ISSN 2224-087X. Електроніка та інформаційні технології. 2020. Випуск 13. С. 19–27
Electronics and information technologies. 2020. Issue 13. P. 19–27

UDK 004.8: 004.89: 519.7 DOI: https://doi.org/10.30970/eli.13.2

QUESTION OF THE OPTIMALITY CRITERION OF A REGULAR LOGICAL TREE BASED ON THE CONCEPT OF SIMILARITY

I. Povhan

Department of software systems,
State High School "Uzhgorod National University",
st. Zankovetska №89B, 88000 Uzhgorod, Ukraine
igor.povkhan@uzhnu.edu.ua

This work is devoted to the problem of logical trees – the question of the optimality criterion of a regular logical tree based on the concept of similarity. Solving this issue will allow us to develop effective methods and algorithms for minimizing the structure of logical classification trees. The minimal structure of the logical classification tree provides a fast and efficient mechanism for classifying discrete objects. The obtained result is fundamentally important in the problem of evaluating the effect of permutation of the tiers of the maximum logical classification tree and the question of the structural complexity of the obtained logical trees. Fixing the initial training sample in the form of a logical tree, creates a fixed tree-like data structure, which to some extent provides even compression and transformation of the initial data of the training sample, and therefore allows significant optimization and saving of hardware resources of the information system. The work is relevant for all image recognition approaches in which the resulting classification scheme can be represented as a logical tree.

Key words: tasks of pattern recognition, logical tree, graph-scheme of the model.

Introduced.

This study is a logical continuation of the cycle of works [1-5] which raises fundamental questions related to logical classification trees (in this case, a logical tree will be understood as a graph-schematic representation of the resulting pattern recognition scheme), such as questions of minimizing logical trees, stability studies on the permutation of tiers, estimating the complexity of the largest tree, a general algorithm for building the most complex logical tree [6-8]. Here we study the complexity of graph-schema models (logical classification trees) that are constructed in the process of learning the recognition system (the logical classification tree is actually a generated recognition function) and propose a criterion for the optimality of its structure based on the concept of similarity [9-13]. To study the question of the optimality criterion of logical trees, it is necessary to introduce some definitions at the first stage.

As you know, the functions of $k-$ valued logic, by analogy with two-digit, can be represented in tabular, analytical and graphical forms. In this study, the main attention will be paid to the graphical form of representation of functions of $k-$ valued logic (logical tree). The main idea of which is that an arbitrary $k-$ valued logical function can be represented as a connected graph that does not contain cycles (in graph theory, this type of graph is called a tree). So let's call the specified representation of a logical function-a logical tree (or just a tree).

At the beginning of the study, we will introduce the necessary definitions in the future.

Definition №1. Logical tree (graph) representing a $k-$ valued function $f(x_1, \dots, x_n)$ is a connected graph without cycles, whose non-ring vertices contain the variables $x_1, \dots, x_n$ and the edges are numbered with the values of these variables. The final vertices of the tree contain the values of the function $f(x_1, \dots, x_n)$. Moreover, on a fixed tree path, the same variable occurs only once.

Definition №2. The arrow mark that is included in the top of the tree characterizes the function of the subtree defined by this mark. If all the output arrows of a vertex are marked with the same label α, then the output label of this vertex is marked with the same label α.

Definition №3. If all edges included in the vertex of the logical tree (graph) x_i are marked identically, then the vertex x_i is called similar.

Definition №4. A logical tree (subtree) whose top vertex is similar is called a special tree (subtree). All other logical trees and subtree (graphs) in this study will be called non-personal.

Definition №5. A logical tree (graph function), in the vertices of each horizontal tier of which the same variables (attributes) are written, is called a regular tree. Otherwise, the logical tree (graph function) is called irregular.

Definition №6. Under the break level, we will understand such a tier of some logical tree with the number i, where all functions are different at all vertices (nodes), that is, $2^i = 2^{2^{n-i}}$ (note that here $n-$ is the number of variables of the corresponding function).

Definition №7. Under a non - uniform logical tree, we will understand a tree that has variables of different indexes on certain tiers. For a uniform tree, only the structure in which the variables of the same index are located on a fixed tier is possible.

Definition №8. The most complex logical tree will be called a tree that contains in its structure the maximum number of different labels (vertices, functions).

Definition №9. In this study, we will understand the level of a logical tree as a horizontal row of vertices of a given graph (logical tree) with a changeable single index.

Main part.

Let there be some logical tree that represents the function $f(x_1, x_2, \dots, x_n)$ for $k-$ valued logic from n variables – (Fig. 1). Next, we define the number of characteristic functions $L_{arc}(f)$ in the arc form of the function $f(x_1, x_2, \dots, x_n)$, obtained using this logical tree.

Note that the transitions of the logical tree (graph edges) correspond to the characteristic functions in the arc form, but because there are such vertices and the identity $\varphi_i(x) * 0 = 0$, the number of labels in the tree will be less than the number of edges (transitions). Then we will have the following:

$$L_{arc}(f) = \frac{k^{n+1}-k}{k-1} - S_n$$

logic function from n variables, the value S_n is called the similarity of the logical tree of the corresponding function $f(x_1, x_2, \dots, x_n)$, where $S_n > 0$. Obviously, the optimal logical tree is the one for which $S_n = \max(S_{n_i})$ will be executed.

Note that the similarity value S_n can serve as a criterion for the optimality of a logical tree (graph).

In the next stage of the study, we will consider two possible approaches for finding the similarity of the logical tree S_n.

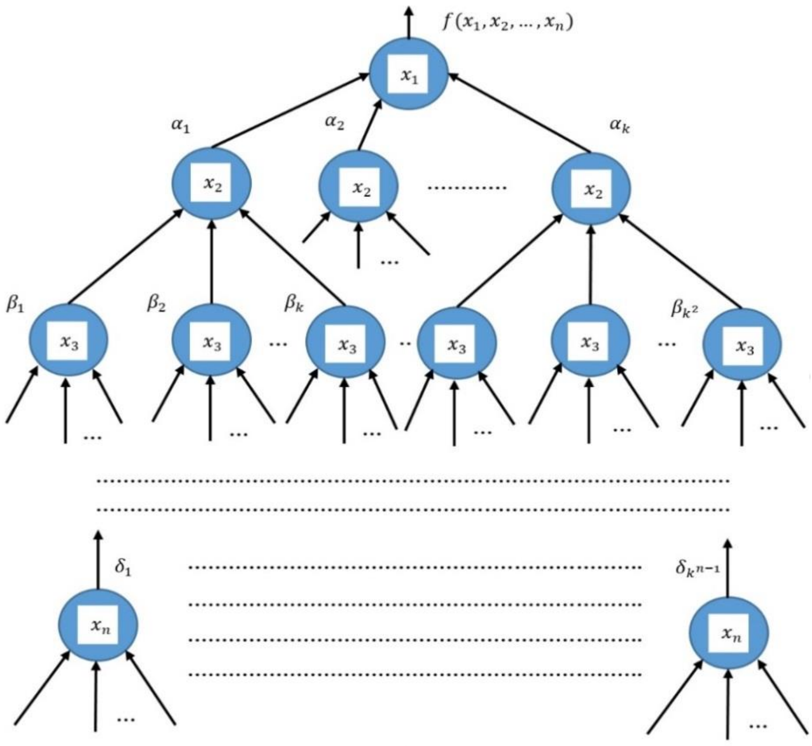

Fig. 1. Constructed logical tree of the function $f(x_1, x_2, \dots, x_n)$ maximum complexity.

Method (A) finding similarity S_n for a logical tree.

Let us have a logical tree of the function $f(x_1, x_2, \dots, x_n)$. There are two possible cases:

a) The top vertex (the top of the first tier) of the logical tree is similar. In this case, the logical tree can be represented as a subtree-the left branch of the initial tree. Let the top vertex of the logical tree be the variable x_i. Then the following formula takes place:

$$L_{arc}(f) = L_{arc}\left(f_{\varphi_0}(x_i)\right) = \frac{k^n - k}{k-1} - S_{n-1}^{\varphi_0^i}. \quad (1\text{ a})$$

Note that here $S_{n-1}^{\varphi_0^i}$ is the similarity of the subtree that represents the left branch of the initial logical tree.

That is, in this case, we will have the following:

$$S_n = k^n + S_{n-1}^{\varphi_0^i}. \quad (1\text{ b})$$

b) The top vertex of the logical tree (the top of the first tier) is not similar, then we will have the following:

$$L_{arc}(f) = L_{arc}\left(f_{\varphi_0}(x_i)\right) + L_{arc}\left(f_{\varphi_1}(x_i)\right) + L_{arc}\left(f_{\varphi_{k-1}}(x_i)\right) + k. \quad (1\text{ c})$$

Note that the formula (1 c) occurs if there is a function $\equiv 0$ on some edge of the logical tree, then from (1 a) and (4), we get that $L_{arc}(f \equiv 0) = -1$.

Note that the number of labels for each of the subtrees of the initial logical tree is defined as follows:

$$L_{arc}\left(f_{\varphi_0}(x_i)\right) = \frac{k^n-k}{k-1} - S_{n-1}^{\varphi_0^i},$$
$$L_{arc}\left(f_{\varphi_1}(x_i)\right) = \frac{k^n-k}{k-1} - S_{n-1}^{\varphi_1^i},$$
… … … … … … … … … … … … … … ….
$$L_{arc}\left(f_{\varphi_{k-1}}(x_i)\right) = \frac{k^n-k}{k-1} - S_{n-1}^{\varphi_{k-1}^i}.$$

Note that here $S_{n-1}^{\varphi_0^i}, S_{n-1}^{\varphi_1^i}, \dots, S_{n-1}^{\varphi_{k-1}^i}$ - similarity of the corresponding subtrees of the initial logical tree.

So finally we will have the following:

$$L_{arc}(f) = \frac{k^n-k}{k-1} - S_{n-1}^{\varphi_0^i} + \frac{k^n-k}{k-1} - S_{n-1}^{\varphi_1^i} + \dots + \frac{k^n-k}{k-1} - S_{n-1}^{\varphi_{k-1}^i} + k =$$
$$= \frac{k^n-k}{k-1} * k + k - S_{n-1}^{f_i} = \frac{k^{n+1}-k^2+k^2-k}{k-1} - S_{n-1}^{f_i} = \frac{k^{n+1}-k}{k-1} - S_{n-1}^{f_i}.$$

So at the output we get:

$$S_{n-1}^{f_i} = \sum_{j=1}^{k-1} S_{n-1}^{\varphi_j^i}. \quad (2)$$

So with (2) we get $S_n = S_{n-1}^{f_i}$.

Then, by combining the results of items (a) and (b), you can present the following:

$$S_n = \begin{cases} k^n + S_{n-1}^{\varphi_0^i}, \ if\ the\ top\ vertex\ of\ the\ tree\ is\ similar. \\ S_{n-1}^{f_0^i}, \ if\ the\ top\ vertex\ of\ the\ tree\ is\ not\ similar. \end{cases} \quad (3)$$

At the next stage, considering a subtree consisting of one vertex of the lower tier of the tree, we come to the value $S_0 -$ similarity of the final edge of the logical tree, and:

$$S_0 = \begin{cases} 1, if\ the\ final\ edge\ corresponds\ to\ zero. \\ 0,\ if\ the\ end\ edge\ does\ not\ match\ zero. \end{cases} \quad (4)$$

Method (B) finding similarity S_n for a logical tree.

Definition №10. The set of cells in the Venn diagram corresponds to sets of variables:

$x_1^{\sigma_1}, x_2^{\sigma_2}, \dots, x_{i-1}^{\sigma_{i-1}}, 0, x_{i+1}^{\sigma_{i+1}}, \dots, x_n^{\sigma_n};$

$x_1^{\sigma_1}, x_2^{\sigma_2}, \dots, x_{i-1}^{\sigma_{i-1}}, 1, x_{i+1}^{\sigma_{i+1}}, \dots, x_n^{\sigma_n};$

… … … … … … … … … … … … … … ….

$x_1^{\sigma_1}, x_2^{\sigma_2}, \dots, x_{i-1}^{\sigma_{i-1}}, k-1, x_{i+1}^{\sigma_{i+1}}, \dots, x_n^{\sigma_n}.$

We will call it a complete system of $x_i -$ compared cells.

Note that the entry $x_l^{\sigma_l}$ means that the variable x_l on this set takes the corresponding value σ_l, ($\sigma_l \in \{0,1,\dots,k-1\}$).

So, if all the cells of an $x_i -$ comparable system of cells have the same number, that is, the same function value, then we will say that they are $x_i -$ like. The number of $x_i -$ like cell systems in the Venn table of the logical function $f(x_1, x_2, \dots, x_n)$ let's call the $x_i -$ similarity of the function. Let's denote this value by S_f^i.

Definition №11. We will call the following value a complete similarity of some logical function:

$$S_f = k * \sum_{i=1}^{n} S_f^i. \tag{5}$$

Next, we introduce the following value for consideration:

$$R_i = k * \sum_i S_{f_{r_i}}^i. \tag{6}$$

Note that the index $f_{r_i} -$ means that the original function of the vertex i is taken (i.e. the vertex where x_i is written). The superscript i means that the function f_{r_i} has similarity only in x_i.

Note that here $f_{r_i} -$ is the output function of the such vertex i. Summation is performed on all such vertices that are not included in the left branches of special subtrees.

Let's introduce another value:

$$R_j = k * \sum_j S_{f_{r_j}}^j. \tag{7}$$

Note that here the analogous $f_{r_i} -$ is the output function of the vertex j that is not similar to the one in which j is written. Summation is performed for all vertices that are not similar.

The similarity based on the finite edges of the logical tree is denoted by Q. in Other words, $Q -$ is the number of zero edges in the logical tree that are taken into account when calculating the similarity using the above method.

Lemma 1. Complete similarity of the logical function $f(x_1, x_2, \dots, x_n)$ can be determined from the following relation:

$$S_f = k * \sum_{l=1}^{n} \sum_{f_{r_l}} S_{f_{r_l}}^l. \tag{8}$$

Note that here $f_{r_l} -$ is the output function of the vertex l of the logical tree. Summation is performed at all vertices of the tree (graph).

Let there be some variable x_{i_1} at the top of the logical tree. The output function of the top vertex of the logical tree coincides with the function $f(x_1, x_2, \dots, x_n)$, so $S_f^{i_1}$ of the function and the logical tree are the same.

Let the left branch of the second tier of the logical tree contain the variable x_{i_2} (at the top of the corresponding subtree). Then none of the sets that correspond to the original functions of the vertices x_{i_2} located in other pid trees of the initial tree can be $x_{i_2} -$ comparable to the sets of the left subtree, since they differ from the variable x_{i_1}.

Similarly, sets that correspond to the original functions of different tree vertices that contain x_{i_2} differ at least in the variable that is located at the vertex where these output functions occur. That is, some vertex of the logical tree is the vertex where the output functions of two other vertices of the logical tree converge, if when moving up the edges from the last two vertices of these functions, their first common vertex (through which they will pass) is the above vertex.

Thus, it was shown that sets that correspond to the original functions of different vertices of this logical tree can not be $x_{i_2} -$ comparable, and therefore can not be $x_{i_2} -$ similar. So, you can write the following:

$$S_f^{i_2} = \sum_{fr_{i_2}} S_{fr_{i_2}}^{i_2}.$$

Note that here summation is performed for all output functions of logical tree vertices that contain i_2.

So, at the next stage-reasoning in the same way for the vertices of a logical tree that correspond to an arbitrary x_l, we come to the statement of the lemma:

$$S_f = k * \sum_{l=1}^{n} S_f^l = k * \sum_l \sum_{fr_l} S_{fr_l}^l .$$

Hence we conclude that the lemma is proved.

Theorem №1. For an arbitrary logical tree that represents some fixed logical function $f(x_1, x_2, \dots, x_n)$ of $k -$ valued logic we have the following:

$$S_n = S_f - R_i - R_j + Q. \quad (9)$$

Note first that the expression $R_i + R_j$_is equal to the sum $x_i -$ similarity for all source functions of all nodes of the logical tree, except those of its vertices that participate in computing the similarity, that is, except for vertices that are included in the left branch of the special subtrees (this follows from the definition of the variables R_i and R_j).

But the similarity of S_n is equal to the similarity due to vertices plus Q. let's denote the similarity due to similar vertices by S_n^1. Then we will have the following:

$$S_n = S_n^1 + Q. \quad (10)$$

But on the other hand it is known:

$$S_n^1 + R_i + R_j = S_f. \quad (11)$$

Next, with (11), we get the following:

$$S_n^1 = S_f - R_i - R_j. \quad (12)$$

At the last stage, substituting (12) into (10) will have the following:

$$S_n = S_f - R_i - R_j + Q.$$

So, we can conclude that the theorem is proved.

Thus, two methods for determining the similarity of the function S_n were presented above. When calculating similarity using a logical tree, it is rational to use the first method.

Example №1. To explain the above methods, consider the example of determining the similarity of a function that is set in the form of the following Venn table – (Table. 1).

Note that when calculating similarity, only those similar vertices of the logical tree that are not included in the right branches of special subtrees and the same zeros are taken into account. Here we used the first method (Method (A) for the logical tree) to determine similarity.

Table 1.

Venn table of function $f(x_1, x_2, x_3, x_4)$ three-digit logic of example №1.

	x_4^0	x_4^1	x_4^2	x_4^0	x_4^1	x_4^2	x_4^0	x_4^1	x_4^2	
	0	0	1	2	2	2	0	0	0	x_2^0
x_1^0	0	0	1	2	2	2	0	0	0	x_2^1
	0	0	1	2	2	2	0	0	0	x_2^2
	1	1	1	0	0	1	2	2	2	x_2^0
x_1^1	1	0	2	1	0	2	1	0	2	x_2^1
	0	0	2	0	0	0	1	1	0	x_2^2
	2	2	2	0	0	0	2	1	0	x_2^0
x_1^2	1	0	1	1	0	1	1	0	1	x_2^1
	0	2	1	0	2	1	0	2	1	x_2^2
		x_3^0			x_3^1			x_3^2		

Note that in the figure, the zeros that were taken into account when calculating the similarity are indicated by lower dashes (underscores).

Let's define the number of characteristic functions in the bracket form of a logical function:

$$L_{arc}(f) = \frac{k^{n+1} - k}{k - 1} - S_n = \frac{3^5 - 3}{2} - 89 = 120 - 89 = 31.$$

The logical tree of the function $f(x_1, x_2, x_3, x_4)$ is shown in (Fig. 2). Note that all such vertices of the logical tree are marked with an upper risk. After determining the similarity of this logical tree, we get $S_n = 89$.

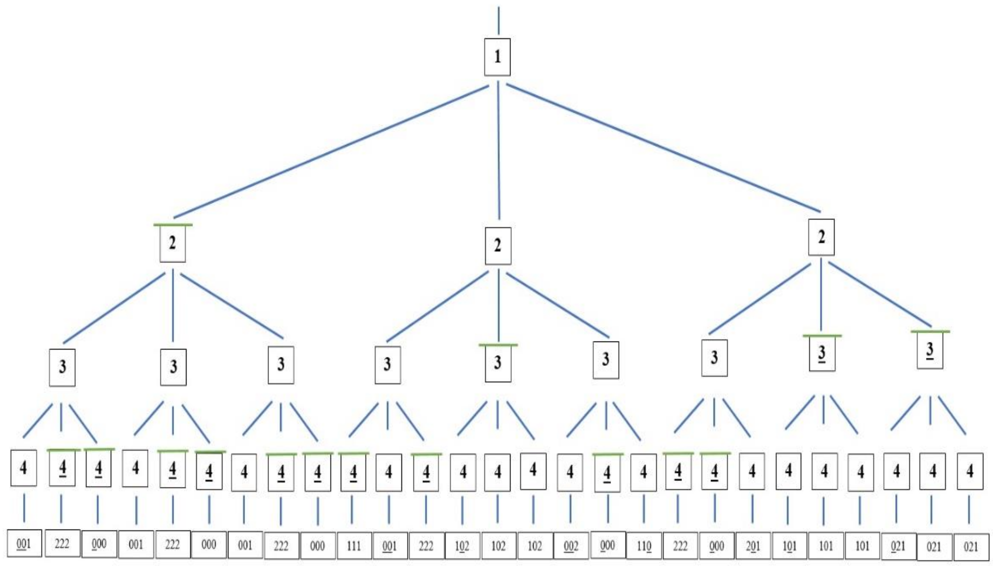

Fig. 2. Logical tree of the function $f(x_1, x_2, x_3, x_4)$ of example №1.

Main result.

In view of all the above mentioned in this paragraph, you can fix the following:

1) Note that the number of characteristic functions in the arc form $L_{arc}(f)$ of the function $f(x_1, x_2, \dots, x_n)$, can be determined using the formula $L_{arc}(f) = \frac{k^{n+1}-k}{k-1} - S_n$.

2) We will call the following value $S_f = k * \sum_{i=1}^{n} S_f^i$ a complete similarity of some logical function .

3) Complete similarity of the logical function $f(x_1, x_2, \dots, x_n)$ can be determined using the following relation $S_f = k * \sum_{l=1}^{n} \sum_{fr_l} S_{fr_l}^l$

4) For an arbitrary logical tree that represents some fixed logical function $f(x_1, x_2, \dots, x_n)$ of $k-$ valued logic we have that $S_n = S_f - R_i - R_j + Q$.

REFERENCES

1. *Povhan I.* Conceptual framework of recognition systems based on the method of extensive array of features / Povhan I., Vasilenko Yu., Vasilenko E. // Scientific and technical journal "European Journal of Enterprise Technologies". – 2004. –№7[1]. – P. 13-15.
2. *Povhan I.* Method of extensive choice of signs in the mathematical construction of multilevel systems of pattern recognition / Povhan I., Vasilenko Yu., Vasilenko E. // Scientific and technical journal "Artificial Intelligence". – 2003. – №7. – P. 246-249.
3. *Povhan I.* Definition of the concept of a sign in the theory of pattern recognition / , I., Vasilenko, Yu., Vassilenko, E. // Scientific and technical journal "Artificial Intelligence". – 2002. – №4. – P. 512-517.
4. *Povhan I.* Calculation of the importance of discrete traits (analysis of some approaches) / Povhan I., Vashchuk F., Vasilenko Yu., Povhan L. // East European journal of advanced technologies. ─ 2010. ─ No. 5/4 (47). - P. 71-75.
5. *Povhan I.* Group and individual assessment of the importance of Boolean arguments / Povhan I., Vasilenko Yu., Vashchuk F., Polischuk V. // Bulletin of national technical University "Kharkiv Polytechnic Institute". – 2011. – No. 53.– P. 57-64.
6. *Srikant R.* Mining generalized association rules. / R. Srikant, R. Agrawal, // Future Generation Computer Systems, 1997, Vol.13, №2, 61–180.
7. *Vasilenko Y.A.* The problem of estimating the complexity of the logic trees, recognition, and a general method of optimization. / Y.A. Vasilenko, F.G. Vashchuk, I.F. Povkhan. // Scientific and technical journal "European Journal of Enterprise Technologies", 2011, 6/4(54), 24-28.
8. *Vasilenko Y.A.* General estimation of tree logical structures minimization. / Y.A. Vasilenko, F.G. Vashchuk, I.F. Povkhan // Scientific and technical journal "European Journal of Enterprise Technologies", 2012, 1/4 (55), 29-33.
9. *Povkhan I.* General scheme for constructing the most complex logical tree of classification in pattern recognition of discrete objects. / I. Povkhan // "Electronics and information technology", 2019, Issue 11, 112-117.
10. *Laver V.O.* Algorithms for constructing logical classification trees in pattern recognition problems. / V.O. Laver, I.F. Povkhan // Scientific notes of Tauride national University. Series: technical Sciences, Volume 30(69) No. 4 - 2019, 100-106.
11. *Vtogoff P.E.* Incremental Induction of Decision Trees. / P.E. Vtogoff // Machine Learning, 2009, № 4, 61−186.

12. *Povkhan I.F.* The problem of functional evaluation of the training sample in the problems of recognition of discrete objects. / Povkhan I.F. // Scientific notes of Taurida national University. Series: technical Sciences, 2018, Volume 29(68) №.6, 217-222.
13. *Whitley D.* An overview of evolutionary algorithms: practical issues and common pitfalls. / D. Whitley // Information and Software Technology, 2001, Vol.43, №14, P. 817–831.

ПИТАННЯ КРИТЕРІЮ ОПТИМАЛЬНОСТІ РЕГУЛЯРНОГО ЛОГІЧНОГО ДЕРЕВА, ЗАСНОВАНОГО НА ПОНЯТТІ ПОДІБНОСТІ

І. Повхан

Кафедра програмного забезпечення систем,
ДВНЗ "Ужгородський національний університет",
вул. Заньковецкої №89 Б, 88000 Ужгород, Україна
igor.povkhan@uzhnu.edu.ua

Дана робота присвячена проблематиці логічних дерев – питанню критерію оптимальності регулярного логічного дерева на основі поняття подібності. Розв'язок даного питання дозволить забезпечити розробку ефективних методів та алгоритмів мінімізації структури логічних дерев класифікації. Мінімальна структура логічного дерева класифікації забезпечує швидкий та ефективний механізм класифікації дискретних об'єктів. Вводиться поняття ярусу злому конструкції логічного дерева, яке дозволяє в перспективі оцінити межі мінімізації таких структур шляхом простої перестановки ярусів.

Отриманий результат принципово важливий в задачі оцінки ефекту перестановки ярусів максимального логічного дерева класифікації та питанні структурної складності отриманих логічних дерев. Фіксація початкової навчальної вибірки у вигляді логічного дерева, створює фіксовану деревоподібну структуру даних, яка в якісь мірі забезпечує навіть стиск та перетворення початкових даних навчальної вибірки, а отже дозволяє суттєву оптимізацію та економію апаратних ресурсів інформаційної системи.

Відмітимо, що галузь застосування концепції логічних дерев в даний час надзвичайно об'ємна, а множина задач та проблем, які розв'язуються даним апаратом може бути зведена до наступних трьох базових сегментів – задачі опису структур даних, задачі розпізнавання та класифікації, задачі регресії. В роботі вводиться величина подібності логічного дерева, яка може слугувати критерієм оптимальності даного логічного дерева (графа), а саме дослідження має актуальність для всіх підходів, методів та алгоритмів розпізнавання образів в яких отримана результуюча схема класифікації може бути представлена у вигляді логічного дерева.

Ключові слова: задачі розпізнавання образів, логічне дерево, граф-схема моделі, критерій оптимальності.

Стаття: надійшла до редакції *23.01.2020,*
доопрацьована *03.02.2020*
прийнята до друку *05.02.2020*