ПРОГРЕСИВНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

PROGRESSIVE INFORMATION TECHNOLOGIES

ПРОГРЕССИВНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

UDC 004.658.3

USING THE ANALYTIC HIERARCHY PROCESS WITH FUZZY LOGIC ELEMENTS TO OPTIMIZE THE DATABASE STRUCTURE

Dvoretskyi M. L. – PhD, Senior Lecturer of the Department of Software Engineering, Petro Mohyla Black Sea National University, Mykolaiv, Ukraine.

Savchuk T. O. – PhD, Professor, Professor of the Department of Computer Science, Vinnytsia National Technical University, Vinnytsia, Ukraine.

Fisun M. T. – Dr. Sc., Professor, Professor of the Department of Software Engineering, Petro Mohyla Black Sea National University, Mykolaiv, Ukraine.

Dvoretska S. V. – Senior Lecturer of the Department of Software Engineering, Petro Mohyla Black Sea National University, Mykolaiv, Ukraine.

ABSTRACT

Context. Informational systems are very common and use databases to store information that users need. Many different data models can be used but the relational model is still relevant. The last decade show tendency of using distributed databases while working with relational data model and this approach requires a specially designed module to synchronize data of all separate databases. Considering optimizing the database structure, researchers didn't pay much attention to the potential of users' SQL-queries history. The optimal structure of all the distributed nodes could reduce the necessity of synchronization while the data access speed and its actuality would remain stable. The object of the research is the process of optimizing the structure of the distributed database of corporate information systems, which are based on the relational database's model.

Objective. The research aims at improving the accuracy of the data representation marker's value on the distributed corporate information system's (DCIS) node, obtained using the analytic hierarchy process by applying the fuzzy logic elements while processing the alternatives' global priority vector.

Method. The research's authors in the set of their previous works emphasize the potential of using the collected history of users' SQL queries. Firstly presented technology of users' queries parsing. Then, the idea of using the multidimensional database for analyzing users' queries by slices of workstation type, application, user, and his/her position was considered. Finally, the authors gave the full-scaled mathematical model for formalizing database and query models, and criteria of database structure's optimality.

The current research continues the given sequence and tries to increase the efficiency of the decision support system, by introducing elements of fuzzy logic to the analytic hierarchy process algorithm. The approach's main idea is in presenting the global priorities vector in the form of a series of fuzzy sets of one variable with subsequent transformation to the exact value. This approach made it possible to maintain the accuracy of the obtained result while decreasing the number of solution alternatives.

For new tuples added to the database's tables after all calculations had been performed, the classification problem was formalized. After obtaining the probability of a tuple belonging to the class "needed" and performing the normalization of the value, it is taken as the level of the representation marker. Accordingly, the data is loaded onto the node if this value is greater than the optimal level of the representation marker for the DCIS node.

Results. After calculating and obtaining the alternatives global priorities' vector in order to improve the accuracy of the obtained result, the apparatus of fuzzy sets was used. The obtained vector of global priorities was presented as a vector of fuzzy digits for the data representation marker with subsequent transformation to the exact value. This approach made it possible to maintain the accuracy of the obtained result while decreasing the number of solution alternatives.

Conclusions. While working on the research, the concept of a data representation marker on the DCIS node for the elements of the SQL query model was introduced. An aggregation function has been developed that allows determining the level of need for attributes and tuples in the database's relation for the DCIS node based on the statistics of SQL queries. A model of the dependence of the database structure's optimality criteria on the value of the data representation marker is built. Received further development method of analytic hierarchy process. The initialization of the alternatives' pairwise comparisons matrix can be performed automati-

DOI 10.15588/1607-3274-2022-2-10

cally according to the obtained mathematical models. Representation of the obtained result in the form of the vector of fuzzy numbers with the reduction to the exact value allows increasing the accuracy of the obtained results.

KEYWORDS: corporate information system, database management system, distributed database, SQL-query, data replication, multicriteria problem, analytic hierarchy process, fuzzy logic, classification problem, naive Bayes algorithm.

ABBREVIATIONS

CIS is a corporate information system;

DB is a database;

DBMS is a database management system;

DCIS is a distributed corporate information system;

ORM is an object-relational mapping;

SQL is a structured query language.

NOMENCLATURE

R is a database relation or database table;

$R[P]$ is a database relation's or database table's projection;

$R[S]$ is a table selection;

tup is a tuple of the relation R;

R'' is a result of sequential execution of selection and projection operations to the base relation (table);

$F(tup, S)$ is a conditional boolean function on the tuple of the relation;

R''_{union} is a subset of the relation R that is defined as the union of R'' subsets from the DCIS remote node;

P_{union} is a union of the relation's R attributes from the set of projections $R[P]$;

S_{union} is a union of the relation's R selection conditions from the set of selections $R[S]$;

R_{shema} is a set of relations' attributes;

$R_{primary}$ is a subset of relations' attributes that uniquely identify cortege;

$F_a(Node, A)$ is an evaluation function that determines whether the A attribute is needed on the remote node $Node$;

R^{remote}_{schema} is a subset of the R_{shema}, part of relations' attributes which are presented on the remote node;

$F_{tup}(Node, tup)$ is an evaluation function that determines whether the tup tuple is needed on the remote node $Node$;

$R^{remote-dep}_{data}$ is a set of the relation tuples which are linked from other relations' foreign keys;

Q is a set of dimensions of a user's query;

Q^{inner}_{set} is a set of inner SQL-queries of outer SQL-query Q;

R''_{shema} is a resulting relations' set;

Mrk is a data representation marker that reflects the level of data representation necessity at the DCIS node;

$koef^{node}_{repr}$ is a data representation's threshold coefficient, defined on the range [–1, 1];

$Repr(Node,R,A,tup)$ is a conditional boolean function that determines the necessity of the data slice $<R, A, tup>$ on the remote node $Node$;

$F_{availab}$ is an estimation function that determines the value of the data availability criterion (independence from the central database node);

F_{size} is a value relative value of remote node's database size that is the database size criterion;

$F_{synchro}$ is a need for data synchronization criterion;

W^{global} is an alternatives' global priority vector;

W^{global}_5 is a global priority vector in case of five alternatives;

μ is a belonging function;

a_r is a center of mass.

INTRODUCTION

Nowadays informational systems are everywhere around us, and their users to even notice to use them. This list of examples can be very long, from reading news on the internet, e-commerce, remote education to e-banking, accounting systems, decision support, and so on. All these systems use databases to store information that users need, process it, and present it in appropriate form. Historically several data models are used to present informational system data among which can be mentioned hierarchical, network, relational, object, and document models. Two of them (relational and document) for the set of reasons are particularly popular [1].

Some works positioned relational model as an old and irrational way of storing data while document model is shown to be easily scaled and more productive and sufficient [2]. However, deeper considering which of these two is the most appropriate to use makes it clear that there is no precise answer. It depends on many conditions. For example, among them can be given the necessity of transactional data processing and extracting objects from the database not only by their unique identifier values [3].

Meanwhile, most of the accounting systems were historically built on relational database management systems this fact makes that model particularly useful. The possibility of using object-relational mapping (ORM) technologies [4] also makes the convenience of the relational model almost equal to documental while working with object-oriented methodology.

All this highlighted that the relational model is still relevant but according to the facts of rising the storage capacities, increasing data access speed, using of the ORM technologies, and so on, normalization now is not the primary trend and data duplication can be justified. In this context, the question of database structure optimizing can be viewed from a different angle.

The object of study is the process of optimizing the database structure of the distributed corporate information systems based on the relational data model

The key factor influencing the reliability and availability of the database is the localization of links. The high degree of localization of links can be made by the presentation on the node of the data that is needed exclusively by the current node's users. Database relations are pre-

DOI 10.15588/1607-3274-2022-2-10

sented at the DCIS node after applying the projection and selection operations. That is, for optimal presentation of data it is necessary to use elements of vertical and horizontal data fragmentation.

The subject of study is the analytic hierarchy process for choosing the best alternative of the DCIS node's structure based on the created model of SQL-queries to the relational database

The method allows picking the most optimal decision from the set of alternatives. Increasing the number of alternatives can improve the accuracy of the obtained numeric solution but leads to rising the size of the matrices of pairwise comparison. It was suggested using elements of fuzzy logic while working with the obtained vector of alternatives' global priority.

The purpose of the work is to improve the accuracy of the data representation marker's value on the distributed corporate information system's (DCIS) node, obtained using the analytic hierarchy process by applying the fuzzy logic elements while processing the alternatives' global priority vector.

1 PROBLEM STATEMENT

The last decade shows the tendency of using distributed databases while working with the relational data model. And there are several reasons for this. There are several arias of accounting within one company [5]. For example, it can be a warehouse, human resource, access control, and other types of accounting. The attempt of combining them in one "universal" information system (corporate information systems, CIS) provides a single accounting environment and gives access to all company data for future analysis and decision-making. Nevertheless, presenting all data in one database is connected with the set of potential problems [6], among which productivity, reliability, and safety should be mentioned first.

To solve part of them, the special accounting systems can be separated and each of them will use its own database. This technic is also known in modern application development as the use of the microservices approach in high-load projects [7]. It is clearly understandable that this approach requires a specially designed module to synchronize data of all separate databases.

Company structure also can be geographically distributed and some parts of data have to be presented locally so that data consumers won't rely on the availability of the remote database server. According to this, the set of company databases $D = \{D_1, D_2, \dots D_n\}$, or its subset $D` \subset D$, or maybe some subset of tables $R`_{set} \subset R_{set} = \{R_1, R_2, \dots R_m\}$ should be placed on the local database server and periodically synchronized with the central database version.

Considering optimizing the database structure [8–11], researchers didn't pay much attention to the potential of users' SQL-queries history. In works [9–11] took into account increasing productivity by using the materialized views, database restructuring, and denormalization. But the problem wasn't considered in the context of the single distributed CIS node.

According to the given above, the tendency of using the distributed databases is justified and the need for their parts to be synchronized is clear. And the optimal structure of all the distributed nodes could reduce the necessity of synchronization while the data access speed and its actuality would remain stable. Therefore, the task of optimizing the remote node's database structure is quite relevant.

While defining the objective function, three optimality criteria can be defined. These are independence from the central database node $F_{availab}$,local database size F_{size}, and the need for data synchronization $F_{synchro}$. So the objective function has three input variables $F_{objective}(F_{availab}, F_{size}, F_{synchro})$. Taking into account the goal of minimizing the F_{size} and $F_{synchro}$ criteria, and maximizing the $F_{availab}$, objective function hypothetically can be defined as $F_{objective} = \dfrac{F_{availab} \times W_{availab}}{F_{size} \times W_{size} + F_{synchro} \times W_{synchro}} \to \max$. But the definition of the weight coefficients could be a very difficult task for the decision-maker, so the analytic hierarchy process is considered rational.

2 REVIEW OF THE LITERATURE

The research's authors in the set of their previous works emphasize on potential of using the collected history of users' SQL-queries [12–15]. The queries should previously be parsed to extract the sets of entities (table), attributes (columns), and tuples (rows). Then, this data can be used to determine the level of necessity for this data to be presented on the node of the distributed database. And finally after the optimal level of data representation on the distributed database node will be found the optimal structure of its database can be built.

In work [12] authors firstly presented technology of users' queries parsing. Then, in [13] the idea of using multidimensional database for analysis users' queries by slices of workstation type, application, user and his/her position was considered. Finally, in [14] authors gave the full scaled mathematical model for formalizing database and query models, and criteria of database structure's optimality. It was presented the "data representation marker" term, which determined the level of their need at the node of the distributed corporate information system (DCIS). The multicreterial decision support system that was also presented in [15], searches the optimal value of data representation marker based on the following criteria: independence from the central node of the database, the size of the local database and the level of needed data synchronization.

The current research continues the given sequence and tries to increase the efficiency of decision support system, given in [14], by introduction to the analytic hierarchy process algorithm (that was used previously) elements of fuzzy logic. The approach's main idea is in presenting the global priorities vector in the form of series of fuzzy sets of one variable. The following representation this set in

© Dvoretskyi M. L., Savchuk T. O., Fisun M. T., Dvoretska S. V., 2022
DOI 10.15588/1607-3274-2022-2-10

the form of exact value (defasification) will increase the accuracy of the obtained solution without the necessity of dividing the range of solution's valid values to more number of intervals. Also, in order to reduce the necessity of re-conducting the analysis after some time passes, the solution of classification task is considered that will help to determine the "data representation marker" for the new data in the database.

3 MATERIALS AND METHODS

While working on [14] authors proposed the model of users' SQL-queries according to which subsets of data can be extracted from the main database to be presented on the remote node of the DCIS. It was suggested following legends: R – database relation (table); $R[P]$ – table projection; $R[S]$ – table selection; *tup* –a tuple of the relation R. Within the SQL-query for data selecting, a number of relations can be involved, all of which are the result of sequential execution of selection and projection operations to the base relation (table). $R'' = R'[P]$, where $R' = R[S]$, i.e. $R'' = \{tup[P] \mid tup[P] \in R[P]_{data} \wedge F(tup, S) = true\}$ [14]. When working with the sequence of the database's queries, the subset R''_{union} of the relation R was defined as the union of subsets R' of each SQL-query that came from the DCIS remote node. $R''_{union} = \bigcup_{i=1}^{n} R''_i$, or $R''_{union} = \{tup[P_{union}] \mid tup[P_{union}] \in R[P_{union}]_{data} \wedge F(tup, S_{union}) = true\}$, where $tup[P_{union}] = \bigcup_{i=1}^{n} tup[P_i]$, and $S_{union} = \bigcup_{i=1}^{n} S_i$.

Some part of data can be presented locally to avoid the necessity of remote queries and another part is placed at the remote node to reduce the amount of data that should be replicated between the nodes. The subset of the base relation R that describes the relation of the remote node was represented as follows: $R^{remote}_{schema} = \{A \mid A \in R_{shema}, R_{primary} \subset R^{remote}_{schema}, A \in R_{primary} \vee F_a(Node, A) = true\}$. And the set of table's tuples was determined by the formula $R^{remote}_{data} = \{ tup \mid tup \in R_{data}, tup_{primary} \in R^{remote-dep}_{data} \vee F_{tup}(Node, tup) = true\}$. The need for the data on the remote node is determined mostly by the evaluation function $F_{tup}(Node, tup)$.

The model of SQL-queries supports further classification according to belonging to the workplace, location, user role and other potential dimensions $Q = <$*Workplace*, *User*, *Application*, R''_{shema}, $Q^{inner}_{set}>$. In this formula $R''_{set} = \{ R'' \mid \{tup[P] \mid tup[P] \in R[P]_{data} \wedge F(tup, S) = true \}$ – is the resulting relations' set; Q^{inner}_{set} – the nested queries' set. Based on this query model the multidimensional database [13] was created. It has the following set of dimensions: <*DateTime*, *WorkplaceType*, *Location*, *UserRole*, *Application*, *R*, *A*, *tup*>. Then, the term of data representation marker was proposed that reflects the level of data representation necessity at the DCIS node. For each dimension's element value of the marker is determined from the following set: {"*obligatorily*", "*necessary*", "*neutral*", "*not required*", "*forbidden*"}. After been converted to a numeric values ("*obligatorily*" – "2", "*necessary*" – "1", "*neutral*" – "0", "*not required*" – "–1", "*forbidden*" – "–2"), it was defined the aggregation function for the marker:

$$Aggr_{i=1}^{n} Mrk_i = \begin{cases} 2, if\ \exists Mrk_i = 2 \\ -2, if\ \exists Mrk_i = -2 \wedge \nexists Mrk_i = 2. \\ \sum_{i=1}^{n}(Mrk_i \times \frac{vol_i}{\sum_{i=1}^{n} vol_i}) \end{cases} \tag{1}$$

The decision about the necessity of the data slice <*R*, *A*, *tup*> on the remote node is made according to the following condition:

$$\begin{aligned} &Repr(Node, R, A, tup) = \\ &= Aggr_{i=1}^{n}(R, A, tup) Mrk_i > koef_{repr}^{node}, \end{aligned} \tag{2}$$

where $koef_{repr}^{node}$ – the data representation's threshold coefficient, defined on the range [–1, 1].

The optimality of the $koef_{repr}^{node}$ is defined by three criteria. These are independence from the central database node, local database size, and the need for data synchronization. The value of each criterion is defined by (3), (4), and (5) accordingly

$$F_{availab} = \frac{\sum_{i=1}^{n} F_{availab}(Q_n)}{n}, \tag{3}$$

$$F_{size} = \sum_{i=1}^{n} \frac{size(R''_i)}{size(R_i^{DBMS})}, \tag{4}$$

$$F_{synchro} = \frac{p_{node}^{\mathrm{mod}\, if}}{p_{node}}. \tag{5}$$

A deeper explanation of given formulas can be found in [15].

The obtained multicriteria problem was solved using the analytic hierarchy process. When compiling the hierarchy, the following relationship between the levels' elements was used: goal – stakeholders – criteria – alternatives. The value of the data representation marker (alternative) is a real number in the interval [–1, 1]. It leads to the potentially large number of alternatives at the 4th level of the hierarchy and therefore the matrices of pairwise comparisons by criteria can become very big. This complicates the estimation process for the decision-makers. It is proposed to simplify the task by reducing the number of alternatives to 5: "*low*" (*L*) – "–1", "l*ower them medium*" (*LM*) – "–0.5", "*medium*" (*M*) – "0", "*higher then medium*" (*HM*) – "0.5", and "*high*" (*H*) – "1". The level of "*decision-makers*" is represented by the elements "*Owner*", "*Database Administrator*", "*Database*

© Dvoretskyi M. L., Savchuk T. O., Fisun M. T., Dvoretska S. V., 2022
DOI 10.15588/1607-3274-2022-2-10

Developer", and "*CIS Operator*". The obtained hierarchical model is shown in Fig. 1.

When splitting the set of alternatives of the data representation marker's value (determined on the set of real numbers in the interval [–1; 1]) into a larger number of intervals, it is possible to increase the accuracy of the obtained solution.

Based on the difficulties of the larger number of intervals implementation and the need to improve the accuracy of the method, it was suggested to use the elements of the fuzzy logic apparatus [16–19]. This makes it possible to obtain the same result's accuracy as in the case of the larger number of alternatives, using a fewer number of intervals of the data representation marker.

In the theory of sets, an element either belongs to the set or not. A fuzzy set is defined using a membership function that corresponds to the concept of a characteristic function in classical logic. The membership function can take any form, but the piecewise linear form is used most often to represent it. Piecewise linear membership functions are traditionally used for several reasons: they are characterized by simplicity; they contain points that allow defining areas where the concept is true and where it is false, which simplifies the system's description [17].

In Fig. 2 the selected intervals of the data representation marker level values ("*low*", "*lower than medium*", "*medium*", "*higher than medium*", "*high*") are presented in the form of a series of fuzzy sets of one variable with piecewise linear membership functions.

The classical process of fuzzy inference consists of the following stages: fuzzification (presenting the exact numeric value in a fuzzy form), fuzzy inference itself, usually based on a set of rules, and defuzzification (numerical expression of a fuzzy result) [20].

In our case, the solution of the multicriteria analysis problem was performed previously using the analytic hierarchy process method, and the following vector of alternatives' global priorities was obtained (10).

Further, the found vector of global priorities is represented as a vector of fuzzy numbers for the data representation marker. That is, to obtain the numerical value of the data presentation marker's optimal level, should be accomplished mapping of the results with the next defuzzification phase. Defuzzification is the process of trans-

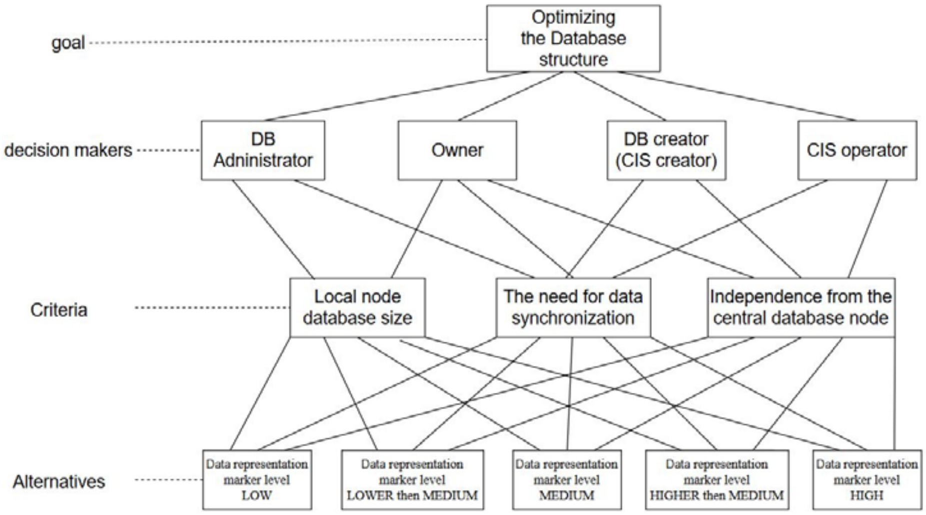
goal
Optimizing the Database structure
decision makers
DB Administrator
Owner
DB creator (CIS creator)
CIS operator
Criteria
Local node database size
The need for data synchronization
Independence from the central database node
Alternatives
Data representation marker level LOW
Data representation marker level LOWER then MEDIUM
Data representation marker level MEDIUM
Data representation marker level HIGHER then MEDIUM
Data representation marker level HIGH

Figure 1 – Hierarchical model of the distributed CIS node structure optimization problem

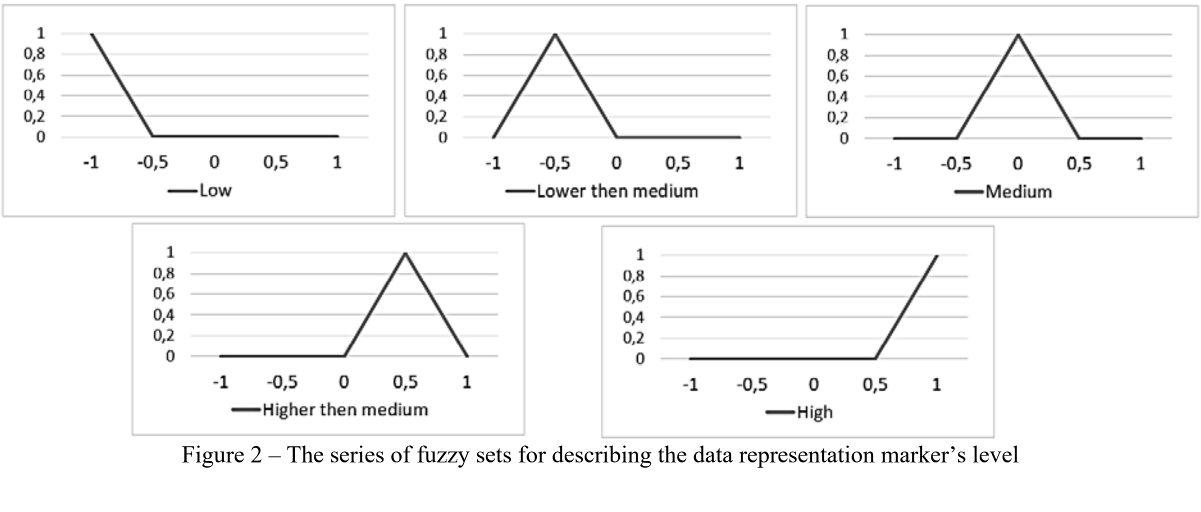

Figure 2 – The series of fuzzy sets for describing the data representation marker's level

DOI 10.15588/1607-3274-2022-2-10

forming a fuzzy number to its exact numeric interpretation. In the theory of fuzzy sets, defuzzification is similar to finding characteristics of the random variables' position (mean, mode, median) in the theory of probability. Among the defuzzification methods, there is the choice of an exact number with the maximum value of the membership function. Alternatively, it can be methods based on the idea of finding the point of concentration (center of mass) of the area located between the membership function's graph and the abscissa axis. This is done in order to "average" the possible values taken by a fuzzy number, taking into account the belonging function value.

Among the most common methods is the center of mass method $a_r = \int_{\min}^{\max} u\mu_{\varepsilon}(u)du \Big/ \int_{\min}^{\max} \mu_{\varepsilon}(u)du$ and the median method $a_r : \int_{\min}^{\max} u\mu_{\varepsilon}(u)du = \int_{\min}^{\max} \mu_{\varepsilon}(u)du$.

In the process of using the described technology, there is a problem related to adding new tuples to the database tables. The level of data presentation's need is determined in accordance with the accumulated statistics of SQL queries and the range of primary keys that meet the selection condition. The range of the primary key of the recently added to the database tuples was not available during the analysis. Therefore, they were not marked according to the statistics of queries in the database. For these tuples, there is no value for the level of the need to represent data in the DCIS node. The simplest solution to the problem is to replicate all new data to the remote DCIS node (since there is no data about the degree of their usefulness for the node), that is, setting for them the value of the need for presentation at the level of "obligatory". This approach is not fully justified, because, firstly, it does not take into account the influence of data (added and specially modified) rows on other rows due to their intersection within the same query (software or host type). In addition, when the amount of data reaches the critical point, based, for example, on load during selecting or data synchronization operations, there comes a time when it is necessary to re-initialize the multidimensional database of SQL-queries and update the data of using the attributes and tuples of the database tables by the DCIS node [13, 21].

Accomplishing the complete analysis of using the attributes and tuples of database tables is a resource-expensive operation and cannot be performed frequently. Therefore, the given above approach is unacceptable for large and frequently changing databases. At the same time, secondary and subsequent monitoring of the database users' activity is complicated by the fact that the database of the remote node is already in use, and requests to it must also be taken into account.

In most cases, the main influence on the level of representation in the analysis of user queries to the database makes the combination of the attributes' values of the relation's tuple. Taking into account this fact, it is proposed to present the problem with determining the representation marker's level of the new data in the form of a classification problem. The input parameters are the name of the table (relationship) and the list of tuple attributes' values and the result will be a decision of presenting the tuple for the DCIS node.

Among the many algorithms for solving the classification problem by means of machine learning, the most popular approaches are identified, including linear and logical regression, discriminant analysis, decision trees, the Naive Bayes algorithm, k-nearest neighbors, and the use of various types of neural networks [22]. Linear and logical regression are some of the most well-known methods, but according to the non-numerical characteristics of most of the input variables (table attributes), they are not optimal in this case. For the same reasons, and also because of the complexity of determining the distance, the k-nearest neighbors aren't considered either.

The use of neural networks is now the most popular direction in solving the classification problem [23–24]. However, not all tables contain a sufficient number of rows to carry out a high-quality training stage (insufficient amount of data for training and testing the results). In addition, each table (relation) has a different number of attributes, and each of them is defined on its own domain. Based on this, the problem of classifying new data for each relation must be solved using a separately trained neural network. According to the given above, the use of neural networks isn't also the appropriate approach.

To solve the problem, the Bayes naive algorithm [25] was used, which is simple but effective and allows to quickly determine the probability of an object belonging to a particular class. The algorithm is known to be based on the assumption that each input variable is independent, which is often not true. But production versions of the relational databases in most cases are in the 3rd normal form, which indicates the absence of transitive dependencies inside the relation [26]. This fact allows asserting that the main part of the input variables corresponds to the basic algorithm's assumption.

The algorithm is based on Bayes' theorem, which allows calculating the probability of an object belonging to a particular class [25]. For the current task, the probability that the tuple *X* is needed to be presented at a remote node according to the value of the *i*-th attribute x_i, was found using the following formula:

$$P_x(needed \mid x_i) = \frac{P(x_i \mid needed) \times P(needed)}{P(x_i)}, \qquad (6)$$

where $P(needed)$ – the total probability of the relation that the tuple has to be represented on a remote node; $P(x_i)$ – the probability of the x_i value of the *i-th* attribute; $P(x_i \mid needed)$ – the probability of the x_i value of the *i-th* attribute on the subset of the relation's tuples of the remote node.

Also, the same way calculate the probability that the tuple *X* is not needed to be presented at a remote node:

DOI 10.15588/1607-3274-2022-2-10

$$P_x(not\,needed \mid x_i) = \frac{P(x_i \mid not\,needed) \times P(not\,need}{P(x_i)} \qquad (7)$$

After obtaining probabilities $P_x(not\,needed \mid x_i)$ for all tuple's attributes based on (6) and (7), the calculation of the probability that the whole tuple will be needed is performed:

$$P_x(needed \mid X) = \\ = \frac{\prod_{i=1}^{n} P_x(needed \mid x_i)}{\prod_{i=1}^{n} P_x(needed \mid x_i) + \prod_{i=1}^{n} P_x(not\,needed \mid x_i)}. \qquad (8)$$

If the value, obtained according to (6) is greater than the optimal value of the data representation marker, then the tuple is accepted as the one that has to be presented on the remote node of the DCIS:

$$F_X^{needed} = \begin{cases} true, \dfrac{koef_{distrib}^{node} + 1}{2} > P(needed \mid X) \\ false, \dfrac{koef_{distrib}^{node} + 1}{2} \le P(needed \mid X) \end{cases}. \qquad (9)$$

So, using (8) and (9) it is possible to make a decision about presenting a new table's tuples on the remote node of DCIS without the necessity of re-processing the SQL-queries statistic. It simply can be done based on the data for which the decision about its presenting on the remote node was already made.

4 EXPERIMENTS

Using SQL-queries statistics and models (1–5) the global alternatives' vector was obtained:

$$W^{global} = \begin{bmatrix} 0.000 \\ 0.273 \\ 0.334 \\ 0.393 \\ 0.000 \end{bmatrix}. \qquad (10)$$

While working with pairwise comparison matrices the ranges of available values of optimality criteria were also taken into account. The full description of obtaining the given result can be found in [15].

The analytic hierarchy process method that was given in [14], was used to obtain the optimal solution from 5 and 21 alternatives (intervals of the data representation marker's values). For the case of 21 alternatives the task has to be simplified, so no restrictions on the optimality criteria were introduced and a three-level hierarchical model without the "*stakeholder*" level was used. As a matrix of pairwise comparisons of optimality criteria, the matrix that was filled by the "*Owner*" person was taken [15]. Also, the initial state of the alternatives' advantages matrices according to the optimality criteria, which were obtained in accordance with the mathematical models (3), (4), and (5) is accepted as final.

Taking into account the introduced simplifications of the model, the calculation of the global priorities' vector for five alternatives ("*low*" (*L*) – "–1", "*lower than medium*" (*LM*) – "–0.5", "*medium*" (*M*) – "0", "*higher than medium*" (*HM*) – "0.5", and "*high*" (*H*) – "1") gives the following result with the optimal alternative "*high*", which corresponds to the level of the data representation marker equal to "1":

$$W_5^{global} = \begin{bmatrix} 0.09 \\ 0.09 \\ 0.15 \\ 0.30 \\ 0.37 \end{bmatrix}. \qquad (11)$$

After dividing the range of marker's values into *21* intervals with a step of 0.1, the 21 alternatives were obtained. The set of the data representation marker's values is following: A = {–1.0; –0.9; –0.8; –0.7; –0.6; –0.5; –0.4; –0.3; –0.2; –0.1; 0; 0.1; 0.2; 0.3; 0.4; 0.5; 0.6; 0.7; 0.8; 0.9; 1.0}. The obtained alternatives are designated as *A1*, *A2*, *A3*,…, *A21.* After calculating the values of the optimality criteria, according to (3), (4), and (5), normalizing the obtained values, filling in the matrices of pairwise comparisons, and calculating the vector of global priorities in accordance with W_i=k_{size}×$W_{size,i}$+ $k_{availab}$×$W_{availab,i}$+ $k_{synchro}$×$W_{synchro,i}$ and Barkly's formula [16], the corresponding value of the alternatives' global priorities vector, given in Table 1, was obtained.

In this case, the decrease of the interval's step and, accordingly, an increase in the accuracy reveals the alternative A18 as the best. It corresponds to the data representation marker's value = 0.7. However, the use of this approach leads to the need for further filling of the pairwise comparisons' matrix of size 21x21 by the decision-maker, i.e. requires to perform 210 operations of pairwise comparisons of alternatives for each criterion of optimality by each decision-maker. In addition, it is very difficult for a person to prioritize alternatives with insignificant changes in their qualitative characteristics.

In Fig. 3 is shown the graphical representation of the fuzzy numbers' vector. This vector was obtained for the simplified version of the problem and given in the form of the global priorities matrix (11) for the data representation marker.

Table 1 – Alternatives' global priorities vector for 21 intervals of the data representation marker's values

Alternative	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11
Data representation marker	–1.0	–0.9	–0.8	–0.7	–0.6	–0.5	–0.4	–0.3	–0.2	–0.1	0.0
Priorities vector's element	0.008	0.012	0.009	0.013	0.01	0.012	0.009	0.01	0.013	0.021	0.034
Alternative	A12	A13	A14	A15	A16	A17	**A18**	A19	A20	A21	–
Data representation marker	0.1	0.2	0.3	0.4	0.5	0.6	**0.7**	0.8	0.9	1.0	–
Priorities vector's element	0.05	0.054	0.063	0.076	0.082	0.101	**0.117**	0.111	0.103	0.093	–

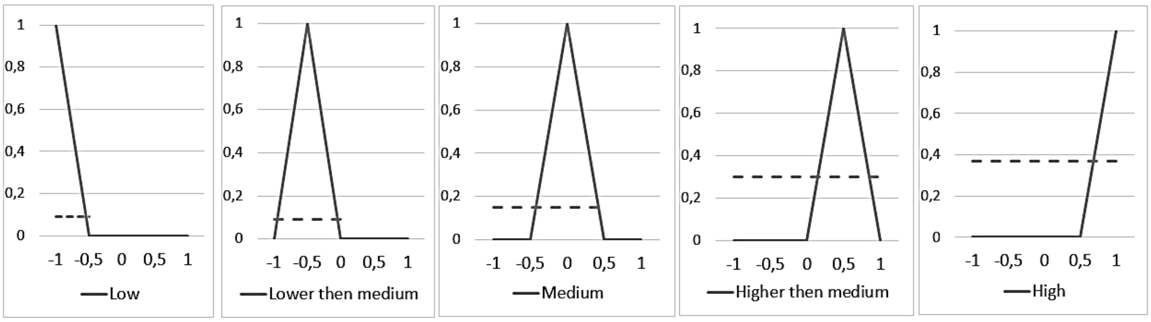

Figure 3 – Fuzzy numbers' vector of alternatives' global priorities

After performing the aggregation and defuzzification operations of the obtained results, the data representation marker's value at the DCIS node at the level of 0.7 was received (Fig. 4).

The obtained optimal solution is the same as for the case of 21 intervals of the data representation marker's values (table 1). According to the received results, it can be concluded that it is possible to use elements of fuzzy inference to increase the solution accuracy with a small number of intervals of the data representation marker's value. The "elements of fuzzy inference" mean the representation of the alternatives global priorities' vector in the form of the fuzzy numbers' vector with subsequent aggregation and defuzzification of the result.

5 RESULTS

After making sure on test solution that method works the research applies the modified method to the alternatives global priorities' vector that was got after solving the task with all available value's area restrictions and filling pairwise comparison matrix by the decision-makers. In Fig. 5 shows the result obtained previously according to the data of the alternatives global priorities' vector (10) and additionally processed using fuzzy logic elements. According to it, the optimal value of the data representation marker of the DCIS node for the work's implementation subject area of the results is obtained at the level of 0.2.

Consequently, the use of the analytic hierarchy process method based on a limited set of alternatives makes it possible to construct advantages matrices and perform the necessary calculations to obtain the values of the alternatives' global advantages. At that, in some cases with the involvement of a decision-maker, and in some cases using mathematical models of optimality criteria presented previously. Using elements of fuzzy logic, specifically the defuzzification of the fuzzy numbers' vector for the data representation marker (vector of global priorities), makes it possible to increase the accuracy of the result while determining the optimal value of the data representation marker's level.

6 DISCUSSION

To select the best alternative of the data representation marker's level the analytic hierarchy process was used. To construct matrices of the alternatives' advantages, their set was reduced to five alternatives. The matrix of the optimality criteria's advantages was obtained classically with the involvement of a decision-maker and subsequent concordance index estimation. The matrices of the alternatives' advantages were filled in automatically, without the participation of a decision-maker, based on the models of optimality criteria. Also, restrictions for the range of valid values of the optimality criteria were introduced. Were presented maximum and minimum values for each criterion, which leads to reducing the number of alternatives at the last level of the hierarchy model.

After calculating and obtaining the alternatives global priorities' vector in order to improve the accuracy of the obtained result, the apparatus of fuzzy sets was used. The obtained vector of global priorities was presented as a vector of fuzzy digits for the data representation marker with subsequent transformation to the exact value. This approach made it possible to maintain the accuracy of the obtained result while decreasing the number of solution alternatives.

For new tuples added to the database's tables after all calculations had been performed, the classification problem was formalized. This task assumes determining the belonging of the new tuple to, one of two classes – "*needed*" at a remote node and "*not needed*". After comparing the most popular approaches to solving the classification problem, the Naïve Bayes algorithm was chosen,

DOI 10.15588/1607-3274-2022-2-10

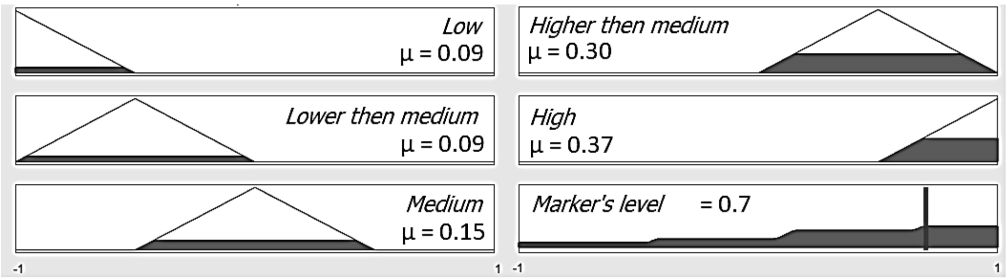

Figure 4 – Stage of aggregation and dephasification

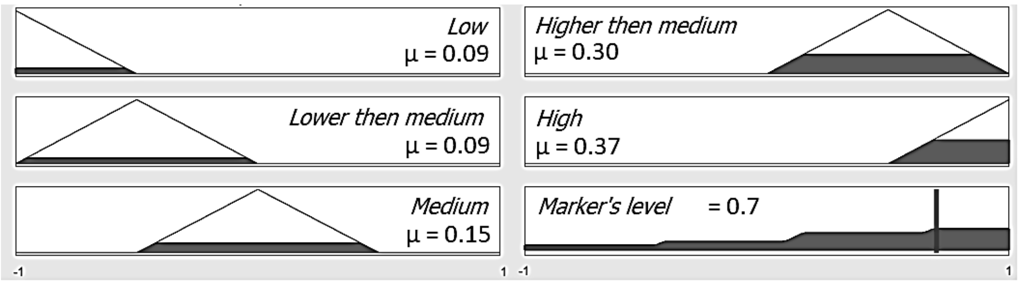

Figure 5 – Aggregation and dephasification of the obtained results

since, in the absence of transitive dependencies (the requirement for the third normal form of a relational database), all the attributes of the relation are independent. After obtaining the probability of a tuple belonging to the class “*needed*” and performing the normalization of the value, it is taken as the level of the representation marker. Accordingly, the data is loaded onto the node if this value is greater than the optimal level of the representation marker for the DCIS node.

While working on the research, the concept of a data representation marker on the DCIS node for the elements of the SQL query model was introduced. An aggregation function has been developed that allows determining the level of need for attributes and tuples in the database’s relation for the DCIS node based on the statistics of SQL queries. A model of the dependence of the database structure’s optimality criteria of the DCIS node on the value of the data representation marker is built. It, in contrast to existing approaches, allows determining the optimal value of the marker based on the statistics of SQL queries. Received further development method of analytic hierarchy process. The initialization of the alternatives’ pairwise comparisons matrix can be performed automatically according to the obtained mathematical models. Representation of the obtained result in the form of the vector of fuzzy numbers with the reduction to the exact value allows increasing the accuracy of the obtained results.

CONCLUSIONS

While resolving the problem of optimizing the structure of the database node in the corporate information systems solved the urgent problem of determining the optimal value of the data representation marker on the DCIS node. Due to the use of fuzzy logic elements in solving the problem by the analytic hierarchy process, the accuracy of the obtained result is increased without the need to increase the number of solution alternatives.

The scientific novelty of obtained results is that the concept of data representation marker of DCIS node for dimension’s elements of SQL-query model was firstly introduced with following developing of the aggregation function, presenting the model of dependence of DCIS node’s database structure optimality criteria on the data representation marker’s value.

The analytic hierarchy process has received further development. It happened due to the automatic initialization of the alternatives’ pairwise comparison matrix according to the received mathematical models. The vector of alternatives’ global priorities was given in the form of the vector of fuzzy numbers with the future reduction to the numeric value, which increased the accuracy of the obtained value.

The practical significance of obtained results is that the software supporting the decision-making in determining the representation of data on the DCIS node was developed. It allows optimizing the structure of the database node. Developed models and information technology were implemented at “Elite Building TOV” to identify the dependence of the optimal database structure’s criteria and the level of data representation marker. The effect of the implementation is to increase the speed of requests’ execution by 14%

Prospects for further research are to study the results of experiments of implementing the suggested modification in the analytic hierarchy process on different subject areas. It is also will be interesting to try the model on the DCIS with no central data node.

ACKNOWLEDGEMENTS

This study was funded and supported by Petro Mohyla Black Sea National University (PM BSNU) in Mykolaiv (Ukraine), and also financed in part of the PM BSNU Science-Research Work by the Ministry of Education and Science of Ukraine “Development of the modules for automatization of wireless devices for recovery of postinfarction, post-stroke patients in individual conditions of

DOI 10.15588/1607-3274-2022-2-10

remote individual rehabilitation" (State Reg. No. 0121U109898).

REFERENCES

1. Hamouda S., Zainol Z. Document-Oriented Data Schema for Relational Database Migration to NoSQL, *2017 International Conference on Big Data Innovations and Applications (Innovate-Data), Czech Republic*, 2017, pp. 43–50. DOI: 10.1109/Innovate-Data.2017.13
2. Hows D., Membrey P., Plugge E., Hawkins T. The Definitive Guide to MongoDB. Berkeley, CA, Apress, 2015, 343 p. DOI: 10.1007/978-1-4842-1182-3
3. Thakur N., Gupta N. Relational and Non Relational Databases: A Review, *Journal of University of Shanghai for Science and Technology*, 2021, Vol. 23, No. 8, pp. 117–121. DOI: 10.51201/jusst/21/08341
4. Kundu P., Arora T. Research of Persistence Solution Based on ORM and Hibernate Technology, *International Journal of Advanced Research in Computer Science and Software Engineering*, 2017, Vol. 7, No. 4, pp. 359–362. DOI: 10.23956/ijarcsse/v7i3/0154
5. Becker J., Uhr W., Vering O. Systems for the Support of the Company Management, Retail Information Systems Based on SAP Products. Berlin, Springer Berlin Heidelberg, 2013, Chapter 5, pp. 121–150. DOI: 10.1007/978-3-662-09760-1_5
6. Petrova E. Overview of modern automation information systems activities of trade enterprises, *Journal of management studies*, 2018, Vol. 4, No. 9, pp. 76–85. DOI: 10.12737/article_5d68d5afb331c1.42407139
7. Christudas B. Practical Microservices Architectural Patterns. Berkeley, CA, Apress, 2019, 812 p. DOI: 10.1007/978-1-4842-4501-9
8. Peterson C., Wilson A., Pirkelbauer P. et al. Optimized Transactional Data Structure Approach to Concurrency Control for In-Memory Databases, *2020 IEEE 32nd International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD),* 2020, pp. 107–115. DOI: 10.1109/SBAC-PAD49847.2020.00025
9. Perez L. L., Jermaine C. M. History-aware query optimization with materialized intermediate views, *2014 IEEE 30th International Conference on Data Engineering,* 2014, pp. 520–531, DOI: 10.1109/ICDE.2014.6816678
10. Tsegelyk G. G., Krasniuk R. P. The optimization of databases replication in distributed information systems, *Information Extraction and Processing*, 2017, Vol. 45, No. 121, pp. 104–112. DOI:https://doi.org/10.15407/vidbir2017.45
11. Korniyenko B. Y., Galata L. P. Optimization of the Information System of the Corporate Network, *MCM-TECH, Kamianets-Podilskyi National Ivan Ohiienko University*, 2019, pp. 56–62. DOI: 10.32626/2308-5916.2019-19.56-62
12. Fisun M., Dvoretskyi M., Shved A. et al. Query parsing in order to optimize distributed DB structure, *9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems, Technology and Applications (IDAACS), Bucharest, 2017, proceeding.* Bucharest, IEEE, 2017, pp. 172–178. DOI: 10.1109/IDAACS.2017.8095071
13. Dvoretskyi M., Dvoretska S., Nezdoliy Y. et al. Data Utility Assessment while Optimizing the Structure and Minimizing the Volume of a Distributed Database Node, *1st International Workshop on Information-Communication Technologies & Embedded Systems (ICTES), 2516, 2019, proceeding*, CEUR Workshop, 2019, pp. 128–137. Available online: http://ceur-ws.org/Vol-2516/paper10.pdf
14. Dvoretskyi M., Dvoretska S., Horban H. et al. Optimization of the database structure of a distributed corporate information system node using the analytic hierarchy process, *T&I Workshops, 2845, 2020, proceeding, CEUR Workshop*, 2020, pp. 193–203. Available online: http://ceur-ws.org/Vol-2845/Paper_19.pdf
15. Fisun M., Dvoretskiy M., Dvoretska S. Building a model to optimize the database structure of the node in corporate information systems, *Information technology and computer engineering: International Scientific and Technical Journal of Vinnytsia National Technical University*, 2020, Vol. 48, No. 2, pp. 52–60. DOI: 10.31649/1999-9941-2020-48-2-52-60
16. Zadeh L. A., Klir G. J., Yuan B. Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems, World scientific, 1996, 840 p. DOI: 10.1142/2895
17. Alang-Rashid N. K., Heger A. S. A general purpose fuzzy logic code, *IEEE International Conference on Fuzzy Systems, 1992, proceeding*, IEEE, 1992, pp. 733–742. DOI: 10.1109/FUZZY.1992.2587
18. Gozhyj A., Kalinina I., Gozhyj V. Fuzzy cognitive analy-sis and modeling of water quality, *9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 2017, proceeding*. IEEE, 2017, pp. 289–293. DOI: 10.1109/IDAACS.2017.8095092
19. Yager R. R. On inference structures for fuzzy systems modeling, *IEEE 3rd International Fuzzy Systems Conference. –* 1994, Vol. 2, pp. 1252–1256. DOI: 10.1109/FUZZY.1994.343642
20. Nakamura K., Sakashita N., Nitta Y. et al. Fuzzy inference and fuzzy inference processor, *IEEE Micro,* 1993, Vol. 13, No. 5, pp. 37–48. DOI: 10.1109/40.238000
21. Dvoretskiy M., Dvoretska S., Davidenko E. Information technology for determining useful data while optimizing the structure and minimizing the volume of the distributed database node, *Bulletin of Cherkasy State Technological University*, 2019, No. 4, pp. 26–35. DOI: 10.24025/2306-4412.4.2019.184808
22. [Hegde R., Anusha G. V., Madival S. et al. Review on Data Mining and Machine Learning Methods for Student Scholarship Prediction, *2021 5th International Conference on Computing Methodologies and Communication (ICCMC), 2021, proceeding,* IEEE, 2021, pp. 923–927. DOI: 10.1109/ICCMC51019.2021.9418376
23. Zaki M. J., Meira W. J. Neural Networks, *Data Mining and Machine Learning. Cambridge University Press,* 2020, pp. 637–671. DOI: 10.1017/9781108564175.031
24. Graupe D. Deep Learning Neural Networks. World scientific, 2016, 280 p. DOI: 10.1142/10190
25. Janssen J., Laatz W. Naive Bayes, *Statistische Datenanalyse mit SPSS.* Springer Berlin Heidelberg, 2017, pp. 557–569. DOI: 10.1007/978-3-662-53477-9_25
26. Krishna S. Introduction to Database and Knowledge-Base Systems, World scientific, 1992, 344 p. DOI: 10.1142/1374

Received 20.02.2022.
Accepted 23.04.2022.

УДК 004.658.3

ВИКОРИСТАННЯ МЕТОДУ АНАЛІЗУ ІЄРАРХІЙ ТА ЕЛЕМЕНТІВ НЕЧІТКОЇ ЛОГІКИ ДЛЯ ОПТИМІЗАЦІЇ СТРУКТУРИ БАЗИ ДАНИХ

Дворецький М. Л. – канд. техн. наук, доцент б.в.з. кафедри інженерії програмного забезпечення, Чорноморський національний університет імені Петра Могили, Миколаїв, Україна.

DOI 10.15588/1607-3274-2022-2-10

Савчук Т. О. – канд. техн. наук, професор, професор кафедри комп'ютерних наук, Вінницький національний технічний університет, Вінниця, Україна.

Фісун М. Т. – д-р техн. наук, професор, професор кафедри інженерії програмного забезпечення, Чорноморський національний університет імені Петра Могили, Миколаїв, Україна.

Дворецька С. В. – старший викладач кафедри інженерії програмного забезпечення, Чорноморський національний університет імені Петра Могили, Миколаїв, Україна.

АНОТАЦІЯ

Актуальність. Інформаційні системи дуже поширені і використовують бази даних для зберігання інформації. Для використання доступні різні моделі даних, але реляційна модель залишається популярною. Останнє десятиліття демонструє тенденцію використання розподілених баз даних під час роботи з реляційною моделлю, і цей підхід вимагає спеціально розробленого модуля для синхронізації даних усіх окремих частин БД. Оптимальна структура всіх розподілених вузлів могла б зменшити необхідність синхронізації, а швидкість доступу до даних та її актуальність залишалися б стабільними.

Метод. Автори дослідження в серії своїх попередніх робіт акцентують увагу на можливості використання зібраної історії SQL-запитів користувачів. Спочатку представлена технологія розбору запитів користувачів. Потім була розглянута ідея використання багатовимірної бази даних для аналізу запитів користувачів за зрізами типу робочої станції, програми, користувача та його посади. Також автори надали математичну модель формалізації моделі бази даних і запитів, а також критерії оптимальності структури бази даних. Дослідження продовжує наведену послідовність і намагається підвищити ефективність системи підтримки прийняття рішень шляхом введення в метод аналізу ієрархій елементів нечіткої логіки. Основна ідея підходу полягає в представленні вектору глобального пріоритету у вигляді серії нечітких множин однієї змінної з подальшим перетворенням до точного значення. Для нових кортежів, доданих до таблиць бази даних після виконання всіх обчислень, була сформульована задача класифікації.

Результати. Після розрахунку та отримання вектору глобального пріоритету альтернатив з метою підвищення точності отриманого результату було використано апарат нечітких множин. Отриманий вектор глобальних пріоритетів був представлений у вигляді вектору нечітких множин для маркера представлення даних з подальшим перетворенням до точного значення. Такий підхід дозволив зберегти точність отриманого результату при зменшенні кількості альтернатив рішення.

Висновки. Під час роботи над дослідженням було введено поняття маркера представлення даних на вузлі РКІС для елементів моделі запиту SQL. Розроблено функцію агрегації, яка на основі статистики SQL-запитів дозволяє визначити рівень необхідності атрибутів і кортежів відношення бази даних на вузлі РКІС. Побудовано модель залежності критеріїв оптимальності структури бази даних вузла РКІС від значення маркера представленості даних. Отримав подальший розвиток метод аналізу ієрархій. Ініціалізація матриці попарних порівнянь альтернатив може виконуватися автоматично відповідно до отриманих математичних моделей. Представлення отриманого результату у вигляді вектору нечітких чисел із приведенням до точного значення дозволяє підвищити точність отриманих результатів.

КЛЮЧОВІ СЛОВА: корпоративна інформаційна система, система управління базами даних, розподілена база даних, SQL-запит, реплікація даних, багатокритеріальна задача, метод аналізу ієрархій, нечітка логіка, задача класифікації, наївний алгоритм Байєса.

УДК 004.658.3

ИСПОЛЬЗОВАНИЕ МЕТОДА АНАЛИЗА ИЕРАРХИЙ И ЭЛЕМЕНТОВ НЕЧЕТКОЙ ЛОГИКИ ДЛЯ ОПТИМИЗАЦИИ СТРУКТУРЫ БАЗЫ ДАННЫХ

Дворецкий М. Л. – канд. техн. наук, и.о. доцента кафедры инженерии программного обеспечения, Черноморский национальный университет имени Петра Могилы, Николаев, Украина.

Савчук Т. А. – канд. техн. наук, профессор, профессор кафедры компьютерных наук, Винницкий национальный технический университет, Винница, Украина.

Фисун Н. Т. – д-р техн. наук, профессор, профессор кафедры инженерии программного обеспечения, Черноморский национальный университет имени Петра Могилы, Николаев, Украина.

Дворецкая С. В. – старший преподаватель кафедры инженерии программного обеспечения, Черноморский национальный университет имени Петра Могилы, Николаев, Украина.

АННОТАЦИЯ

Актуальность. Информационные системы широко распространены и используют базы данных для хранения информации. Для использования доступны разные модели данных, но реляционная модель остается популярной. Последнее десятилетие демонстрирует тенденцию использования распределенных баз данных при работе с реляционной моделью, и этот подход требует специально разработанного модуля для синхронизации данных всех отдельных частей БД. Оптимальная структура всех распределенных узлов могла бы снизить необходимость синхронизации, а скорость доступа к данным и ее актуальность оставались бы стабильными.

Метод. Авторы исследования в серии своих предыдущих работ акцентируют внимание на возможности использования собранной истории SQL запросов пользователей. Первоначально представлена технология разбора запросов пользователей. Затем была рассмотрена идея использования многомерной базы данных для анализа запросов пользователей по срезам типа рабочей станции, программы, пользователя и его должности. Также авторы предоставили математическую модель формализации модели базы данных и запросов, а также критерии оптимальности структуры базы данных. Исследование продолжает приведенную последовательность и пытается повысить эффективность системы поддержки принятия решений путем введения в метод анализа иерархий элементов нечеткой логики. Основная идея подхода заключается в представлении вектора глобального приоритета в виде серии нечетких множеств одной переменной с последующим превращением в точное значение. Для новых кортежей, добавленных в таблицы базы данных после выполнения всех вычислений, была сформулирована задача классификации.

Результаты. После расчета и получения вектора глобального приоритета альтернатив с целью повышения точности полученного результата был использован аппарат нечетких множеств. Полученный вектор глобальных приоритетов был представлен в виде вектора нечетких множеств для представления данных маркера с последующим превращением в точное значение. Такой подход позволил сохранить точность получаемого результата при уменьшении количества альтернатив решения.

Выводы. При работе над исследованием было введено понятие маркера представления данных на узле РКИС для элементов модели запроса SQL. Разработана функция агрегации, которая на основе статистики SQL-запросов позволяет определить уровень необходимости атрибутов и кортежей отношения базы данных на узле РКИС. Построена модель зависимости критериев оптимальности структуры базы данных узла РКИС от значения маркера представленности данных. Получил дальнейшее развитие метод

DOI 10.15588/1607-3274-2022-2-10

анализа иерархий. Инициализация матрицы попарных сравнений альтернатив может выполняться автоматически в соответствии с полученными математическими моделями. Представление полученного результата в виде вектора нечетких чисел с приведением к точному значению позволяет повысить точность полученных результатов.

КЛЮЧЕВЫЕ СЛОВА: корпоративная информационная система, система управления базами данных, распределенная база данных, SQL-запрос, репликация данных, многокритериальная задача, метод анализа иерархий, нечеткая логика, задача классификации, наивный алгоритм Байеса.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Hamouda S. Document-Oriented Data Schema for Relational Database Migration to NoSQL / S. Hamouda, Z. Zainol // 2017 International Conference on Big Data Innovations and Applications (Innovate-Data), Czech Republic. – 2017. – P. 43–50. DOI: 10.1109/Innovate-Data.2017.13
2. The Definitive Guide to MongoDB / [D. Hows, P. Membrey, E. Plugge, T. Hawkins]. – Berkeley, CA : Apress, 2015. – 343 p. DOI: 10.1007/978-1-4842-1182-3
3. Thakur N. Relational and Non Relational Databases: A Review / N. Thakur, N. Gupta // Journal of University of Shanghai for Science and Technology. – 2021. – Vol. 23, № 8. – P. 117–121. DOI: 10.51201/jusst/21/08341
4. Kundu P. Research of Persistence Solution Based on ORM and Hibernate Technology / P. Kundu, T. Arora // International Journal of Advanced Research in Computer Science and Software Engineering. – 2017. – Vol. 7, № 4. – P. 359–362. DOI: 10.23956/ijarcsse/v7i3/0154
5. Becker J. Systems for the Support of the Company Management, Retail Information Systems Based on SAP Products / Becker J., Uhr W., Vering O. – Berlin : Springer Berlin Heidelberg, 2013. Chapter 5. – P. 121–150. DOI: 10.1007/978-3-662-09760-1_5
6. Petrova E. Overview of modern automation information systems activities of trade enterprises / E. Petrova // Journal of management studies. – 2018. – Vol. 4, № 9. – P. 76–85. DOI: 10.12737/article_5d68d5afb331c1.42407139
7. Christudas B. Practical Microservices Architectural Patterns / B. Christudas. – Berkeley, CA : Apress, 2019. – 812 p. DOI: 10.1007/978-1-4842-4501-9
8. Optimized Transactional Data Structure Approach to Concurrency Control for In-Memory Databases / [C. Peterson, A. Wilson, P. Pirkelbauer et al.] // 2020 IEEE 32nd International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD). – 2020. – P. 107–115. DOI: 10.1109/SBAC-PAD49847.2020.00025
9. Perez L. History-aware query optimization with materialized intermediate views / L. L. Perez, C. M. Jermaine // 2014 IEEE 30th International Conference on Data Engineering. – 2014. – P. 520–531, DOI: 10.1109/ICDE.2014.6816678
10. Tsegelyk G. G. The optimization of data-bases replication in distributed information systems / G. G. Tsegelyk, R. P. Krasniuk // Information Extraction and Processing. – 2017. – Vol. 45, № 121. – P. 104–112. DOI:https://doi.org/10.15407/vidbir2017.45
11. Korniyenko B. Y. Optimization of the Information System of the Corporate Network / B. Y. Korniyenko, L. P. Galata // MCM-TECH, Kamianets-Podilskyi National Ivan Ohiienko University. – 2019. – P. 56–62. DOI: 10.32626/2308-5916.2019-19.56-62
12. Query parsing in order to optimize distributed DB structure / [M. Fisun, M. Dvoretskyi, Shved A. et al.] // 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Bucharest, 2017 : proceeding. – Bucharest: IEEE, 2017. – P. 172–178. DOI: 10.1109/IDAACS.2017.8095071
13. Data Utility Assessment while Optimizing the Structure and Minimizing the Volume of a Distributed Database Node / [M. Dvoretskyi, S. Dvoretska, Y. Nezdoliy et al.] // 1st International Workshop on Information-Communication Technologies & Embedded Systems (ICTES), 2516, 2019 : proceeding. – CEUR Workshop, 2019. – P. 128–137. Available online: http://ceur-ws.org/Vol-2516/paper10.pdf
14. Optimization of the database structure of a distributed corporate information system node using the analytic hierarchy process / [M. Dvoretskyi, S. Dvoretska, H. Horban et al.] // T&I Workshops, 2845, 2020 : proceeding. – CEUR Workshop, 2020. – P. 193–203. Available online: http://ceur-ws.org/Vol-2845/Paper_19.pdf
15. Fisun M. Building a model to optimize the database structure of the node in corporate information systems / M. Fisun, M. Dvoretskiy, S. Dvoretska // Information technology and computer engineering: International Scientific and Technical Journal of Vinnytsia National Technical University. – 2020. – Vol 48, № 2. – P. 52–60. DOI: 10.31649/1999-9941-2020-48-2-52-60
16. Zadeh L. A. Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems / L. A. Zadeh, G. J. Klir, B. Yuan. – World scientific, 1996. – 840 p. DOI: 10.1142/2895
17. Alang-Rashid N. K. A general purpose fuzzy logic code / N. K. Alang-Rashid, A. S. Heger // IEEE International Conference on Fuzzy Systems, 1992 : proceeding. – IEEE, 1992. – P. 733–742. DOI: 10.1109/FUZZY.1992.2587
18. Gozhyj A. Fuzzy cognitive analysis and modeling of water quality / A. Gozhyj, I. Kalinina, V. Gozhyj // 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 2017 : proceeding. – IEEE, 2017. – P. 289–293. DOI: 10.1109/IDAACS.2017.8095092
19. Yager R. R. On inference structures for fuzzy systems modeling / R. R. Yager // IEEE 3rd International Fuzzy Systems Conference. – 1994. – Vol. 2. – P. 1252–1256. doi: 10.1109/FUZZY.1994.343642
20. Fuzzy inference and fuzzy inference processor / [Nakamura K., Sakashita N., Nitta Y. et al.] // IEEE Micro. – 1993. – Vol. 13, № 5. – P. 37–48. DOI: 10.1109/40.238000
21. Dvoretskiy M. Information technology for determining useful data while optimizing the structure and minimizing the volume of the distributed database node / M. Dvoretskiy, S. Dvoretska, E. Davidenko // Bulletin of Cherkasy State Technological University. – 2019. – № 4. – P. 26–35. DOI: 10.24025/2306-4412.4.2019.184808
22. Review on Data Mining and Machine Learning Methods for Student Scholarship Prediction / [R. Hegde, G. V. Anusha, S. Madival et al.] // 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), 2021: proceeding. – IEEE, 2021. – P. 923–927. DOI: 10.1109/ICCMC51019.2021.9418376
23. Zaki M. J. Neural Networks / M. J. Zaki, W. J. Meira // Data Mining and Machine Learning. Cambridge University Press. – 2020. – P. 637–671. DOI: 10.1017/9781108564175.031
24. Graupe D. Deep Learning Neural Networks / D. Graupe. – World scientific, 2016. – 280 p. DOI: 10.1142/10190
25. Janssen J. Naive Bayes / J. Janssen, W. Laatz // Statistische Datenanalyse mit SPSS. – Springer Berlin Heidelberg, 2017. – pp. 557–569. DOI: 10.1007/978-3-662-53477-9_25
26. Krishna S. Introduction to Database and Knowledge-Base Systems / S. Krishna. – World scientific, 1992. – 344 p. DOI: 10.1142/1374

DOI 10.15588/1607-3274-2022-2-10