent.

Thus, it is possible to generate association rules between the number of attributes of one dimension from 2 to  $m: k=2..m$ . In order to begin generation of association rules, at first it is necessary to form *Ant* set, and then to form *Cons* set from the remaining attributes.

*Ant* set can have from 1 to  $k-1$  attributes ( $j=l..k-l$ ). In case of rules with the total number of 2 attributes, *Ant* can have only one attribute. Forming of *Ant* and *Cons* sets will be carried out in the following way (by the example of dimension with 5 attributes):

$$k = 2; j = 1; A_j^1 = \{\{A_j^1\}, \{A_j^2\}, \{A_j^3\}, \{A_j^4\}, \{A_j^5\}\}; P(A_j^1) = C_5^1 = 5.$$

While alternative extracting of one element of  $A_j^1$  set that will contain all possible combinations from  $k$  to  $j$ , *Ant* set will be obtained for a separate rule. Similarly, for a particular rule, *Cons* set is also obtained, but combinations are taken from *B* set, which represents the difference between the total set of attributes and *Ant* set.

For rules in which the antecedent will contain  $A_j^1$  attribute:

$$B_j^1 = A_j^1 \setminus Ant = A_j^1 \setminus \{A_j^1\} = \{A_j^2, A_j^3, A_j^4, A_j^5\};$$

$$B_j^1 = \{\{A_j^2\}, \{A_j^3\}, \{A_j^4\}, \{A_j^5\}\}; P(B_j^1) = C_4^1 = 4.$$

Thus, the following rules will be formed:

$$A_j^1 \rightarrow A_j^2; A_j^1 \rightarrow A_j^3; A_j^1 \rightarrow A_j^4; A_j^1 \rightarrow A_j^5.$$

Other association rules can also be generated by the same procedure within *I* dimension and between two of its attributes, in the antecedent of which there will be other attributes. By the method described above it is possible to find rules between greater numbers of attributes of a certain dimension.

Association rule support within one dimension will be the ratio of the sum of all measures in the dimension under analysis and attributes that belong both to condition and consequence, to the final value of the measure (ALL) by the appropriate dimension:

$$Supp = \frac{\sum_{k=0}^p M_{Ant_k, a, b} + \sum_{l=0}^r M_{Cons_l, a, b}}{M_{ALL, a, b}}.$$

In its turn, the confidence of the association rule will be equal to the ratio of the amount of measures containing attributes of the analyzed dimension, that present both in the condition and consequence, to the sum of the measures containing attributes of dimension, which are present only in the condition:

$$Conf = \frac{\sum_{k=0}^p M_{Ant_k, a, b} + \sum_{l=0}^r M_{Cons_l, a, b}}{M_{Ant_k, a, b}}.$$

Appropriate formulas for calculating an elevator and leverage can be obtained from the formulas for calculating support and confidence.

However, it should be noted that the formulas described above are not universal yet, since they represent only an example of calculating of relevant characteristics of significance for rules within the first dimension of a three-dimensional cube at constant values in other two dimensions (*a* and *b* respectively). The presentation of a universal mathematical apparatus for calculating of characteristics of the association rule significance is included into author's further plans.

As for values of other cube dimensions in such association rules, varieties can exist here. Association rules within one dimension can be complete and contextual.

A complete association rule will represent the relationship between attributes of one dimension at final values of other dimensions. I.e. the formulas for calculating of the support and confidence for the above example will look like these:

$$Supp = \frac{\sum_{k=0}^p M_{Ant_k, ALL, ALL} + \sum_{l=0}^r M_{Cons_l, ALL, ALL}}{M_{ALL, ALL, ALL}};$$

$$Conf = \frac{\sum_{k=0}^p M_{Ant_k, ALL, ALL} + \sum_{l=0}^r M_{Cons_l, ALL, ALL}}{M_{Ant_k, ALL, ALL}}.$$

Accordingly, incomplete association rule in dimensions by which search of dependencies is not carried out will have certain meanings. Formulas for support and confidence calculating of such association rules have already been presented above.

## 5. Hybrid association rules mining

The difficulty in hybrid association rules mining lays particularly in that some operands belong to one dimension, i.e. it is necessary to check the dependencies not only between data of different dimensions, but also between data within the same dimension.

In the presence of repetitions of the same dimension among data in operands of the association rule, there can be two cases:

- 1) operands with attributes of one dimension refer to one part of the rule (condition or consequence);
- 2) operands with attributes of one dimension refer to different parts of the rule, which means existence of an association rule within the same dimension inside hybrid association rule.

It should also be noted that the number of dimensions, that have more than one attribute among operands of the rule, may be multiple.

Generation of hybrid association rules should be carried out in two steps:

- 1) search for dependencies between dimensions;
- 2) search for dependencies within one dimension.

Dependencies between dimensions can be represented in the form of templates of the following form:

$$D_i \wedge \dots \wedge D_j \rightarrow D_k \wedge \dots \wedge D_l.$$

By substituting specific attributes into dimensions, corresponding instances of association rules will be obtained.

In its turn, search for dependencies within one dimension is carried out among data included in the set of attributes of corresponding dimension. In this case the number of operands in such association rules can range from 2 to the total number of attributes in the dimension:

$$(A_i^x \in D_i) \wedge \dots \wedge (A_i^y \in D_i) \rightarrow (A_i^v \in D_i) \wedge \dots \wedge (A_i^z \in D_i).$$

Development of methods for hybrid association rule mining is also included into author's further plans.

## 6. Generation of Associative Rules in OLAP Cubes

As modern databases can be very large (up to gigabytes and terabytes), you need efficient algorithms to find reflection rules that can be scaled up and that will allow you to find a solution within a reasonable time.

One such algorithm is Apriori, first proposed by Sricant and Agraval [26]. Originally it was developed for relational databases and allowed the generation of frequent data sets from transaction tables.

Frequent subject set in multidimensional data means a set of attribute values for the relevant measurements, the value for which is below the threshold for the minimum support value, which is set by the end user based on his own experience.

When setting the task of searching for frequent subject sets in multidimensional data, the following feature can be highlighted: in the OLAP cube, you can find such frequent sets that belong to completely different sets. This is due to the fact that when considering multivariate data, completely different dimensions of the cube are processed and then their combinations are combined.

This results in frequent subject sets from data first with one dimension, then with two, etc. Finally, frequent subject sets can be found with n dimensions, where n is the total number of measurements in a cube.

In general, let the set of all frequent sets of topics in the OLAP cube be a set of S:



$$S = \{S_1, S_2, \dots, S_i, \dots, S_n\},$$

where  $i$  is the number of elements in a subject set,  $S_i$  is a lot of frequent subject sets with the number of elements and,  $n$  is the total number of elements in a cube.

In turn, sets of  $S_1, \dots, S_n$  contain different subject sets for each of the measurements or sets of measurements if the number of elements in the set is greater than one.

In other words:

$$S_1 = \{s_1, s_2, \dots, s_n\},$$

where  $s_1$  is a set of frequent single element subject sets in the first dimension of the cube,  $s_2$  in the second dimension and  $s_n$  in the  $n$  dimension.

In turn, many two-element subject sets can be presented as follows:

$$S_2 = \{s_{12}, s_{13}, \dots, s_{mn}\},$$

where  $s_{12}$  is a set of frequent subject sets for the first and second dimensions,  $s_{13}$  for the first and third dimensions,  $m \neq n$ .

Let  $k$  be the number of elements in the subject set. So, in general:

$$S_k = \bigcup_{i=1}^{C_n^k} \{s_{i_1, i_2, \dots, i_k}\}.$$

It is clear that when creating frequent OLAP cube subject sets, they will not include all the elements included in the corresponding cube measurements. To include an element or a collection of them in such sets, you must first calculate the support for that collection.

It is proposed to create a frequent subject set in the form of a list, where the first element is a sublist containing the sequence numbers of cube measurements according to which the set is generated [24]. in a single element set, such a list contains only one element. This sublist in the first element in the further generation of associative rules based on subject sets is necessary to identify the measurements for which all sets have been created.

All subsequent items on the list, i.e. from the second to the last item, will contain information about the specific item set found.

In general, the list should be in the following format:

$$\langle \langle \underbrace{\langle id_1 \rangle, \langle id_2 \rangle, \dots, \langle id_k \rangle}_k \rangle, \underbrace{\langle val_{11}, val_{21}, \dots, val_{k1} \rangle}_k \& Supp_1 \rangle, \dots, \underbrace{\langle val_{1z}, val_{2z}, \dots, val_{kz} \rangle}_k \& Supp_z \rangle \rangle,$$

where  $k$  is the number of elements in the subject set,  $id_i$  is the ordinal number of the  $i$ -th cube measurement in the corresponding subject set,  $val_{ij}$  is the value of the  $i$ -th cube measurement attribute in the corresponding  $j$ -th subject set,  $Supp_j$  is the value of  $j$ -th subject set support,  $z$  is the obtained number of frequent subject sets.

Cube elements  $id_1 \dots, id_i \dots, id_k$  play the role of so-called keys which are used to obtain values in subject sets.

The next step is to create frequent two-element subject sets, whose elements belong to different dimensions of the cube. This generation is done on the basis of already generated single element sets used as function parameters, actually generating sets of two elements.

When one of these sets is obtained from two sets of one element, the following procedure must be followed:

1. Find the measurement identifiers of one and two single-element sets ( $id_1$  and  $id_2$  respectively) by extracting them from the first elements of the respective lists:

$$\begin{aligned} list_1 &= \langle \langle id_1 \rangle, \underbrace{\langle val_{11} \rangle \& Supp_1 \rangle, \dots, \langle val_{1z} \rangle \& Supp_z \rangle}_z \rangle; \\ list_2 &= \langle \langle id_2 \rangle, \underbrace{\langle val_{21} \rangle \& Supp_1 \rangle, \dots, \langle val_{2z} \rangle \& Supp_z \rangle}_z \rangle. \end{aligned}$$

then, based on  $id_1$  and  $id_2$  values, a corresponding list of measurement identifiers must be generated and included in the new subject set in the future:

$$DimId = \langle \langle id_1 \rangle, \langle id_2 \rangle \rangle.$$

Next, a blank list of candidates must be created for frequent subject sets and a Cartesian work operation must be performed for two sets of one element, respectively transferred as arguments (i.e. list elements start with the second, etc.). The result of this operation must then be placed on the candidate list:

$$L_{1,2} = L_1 \times L_2.$$

The L1 and L2 sets will contain the corresponding measurement values in the single element sets:

$$L_1 = \{val_{11}, \dots, val_{1z}\};$$

$$L_2 = \{val_{21}, \dots, val_{2z}\}.$$

This means that in order to obtain these sets, it is necessary to analyse list1 and list2 list elements starting from the second as rows, rejecting the value that is saved after the "&" sign.

As a result, the list of candidates for frequent recruitment will look like this:

$$cands = \langle\langle val_{11}, val_{21} \rangle, \langle val_{11}, val_{22} \rangle, \dots, \langle val_{1z}, val_{2z} \rangle\rangle.$$

At this stage, the value for maintaining candidate recruitment is not calculated.

2. Create a list from the list of candidates for frequent sets, by calculating a value to support each set. The new list will include sets that have less than the minimum support. The first element will be the *dimId* list. This is to ensure that all frequent sets lists, regardless of the number of elements in the received sets, have the same format already mentioned above, since the corresponding list of frequent two element sets will be used as a parameter in the generation of three element sets.

Thus, the current list will enter approximately the following type (excluding sets whose support is below the minimum value):

$$itemset = \langle DimId, \langle val_{11}, val_{21} \rangle \& Supp_1, \dots, \langle val_{1z}, val_{2z} \rangle \& Supp_z \rangle.$$

Generation of sets from one and two elements was considered separately and is performed using algorithms that are not similar to the algorithm of generating frequent subject sets where the number of elements exceeds two. From three elements and so on until the maximum number of elements in a set corresponds to the total number of measurements in a cube, the generation of frequent subject sets is done on the same principle.

Consider the generation of sets with k elements from the OLAP cube ( $k = 3 \dots n$ ). This procedure is only possible when sets with k-1 elements have already been generated, as it is done with two different sets, the number of elements in which is less than one element. Through a single list format, regardless of the number of elements in the sets, their first element is a list containing the identifiers of the measurements for which the respective sets have been created. A precondition for the possibility of generating one set with k elements from two sets of k- 1 elements is a common dimension or a combination thereof (if  $k > 3$ ), which have both sets with k-1 elements.

Thus, when generating a set with k elements, two sets of k-1 elements are used, as mentioned above, connected by one set of k-2 elements, the measurements of which are common to both.

For clarity and user-friendliness, we present a generation algorithm of *k-element* sets in the example if  $k = 3$ , i.e. two sets of two elements and one set of one element are needed for the generation. For values of k that are different from 1 and 2, this algorithm will work similarly.

Let us denote lists containing two-element sets as list1 and list2, and a list from a single-element set as sublist. Let a, b, c represent the variables whose values correspond to the measurement identifiers that will be included in the new generated set. Regardless of the number of elements in the sets, their first element is a sublist that contains the measurement identifiers for which the corresponding sets have been created. These lists will therefore be approximately the same:

$$list_1 = \langle\langle a, b \rangle, \underbrace{\langle val_{a1}, val_{b1} \rangle \& Supp_1, \dots, \langle val_{ax}, val_{bx} \rangle \& Supp_x}_{x} \rangle;$$

$$list_2 = \langle\langle b, c \rangle, \underbrace{\langle val_{b1}, val_{c1} \rangle \& Supp_1, \dots, \langle val_{by}, val_{cy} \rangle \& Supp_y}_{y} \rangle;$$

$$sublist = \langle\langle b \rangle, \underbrace{\langle val_{b1} \rangle \& Supp_1, \dots, \langle val_{bz} \rangle \& Supp_z}_{z} \rangle.$$

The first step of the algorithm is to search for measurement identifiers for the respective sets by obtaining the first elements from the above lists. Let us denote these elements according to *idlist<sub>1</sub>*, *idlist<sub>2</sub>* and *idsublist*, which in the context of the example in question have the following values:

$$idlist_1 = \langle a, b \rangle; \quad idlist_2 = \langle b, c \rangle; \quad idsublist = \langle b \rangle.$$

On the basis of the received measurement lists, a list of measurement IDs of the set sought is formed with  $k$  elements, which includes all elements of the  $idlist_1$  list, and each element of the  $idlist_2$  list is checked, if it is already present in the new list, if not, it will be added to it. The list of identifiers for the new subject set thus takes on the following form:

$$DimId = \langle a, b, c \rangle.$$

Further consolidation of the two main sets (list1 and list2) with a binder (sublist) has this feature:

- compared to the list of measurement IDs of the first formation set and the binder set, only the first element  $idlist_1$  differs from the content  $idsublist$ ;
- only the last element of  $idlist_2$  differs from the content of  $idsublist$  in comparison to the measurements of the binder set of the second formation set.

## 7. Conclusion and perspectives of further research

Among multidimensional data similar to tabular one, it is possible to find certain association dependencies represented in the form of rules that can be classified as inter-dimensional, within one dimension and hybrid. The approach to construction of templates of inter-dimensional association rules is proposed by generating all possible combinations of dimensions in OLAP-cube, which allows obtaining possible association rules, as well as the approach to construction of association rules within one dimension by generating all possible combinations of values of a certain dimension, among which search for dependencies is carried out. Appropriate methods have been developed for generating inter-dimensional association rules and association rules within one dimension. In the future, it is planned to study methods of hybrid association rule mining among multidimensional data.

## 8. Acknowledgements

This research was partially supported by the state research projects: “Development of information and communication decision support technologies for strategic decision-making with multiple criteria and uncertainty for military-civilian use” (research project no. 0117U007144, financed by the Government of Ukraine); “Development of information-analytical system for military-civil application as a information protection factor in the conditions of multi-criteria, uncertainty and risk” (research project no. 0120U101222, financed by the Government of Ukraine).

## 9. References

- [1] E. Thomsen, [OLAP Solutions: Building Multidimensional Information Systems], John Wiley & Sons, New York, 2002, pp. 50–688.
- [2] R. Wrembel, C. Koncilia, [Data Warehouses and OLAP: Concepts, Architectures and Solutions], Idea Group Inc., 2007, pp. 12–237.
- [3] D. Hand, H. Mannila, P. Smyth, [Principles of Data Mining], Massachusetts Institute of Technology, Cambridge, 2001, pp. 2–546.
- [4] N. Ye. (Ed.), [The Handbook Of Data Mining], Lawrence Erlbaum Associates Publishers, 2003, pp. 254–690.
- [5] C. Zhang, and S. Zhang, [Association Rule Mining: Models and Algorithms], Springer-Verlag, Berlin, 2002, pp. 12–238.
- [6] I. Kovalenko, Y. Davydenko and A. Shved, Formation of Consistent Groups of Expert Evidences Based on Dissimilarity Measures in Evidence Theory, in: Proceedings of the 14th International Conference on Computer Sciences and Information Technologies (CSIT), Lviv, Ukraine, 2019, pp. 113–116. doi: 10.1109/STC-CSIT.2019.8929858
- [7] H. Zhu, [Online analytical mining of association rules. Master’s thesis], Simon Fraser University, Burnaby, 1998, pp. 5–51.

- [8] Q. Chen, U. Dayal, and M. Hsu, An Olap-based Scalable Web Access Analysis Engine, in: Proceedings of the 2nd International Conference on Data Warehousing and Knowledge Discovery (DAWAK'2000), 2000, pp. 210–223.
- [9] S. Goil, and A. Choudhary, High Performance Multidimensional Analysis and Data Mining, in: Proceedings of High Performance Networking and Computing Conference (SC'98), 1998, pp. 126–134.
- [10] R. Meo, G. Psaila, and S. Ceri, A New SQL-like Operator for Mining Association Rules, in: Proceedings of the 22nd International Conference on Very Large Data Bases Conference (VLDB'1996), 1996, pp. 122–133.
- [11] M. Fisun, M. Dvoretzkyi, A. Shved and Y. Davydenko, Query parsing in order to optimize distributed DB structure, 2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Bucharest, 2017, pp. 172-178. doi: 10.1109/IDAACS.2017.8095071
- [12] H. C. Tjioe, and D. Taniar, Mining Association Rules in Data Warehouses, International Journal of Data Warehousing and Mining, 1(3), 2005, pp. 28–62.
- [13] T. Imielinski, L. Khachiyan, and A. Abdulghani, Cubegrades: Generalizing Association Rules, Data Mining and Knowledge Discovery, 6(3), 2002, pp. 219–257.
- [14] R. Ben Messaoud, R. S. Loudcher, O. Boussaid, and R. Missaoui, Enhanced mining of association rules from data cubes, in: Proceedings of the 9th ACM International Workshop on Data Warehousing and OLAP (DOLAP 2006), 2006, pp. 11–18.
- [15] R. Ben Messaoud, R. S. Loudcher, O. Boussaid, and R. Missaoui, OLEMAR: An Online Environment for Mining Association Rules in Multidimensional Data, Data Mining and Knowledge Discovery Technologies, 2, 2007, pp. 1–36.
- [16] A. Symeonidis and P. Mitkas, Agent intelligence through Data Mining, Springer Science+Business Media, Heidelberg, 2005, pp. 3–200.
- [17] M. Fisun, I. Kulakovska, and G. Gorban, Generation of Frequent Item Sets in Multidimensional Data by Means of Templates for Mining Inter-Dimensional Association Rules, The 8th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, vol. 1, 2015, pp. 368–376.
- [18] M. Fisun, and H. Horban, Generation of the Association Rules among Multidimensional Data in DBMS Caché Environment, Advances in Intelligent Systems and Computing. Selected Papers from the International Conference on Computer Science and Information Technologies, CSIT 2016, 2016, pp. 63–79.
- [19] Y. Krainyk, Y. Davydenko, and V. Starchenko, Message-level Decoding of Error Patterns for Turbo-Product Codes, in: Proceedings of the 39th International Conference on Electronics and Nanotechnology (ELNANO), Kyiv, Ukraine, 2019, pp. 660–663. doi: 10.1109/ELNANO.2019.8783849
- [20] Y. Krainyk, Y. Davydenko, and V. Tomas, Configurable Control Node for Wireless Sensor Network, in: Proceedings of the 3rd International Conference on Advanced Information and Communications Technologies (AICT), Lviv, Ukraine, 2019, pp. 258–262. doi: 10.1109/AIACT.2019.8847732
- [21] O. D. Azarov, O. G. Murashchenko, O. I. Chernyak, A. Smolarz, and G. Kashaganova, Method of glitch education in DAC with weight redundancy, Proc. SPIE 9816, 98161T (2015).
- [22] V. S. Osadchuk, and A. V. Osadchuk, The magneticreactive effect in transistors for construction transducers of magnetic field, №3(109), 2011, pp. 119–122.
- [23] I. Kovalenko, Y. Davydenko, A. Shved, Development of the procedure for integrated application of scenario prediction methods. Eastern-European Journal of Enterprise Technologies. Vol. 2, Issue 4 (98), 2019, pp. 31–38. DOI: 10.15587/1729-4061.2019.163871
- [24] V. Vassilenko, V. Valtchev, J. P. Teixeira, and S. Pavlov, Energy harvesting: an interesting topic for education programs in engineering specialities, Conference Proceedings Internet, Education, Science” (IES-2016), 2016, pp. 149–156.
- [25] O. M. Vasilevskyi, M. Y. Yakovlev, and P. I. Kulakov, Spectral method to evaluate the uncertainty of dynamic measurements, Technical Electrodynamics, (4), 2017, pp. 72–78.
- [26] Agrawal, R., Sricant, R. Fast algorithms for mining association rules. In Proc. 1994 Int. Conf. Very Large Data Bases, Santiago, Chile, September 1994, p. 487-499.