Predicate Clustering Method and its Application in the System of Artificial Intelligence

Ganna Proniuk [1], Nataliia Geseleva [2], Iryna Kyrychenko [1] and Glib Tereshchenko [1]

[1] *Kharkiv National University of Radio Electronics, Nauky Ave, 14, Kharkiv, 61166, Ukraine*
[2] *State University of Trade and Economics, Kyoto str., 19, Kyiv, 02156, Ukraine*

Abstract

The paper proposes a method for clustering predicates of arbitrary dimensionality. To this end, a theorem of the general form of first-order predicate is formulated, which justifies the first-order two-layer decomposition method of predicate, resulting in a predicate defined on a set of significantly smaller power than the original. This allows for the identification of conditions for the most effective identification of processes of human intellectual activity using the generalized comparator identification method. Additionally, a second-order two-layer decomposition method of predicate is developed, which is based on the concept of the general form of second-order predicate. Based on the combination of the concepts of two-layer decomposition of first and second-order predicate, a method of three-layer decomposition of predicate is developed. The resulting method of multilayer decomposition of predicate is suitable for building electronic circuits that implement arbitrary relationships. The predicate scheme is widely parallelized, resulting in its high performance.

Keywords

Text classification, predicate algebra, equivalence predicate, general form of predicate, text understanding, isomorphism, decomposition, comparator identification, three-layer decomposition

1. Introduction

In recent years, great attention has been paid to the development of new methods for parallel information processing. Practically all existing approaches [1-3] are based on the concepts of decomposition and composition in one way or another - be it programming, databases, neural networks, and so on. Therefore, the development of new theoretical methods of decomposition is a highly promising and interesting direction in algebraic logic. This work proposes a new method of multi-layer decomposition of predicates, which will be obtained from the generalized properties of comparator identification.

In predicate algebra, comparator identification is widely used, which is a type of indirect identification [4, 5]. A comparator K is a device with m inputs $y_1, y_2, \ldots, y_m$ and one output t, where $t \in \{0, 1\}$ is the binary reaction of the comparator. The comparator determines whether its input signals $y_1, y_2, \ldots, y_m$ are in a given relation K or not. Comparator identification is successfully applied in solving many artificial intelligence problems. In [5], its capabilities for alternative evaluation models in decision-making systems are shown. In [6], interesting results were obtained for the theory of color vision using the comparison method. This method is designed for investigating objects with input signals that are inaccessible for direct measurement. The subject of such identification is often the human intelligence.

COLINS-2023: 7th International Conference on Computational Linguistics and Intelligent Systems, April 20–21, 2023, Kharkiv, Ukraine
EMAIL: ganna.proniuk@nure.ua (G. Proniuk); gesnata@ukr.net (N. Geseleva); iryna.kyrychenko@nure.ua (I. Kyrychenko); hlib.tereshchenko@nure.ua (G. Tereshchenko)
ORCID: 0000-0001-7648-0360 (G. Proniuk); 0000-0001-9188-9738 (N. Geseleva); 0000-0002-7686-6439 (I. Kyrychenko); 0000-0001-8731-2135 (G. Tereshchenko)

In predicate algebra, during the comparator identification process, the identifiable object P implements a predicate $P(x_1, x_2, \ldots, x_m) = K(f_1(x_1), f_2(x_2), \ldots, f_m(x_m))$, referred to as the predicate of object P. The simplest task of comparator identification is to mathematically describe the output signals $y_1, y_2, \ldots, y_m$ of processes $f_1, f_2, \ldots, f_m$ and the processes themselves based on the given comparator and the known properties of object P. The comparator's behavior implements a predicate $K(y_1, y_2, \ldots, y_m)=t$, corresponding to the relation K. The identifiable processes $f_1, f_2, \ldots, f_m$. are connected to the inputs of the comparator by their outputs. Here, $x_1 \in A_1$, $x_2 \in A_2, \ldots, x_m \in A_m$ are input signals of processes, while $y_1 \in B_1$, $y_2 \in B_2, \ldots, y_m \in B_m$ are their output signals. $A_1, A_2, \ldots, A_m$ are sets of input signals of processes, and $B_1, B_2, \ldots, B_m$ are sets of output signals of processes.

2. Generalization of the comparator identification method

Let us consider the main concepts that will help us generalize the method of comparator identification to a wider class of predicates.

Let $E(x, y)$ be a predicate defined on the Cartesian product of a non-empty set M. Then the predicate E is called [7–9]:

- reflexive if it satisfies the condition $\forall x \in M\ E(x, x)$;
- symmetric if it satisfies the condition $\forall x, y \in M\ (E(x, y) \supset E(y, x))$;
- transitive if ∀x,y,z∈ $\forall x, y, z \in M$, from xEy and yEz result xEz.

Any reflexive, symmetric, and transitive predicate is called an equivalence predicate [1, 7].

Let N be a non-empty set, f – be a surjective function mapping set M to set N and let D – be the equality predicate defined on $N \times N$ with the condition: $\forall x, y \in N\quad D(x, y) = \exists a \in N\ x^a y^a$.

It is known [8, 9] that any predicate E on $M \times M$ expressed for any $x, y \in M$ as

$$E(x, y) = D(f(x), f(y)), \tag{1}$$

is an equivalence predicate. The function f is called the characteristic function of the equivalence predicate.

In predicate algebra, any equivalence predicates, and only they, can be represented in the general form (1) with a suitable choice of the set N and function f. From a mathematical point of view, this result is trivial, but it is very important for the theory of comparator identification because it indicates the necessary and sufficient features that can always be used to determine whether an object implementing the predicate E can be identified by the comparator method.

If a system implements the predicate $t=E(x, y)$, and this predicate satisfies the conditions of reflexivity, symmetry, and transitivity, then it can be identified using the comparator method. However, if at least one of these three conditions is not satisfied, then the comparator method is not applicable for such an object. These results of comparator identification can be applied to any physical objects that satisfy the aforementioned conditions.

Above it has been shown that a pair (N, f), where $f: M \to N$, determines a unique equivalence predicate $E(x, y) = D(f(x), f(y))$ on the set $M \times M$. However, does every equivalence predicate E uniquely determine the pair (N, f)? It turns out, no. There exist different pairs (N, f) and (N', f') that define the same equivalence predicate E. The property formulated below specifies the necessary and sufficient condition under which two pairs (N, f) and (N', f') determine the same equivalence predicate E.

Statement 1. In order for two pairs (N, f) and (N', f') to define the same equivalence predicate E on the Cartesian product of the set M, it is necessary and sufficient for there to exist a bijection T between the domain of N and the range of N' such that for all $x \in M\ f'(x)=T(f(x))$.

From Statement 1 it follows that if the equivalence predicate $E(x, y)$ can be represented as (1) for any $x, y \in M$, then it can also be represented as

$$\mathrm{E}(x, y) = D(T(f(x)), T(f(y))), \tag{2}$$

where T is an arbitrarily chosen bijection.

It follows from statement 1 that if the predicate E is represented by two different methods $E(x, y)= =D(f(x), f(y))=D(f'(x), f'(y))$, then there always exists a bijection T that links the functions f and f' by the dependence $f'(x)=T(f(x))$, which holds for any $x \in M$. Therefore, it is impossible to specify a unique characteristic function f for the equivalence predicate E.

Thus, if some function f is found that mathematically describes the identification object, then an entire family of other functions can also claim to describe this object. In other words, the output signals of the identification object through comparator identification allow for various options for mathematical descriptions. Such multiplicity of object representation may indicate the incompleteness of its description by the comparator identification method and, consequently, the disadvantage of this method compared to classical direct identification methods. In fact, the degree of completeness of object description in these two identification methods is absolutely the same. The fact is that in direct identification, the object description is obtained only by virtue of the fact that the method of describing its output signals was chosen before the identification process began. In the case of comparator identification, however, the method of describing the output signals is chosen in the identification process itself, and this is precisely what leads to multiple descriptions of the object.

It is known that the comparator identification method describes an object up to isomorphism [4, 9]. Essentially, this means that comparator identification, just like direct identification, provides a unique description of the object up to notations.

To solve problems of comparator identification, an important question is the issue of isomorphism of equivalence characteristic functions.

Predicates P and P' on $A \times B$ и $A' \times B'$, are called weakly isomorphic (or simply isomorphic) if there exist bijective functions φ: $A \to A'$ and ψ: $B \to B'$, such that for all $x \in A$ and $y \in B$ the equality is satisfied:

$$P(x, y) = P'(\varphi(x), \psi(y)). \tag{3}$$

We also say that the predicate P(x, y) is isomorphic to the predicate P'. The bijections φ and ψ that satisfy condition (3) are called left and right isomorphisms of the predicates P and P'.

The predicates $P(x, y)$ and $P'(x', y')$ on sets $A \times B$ and $A' \times B'$ are called strongly isomorphic if there exists a bijection φ: $A \cup B \to A' \cup B'$, such that for all $x \in A$ and $y \in B$, the equation is satisfied

$$P(x, y) = P'(\varphi(x), \varphi(y)). \tag{4}$$

We will also say that predicate P is φ-isomorphic to predicate P'. A bijection φ that satisfies condition (4) is called an isomorphism of predicates P and P'.

The concepts of weak and strong isomorphisms of predicates play an important role in the theory of comparator identification. The point is that the choice of designations for the signals of the identified system is within the power of the researcher and is determined by the unit's system adopted by him. If two researchers studying the behavior of the same system use different designations for its input signals, they will obtain different predicates for it. If all input signals of the same system under study are recorded by each researcher in a single (but their own) units' system, then the predicates obtained by them will be strongly isomorphic, and if they are recorded in different systems, the predicates will be weakly isomorphic. In this case, it is said that the studied systems are identified up to designations (common or separate). In the case of strong isomorphism of predicates, it is said that the identified systems coincide up to designations in a single unit's system. In the case of weak isomorphism of predicates, it is said that the identified systems coincide up to designations in different units' systems.

Statement 2. If $A \cap B = \varnothing$ and $A' \cap B' = \varnothing$, then weakly isomorphic predicates P and P', defined on $A \times B$ and $A' \times B'$, will also be strongly isomorphic.

In essence, this property means the following: two different descriptions of the same identifiable system $P(x, y)$, whose input signals are defined on non-intersecting domains, always coincide up to strong isomorphism, i.e., they coincide up to the designation of input signals x and y of the system P described in a unified notation system.

Statement 3. If the equivalence predicates E and E' on sets $A \times A$ and $A' \times A'$ weakly isomorphic, then they are also strongly isomorphic.

The substantive content of property 3 means that the signals x and y of the equivalence predicate $E(x, y)$ cannot be described in different notation systems, but only in the same one. To describe the system $E(x, y)$ with an equivalence model, a researcher must express its output signals x and y in a single notation system.

Statement 4. Let E be an equivalence predicate on $A \times A$ and f: $A \to B$ be its characteristic function. Then, equivalence relation E is isomorphic to equality relation D on $B \times B$ if and only if f is injective.

This statement determines in which cases a human perceives complete information about objects and in which cases not. Information is not lost when a person's sense organs assign a subjective image (regardless of what it is) to each object. However, if the number of images is less than the number of perceived objects, some information about the objects is lost. For example, the human eye loses some information about light when perceiving light radiation. This is proven by the existence of different types of light radiation that appear as the same color to the eye. For instance, there is a mix of red and green monochromatic radiation that appears the same color as yellow monochromatic radiation.

Let us now turn to the study of isomorphism of characteristic functions of equivalences. Suppose we have a signal transformer that implements a function $y=f(x)$, which maps set A to set B. By renaming its input and output signals x and y using bijective maps φ: $A\to B$ and ψ: $A\to B$, we obtain $x'=\varphi(x)$, $y'=\psi(y)$. As a result, the same signal transformer is now described by a different function $y'=f'(x')$, which maps set A' to set B'. Using the inverse function, denoted by φ^{-1}, we express the function f' in terms of f: $f'(x') = \psi(f(\varphi^{-1}(x'))$. Similarly, the function f is expressed in terms of f':

$$f(x) = \psi^{-1}(f'(\varphi(x)), \tag{5}$$

where ψ^{-1} is the inverse function of the bijection ψ.

Let E and E' be equivalences on $A\times A$ and $A'\times A'$; D and D' be equality predicates on $B\times B$ and $B'\times B'$.

Statement 5. If the predicate E is φ-isomorphic to the predicate E', then there exists a bijection ψ: $A\to B$, such that the function f (φ, ψ)-isomorphic to the function f', and the predicate D is ψ- isomorphic to the predicate D'.

Statement 6. If the function f (φ, ψ)-isomorphic to the function f', then the predicate E is φ- isomorphic to the predicate E', and the predicate D is ψ-isomorphic to the predicate D'.

It directly follows from statements 5 and 6 that the following property holds.

Statement 7. For the equivalence E to be φ-isomorphic to the equivalence E', it is necessary and sufficient for the function f to be (φ, ψ)-isomorphic to the function f'.

The substance of statements (5)–(7) means that the behavior of $E(x, y) = D(f(x), f(y))$ of the identified system E, is fully determined (i.e., up to notation) by the action of the identified object f and vice versa. In addition, the action of the zero organ $D(u, v)$ is fully determined both by the behavior of the system E and the action of the object f. All of the above indicates that the comparator method is an effective means of identifying the object f, the internal state $u=f(x)$ of the system E, and the zero organ $D(u, v)$.

There is some inequality between the external behavior E of the test and the corresponding internal information process f since strong isomorphism of the predicates E and E' corresponds to weak isomorphism of the functions f and f'. The following statement establishes a condition under which the predicate E and the function f become equal in this sense.

Statement 8. In order for the φ-isomorphism of any equivalences E on $A\times A$ and E' on $A'\times A'$ to be equivalent to the φ-isomorphism of their characteristic functions f: $A\to B$, f': $A'\to B'$, it is necessary and sufficient for the sets A and B, A' and B' to be disjoint.

Statement 8 states that if it is required that objects and their images can be measured in the same system of physical units and always obtain the system's action in the form of an equivalence predicate, it is necessary to ensure that the set of all analyzed objects and the set of their images do not intersect. For example, when creating an artificial color vision system, colors as physical objects should be represented not by light emissions, but by some physical processes, such as magnetic fields.

Thus, a comparator identification method is described for mathematically describing subjective phenomena. The behavior predicate P of the subject allows determining the set of images or thoughts, as well as the intellectual functions of the human (perception, understanding, recognition) uniquely up to isomorphism. Human behavior in many cases allows description using an equivalence predicate. The question arises: is there another general form of a binary predicate, and what is it? Undoubtedly, there must be some general expression that gives some binary predicate. To answer this question, consider the following concepts.

3. First-order two-layer decomposition of predicate

Let us consider an arbitrary binary predicate P defined on $A_1\times A_2$, and seek a representation for it, in which the comparison of the values of the two corresponding functions f_1 and f_2 is carried out using a simple predicate, in some sense. This predicate should replace the equality predicate in formula (1). In order to obtain the required form of the predicate, let us first consider several important concepts.

Statement 9. On accompanying equivalences. For any predicate P defined on the Cartesian product $A\times B$, the predicates E_L on $A\times A$ and E_R on $B\times B$ of the form

$$E_L(x_1,x_2) = \forall y\in B\ (P(x_1,y) \sim P(x_2,y)), \quad (6)$$

$$E_R(y_1,y_2) = \forall x\in A\ \left(P(x,y_1)\sim P(x,y_2)\right) \quad (7)$$

are equivalences.

Statement 10. Generalization of the theorem on accompanying equivalences for a predicate of arbitrarity. For any predicate P $(x_1, x_2, \dots x_n)$ on $A_1\times A_2\times\ldots\times A$, the predicates E_i on $A_i\times A_i$ $(i = \overline{1,n})$ of the form

$$E_i(x'_i,x''_i) = \forall x_1 \in A_1\ \forall x_2 \in A_2 \ldots \forall x_{i-1} \in A_{i-1} \forall x_{i+1} \in A_{i+1} \ldots \forall x_n \in A_n \quad (8)$$
$$P(x_1, x_2, \ldots, x_{i-1}, x'_i, x_{i+1}, \ldots, x_n) \sim P(x_1, x_2, \ldots, x_{i-1}, x''_i, x_{i+1}, \ldots, x_n)$$

are equivalences.

The predicates E_L and E_R, defined by expressions (6) and (7), are called accompanying equivalences (left and right) of the predicate P. The predicate E_i, defined by expression (8), is called the i-th accompanying equivalence of the predicate P.

Let E and E_1 be equivalences on $A\times A$. We will say that the equivalence E is embedded in the equivalence E_1 and write $E\leq E_1$, if for any $x, y\in A$ from $E(x, y)=1$ then $E_1(x, y)=1$. If $E\leq E_1$ and $E\neq E_1$, then we will write $E < E_1$ and say that the equivalence E is strictly embedded in the equivalence E_1. If $E < E_1$, we will say that the partition R corresponding to the equivalence E is finer than the partition R_1, corresponding to the equivalence E_1. We will also say that the partition R_1 is coarser than the partition R. If $E\leq E_1$, then we will say that the partition R is finer than or equal to the partition R_1. If $E\leq E_1$, then the partition R corresponding to the equivalence E is called a sub-partition of the partition R_1, corresponding to the equivalence E_1. It is easy to see that the embedding relation defined on the set of equivalence predicates is reflexive, transitive, and antisymmetric, i.e., it is a partial order relation.

Theorem 1. On the general form of a binary predicate of the 1-st order. Let P be a predicate on $A_1\times A_2$; E_L and E_R be its accompanying equivalences on $A_1\times A_1$ and $A_2\times A_2$, respectively; E_1 on $A_1\times A_1$ and E_2 on $A_2\times A_2$ be equivalences that satisfy the conditions $E_1\leq E_L$, $E_2\leq E_R$; f_1: $A_1\rightarrow B_1$ and f_2: $A_2\rightarrow B_2$ be characteristic functions of the equivalences E_1 and E_2 respectively. Then there exists a unique predicate L on $B_1\times B_2$, such that for any $x\in A_1$ and $y\in A_2$

$$P(x,y) = \mathrm{L}\left(f_1(x), f_2(y)\right). \quad (9)$$

Expression (9) represents the general form of a binary predicate P. Surjections f_1 and f_2 are called the characteristic functions (left and right) of predicate P. Predicate L is called the image of predicate P under equivalences E_1 and E_2. Equivalences E_1 and E_2 can be taken as accompanying equivalences E_L and E_R, in which case predicate L takes on the simplest form and is called the absolute image of predicate P.

Below, we describe a method for finding the image of a predicate. Given P, f_1 and f_2, predicate L is found using the formula:

$$L(v,w) = \mathrm{P}(f_1^{-1}(v), f_2^{-1}(w)), \quad (10)$$

where f_1^{-1} and f_2^{-1} are inverse mappings of surjections f_1 and f_2.

Theorem 1 can be extended to the case of an arbitrary n-ary predicate $P(x_1, x_2, \ldots, x_n)$. In this case, the theorem can be formulated as follows.

Statement 11. Let $\underline{P}$ be a predicate on на $A_1\times A_2\times\ldots\times A_n$, E_{ic} be its accompanying equivalence relation on A_i $(i = \overline{1,n})$, E_i be an equivalence relation satisfying the condition $E_i\leq E_{ic}$, f_i: $A_i\rightarrow B_i$ be the characteristic function of equivalence relation E_i. Then there exists a unique predicate L on $B_1\times B_2\times\ldots\times B_n$ such that for any $x_1\in A_1$, $x_2\in A_2$,…, $x_n\in A_n$

$$P(x_1,x_2,\ldots,x_n) = L(f_1(x_1), f_2(x_2), \ldots, f_n(x_n)). \quad (11)$$

Expression (11) represents the general form of an n-ary predicate P.

By representing the predicate in its general first-order form (9, 10), we have achieved that the maximum amount of information carried by predicate P about the relationships between the elements of sets A_1 and A_2, has been transferred to functions f_1 and f_2, while the comparator bears minimal burden.

The following property states that by this comparator identification method, n objects f_1, f_2, ..., f_n. can be identified exhaustively (i.e., up to notations). This means that even in the most general case, the depth of analysis of objects using the comparator identification method is not inferior to that of direct identification method.

Statement 12. Let the predicate $P(x_1, x_2,..., x_n)$ be defined on the set $A_1 \times A_2 \times ... \times A_n$, the predicate $P'(x'_1, x'_2,..., x'_n)$ on the set $A'_1 \times A'_2 \times ... \times A'_n$, and let the predicates $L(v_1, v_2, ..., v_n)$ and $L'(v'_1, v'_2, ..., v'_n)$ be defined on the sets $B_1 \times B_2 \times ... \times B_n$, $B'_1 \times B'_2 \times ... \times B'_n$ respectively, where L is the image of predicate P under equivalences $E_1 \times E_2 \times ... \times E_n$, and L' is the image of predicate P' under equivalences $E'_1 \times E'_2 \times ... \times E'_n$. Suppose that the predicates P and P' are $(\varphi_1, \varphi_2,..., \varphi_n)$-isomorphic, and that the predicates E_i and E'_i are φ_i-isomorphic, where φ_i: $A_i \to A'_i$, $i=\overline{1,\ \ n}$. Then there exist bijections ψ_i: $A_i \to A'_i$, $i=\overline{1,\ \ n}$, such that the predicates L and L' $(\psi_1, \psi_2, ..., \psi_n)$ are isomorphic.

Identification of human intellectual activity using this scheme opens the way to a mathematical description and artificial reproduction of such important aspects of the mind for machine intelligence as perception, understanding, recognition, and awareness. Undoubtedly, subjective states in the human brain are implemented in some, as yet poorly understood, material structures and processes. Clearly, direct identification methods are unacceptable in this case, since images of situations and meanings of texts, being subjective states of a human being, are inherently inaccessible to direct physical measurement.

A first-order two-layer decomposition of predicate is called its decomposition into characteristic functions and images according to its general first-order form. The most important case for practice is the decomposition using accompanying equivalences. There is also a case of using equivalences nested in accompanying equivalences.

4. Second-order two-layer decomposition of predicate

It was previously mentioned that any equivalence predicate can be represented in the form (1), where f: $A \to B$ is a surjection, B is the set of images of the objects in set A; $u= f(x)$ is the image of object x. The question arises: what form will the predicate E (type of predicate) take if an arbitrary mapping is used instead of a surjection f? To answer this question, the general form of the predicate needs to be slightly modified. The following statement provides the required modified form of the predicate E.

Statement 13. On the variant of the general form of the equivalence predicate. Let $F(x, u)$ be a predicate on $A \times B$, corresponding to the surjection f: $A \to B$, $f(x)=u$. Then the predicate E, whose values for any $x, y \in A$ are expressed as

$$E(x,y) = \exists u \in \mathrm{B} \left(F(x,u) \wedge F(y,u)\right), \tag{12}$$

is an equivalence relation on A.

And vice versa: For any equivalence predicate E on A, there exist a set B and a well-defined, one-to-one, and surjective predicate F on $A \times B$, such that for any $x, y \in A$ equality (12) holds.

The predicate $F(x, u)$ is called the characteristic predicate of the equivalence. It uniquely determines the characteristic function f of the equivalence. It is important to have a method for constructing the characteristic predicate for any equivalence.

Let us ask the following question: What kind of predicate E will we obtain if, in equality (12) that characterizes the general form of predicate E, we take an arbitrary predicate $F(x, u)$ on $A \times B$ instead of a surjective, one-to-one, and well-defined predicate F? That is, instead of the surjection f: $A \to B$, we take an arbitrary mapping $f(x)=u$, acting from A to B. To answer this question, we will consider some concepts.

The predicate E on $A \times A$ is called quasi-reflexive if it satisfies the condition

$$\forall x \in A(\left(\exists y \in A\left(E(x,y) \vee E(y,x)\right)\right) \supset E(x,x). \tag{13}$$

This property implies that the predicate E is reflexive, but not on the entire set A, but on some of its subset A', defined by the formula $A^{'}(x) = \exists y \in BE(x,y)$.

On the domain $A' \times A'$ the predicate E is reflexive, but outside of it, i.e., for any $x \notin A'$ or $y \notin A'$, this predicate becomes zero $E(x, y) \equiv 0$.

A reflexive and symmetric predicate is called tolerant. A quasi-reflexive and symmetric predicate are called quasi-tolerant. After the natural restriction of the domain of the predicate E from A to $A' \subseteq A$, the quasi-reflexive predicate E on $A \times A$ becomes a reflexive predicate E on $A' \times A'$.

Statement 14. On the general form of a tolerant predicate. Let E be a predicate on $B\times B$. Then E is tolerant if and only if there exist a set A and a predicate F on $B\times A$, such that

a) for any $x, y\in B$

$$E(x,y) = \exists u\in A\ \big(F(x,u)\wedge F(y,u)\big); \tag{14}$$

b) for any $x\in B$, the condition $\exists u\in A\ F(x, u)$ is satisfied.

The expression of tolerance E given by formula (14) is referred to as its general form, and the predicate F is the characteristic predicate of tolerance. The mapping f corresponding to predicate F is called the characteristic mapping of tolerance.

If all restrictions are removed from the predicate F, then the following theorem on the general form of quasi-tolerance predicate [10] holds, according to which the predicate E on $B\times B$ is a quasi-tolerance if and only if there exists a set A and a predicate F on $B\times A$, such that equality (14) holds for any $x, y\in B$.

If we replace the surjection f with an arbitrary (in general, partial and multivalued) mapping in the equivalence scheme $E(x, y) = D(f(x), f(y))$, we obtain the quasitolerant predicate $E(x, y) = \exists\, u, u'\in B$ $(F(x, u)\wedge F(x, u')\wedge D(u, u'))$. The equality predicate $D(u, u')=1$, if at least one of the values of $f(x)=f(y)$ coincides, and F is a mapping defined everywhere. If we change the surjection $f\colon A\rightarrow B$ in the scheme to the function $f\colon A\rightarrow B$, then nothing will change – we will get any equivalence on the left. Thus, quasitolerance is the most general case of a symmetric predicate.

In conclusion, it should be noted that the general form for the most general case of a symmetric predicate, i.e., the quasitolerant predicate, has been obtained. If we remove the last restriction - the symmetry of the predicate E – the following theorem will be valid.

Theorem 2. On the general form of a binary predicate of the 2-nd order. For any binary predicate E on $A_1\times A_2$, there exist a set B and predicates F_1 on $A_1\times B$ and F_2 on $A_2\times B$ such that for any $x_1\in A_1$, $x_2\in A_2$, the following equality holds:

$$E(x_1,x_2) = \exists\, u\in \mathrm{B}\ \big(F_1(x_1,u)\wedge F_2(x_2,u)\big). \tag{15}$$

The formula (15), which represents the general form of a 2nd-order predicate $E(x_1, x_2)$, can be expressed differently as:

$$E(x_1,x_2) = \exists\, u\in \mathrm{B}\ \big(F_1(x_1,u)\wedge F_2(x_2,u)\big) = D_B(h_1(x_1),h_2(x_2)), \tag{16}$$

where D_B is a symmetric and reflexive predicate, which we define as follows

$$\forall M_1,M_2\subseteq \mathrm{B}\quad D_B(M_1,M_2)\sim\big(\exists u\in \mathrm{B}\ \ M_1(u)\wedge M_2(u)\big). \tag{17}$$

It should be noted that in formula (16), h_1 and h_2 are not functions, as in the general form (9) of a 1st-order binary predicate, but rather mappings, i.e., objects of a more general nature than functions. The predicate F_1, which appears in expression (15), is called the left characteristic predicate of the predicate E, and F_2 is the right characteristic predicate.

Statement 15. Generalization of the general view theorem of the 2-nd order into *n*-ar predicates. For any predicate E on $A_1\times A_2\times\ldots\times A_n$, there exist a set B and predicates F_i on $A_i\times B$ $(i=\overline{1,n})$ such that for any $x_1\in A_1$, $x_2\in A_2$, …, $x_n\in A_n$ the equality holds:

$$E(x_1,x_2,\ldots,x_n) = \exists\, u\in \mathrm{B}\ \Big(F_1(x_1,u)\wedge F_2(x_2,u)\wedge \ldots \wedge F_n(x_n,u)\Big), \tag{18}$$

where characteristic predicates F_i on $A_i\times B$ $(i=\overline{1,n})$ of the predicate E on $A_1\times A_2\times\ldots\times A_n$ can be found using the formula:

$$F_i(x_i,u) = \exists x_1\in A_1\exists x_2\in A_2 \ldots \exists x_{i-1}\in A_{i-1}\exists x_{i+1}\in A_{i+1}\ldots\exists x_n\in A_n \\ \mathrm{S}(x_1,x_2,\ldots,x_n,u), \tag{19}$$

where S is a function that assigns different names u to all sets $(x_1, x_2, \ldots, x_n)$, for which $E(x_1, x_2, \ldots, x_n) = 1$; B is the set of all such names.

Expression (18) represents the general form of the 2nd kind predicate $E(x_1, x_2, \ldots, x_n)$ on $A_1\times A_2\times\ldots\times A_n$.

In turn, the 2nd-order two-layer decomposition of a predicate is a representation of the predicate in its general 2nd-order form using formula (16). Thus, the representation of any predicate in its general 2nd-order form has been considered. In this form of predicate representation, a certain classifying function appears that assigns names to all sets of variables. This property is very useful in describing the structures of many information objects (such as databases, microchip design) [11, 12].

5. Three-layer predicate decomposition

In the previous sections, two types of two-layer decomposition of predicates were considered – the first and second orders. Their combination results in a three-layer decomposition of the predicate, which completes the construction of the method of multi-layer decomposition of predicates. The first order of decomposition transforms the predicate E into a construction $E(x, y) = L(f_1(x), f_2(y))$, where f_1 and f_2 are functions, and L is a simpler predicate than E (defined on a set of smaller cardinality). The 2-nd order of decomposition transforms the predicate E into a construction of the form $E(x, y) = D_B(h_1(x), h_2(y))$, where h_1 and h_2 are mappings, i.e., objects of a more general nature than functions; D_B is a predicate defined by expression (17), the same for all predicates E, which in some sense is the simplest predicate.

By performing a 2nd-order two-layer decomposition of the predicate L, it can be represented as $L(v, w) = D_B(h_1(v), h_2(w))$. Here, the mappings h_1 and h_2 have a special form: $h_1(v) = g_1^{-1}(v)$, $h_2(w) = g_2^{-1}(w)$, where g_1: $R \to B_1$ and g_2: $R \to B_2$ are some functions, and h_1: $B_1 \to R$ and h_2: $B_2 \to R$ are mappings whose inverses are functions g_1 and g_2, respectively.

Thus, a three-layer decomposition gives a representation of the predicate E as:

$$\mathrm{E}(x, y) = D_B\big(g_1^{-1}\ \ (f_1(x)), g_2^{-1}\ \ (f_2(y))\big), \tag{20}$$

where f_1, f_2, g_1, g_2 are some functions. Rewriting formula (20) in a different way gives a more compact form of the multilayer decomposition of the predicate:

$$\mathrm{E}(x, y) = D_B(\mathrm{p}, \mathrm{q}) = \bigvee_{\sigma \in B} p^\sigma\, q^\sigma = t. \tag{21}$$

We will interpret the obtained result in technical terms. The signal transformer E (Figure 1a) is transformed into a two-layer connection of blocks f_1, f_2, L with intermediate signals v and w (Figure 1b). The signal transformer L, in turn, is transformed into a two-layer connection of blocks D_B, g_1^{-1} и g_2^{-1}. As a result, we obtain a three-layer connection of blocks D_B, g_1^{-1}, g_2^{-1}, f_1 and f_2 with intermediate signals v, w and p, q (Figure 1c). In it, the blocks that implement the functions f_1 and g_1, f_2 and g_2, are included in reverse order. Above, it was shown how the functions g_1 and g_2 are practically sought.

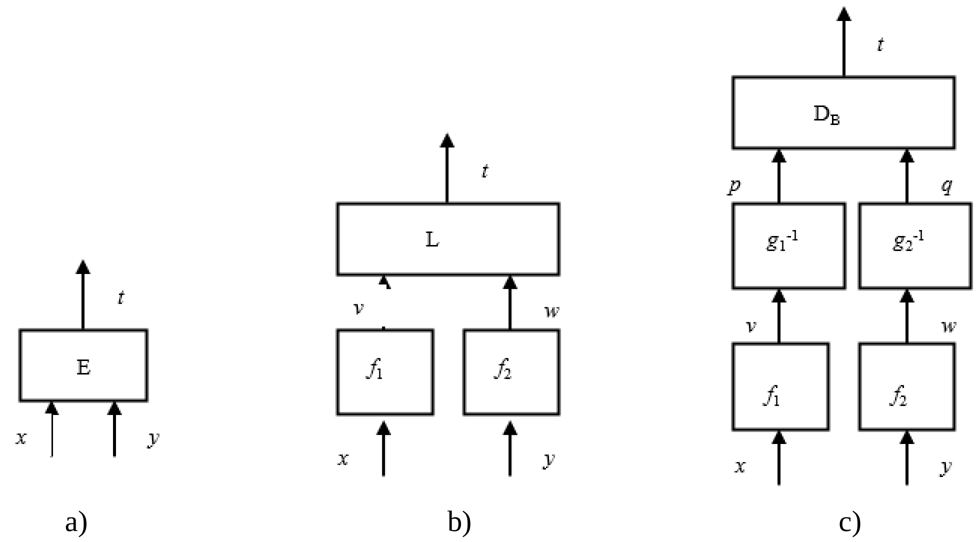

Figure 1: Signal transformation schemes

In this section, a method was developed for constructing schemes that implement arbitrary relations, and relations, as is known, represent a universal tool for modeling any objects and processes. It is important to note that the brain also implements relations, and no other neural structures have been found in the brain [1, 2]. It is natural to assume that the principle of brain operation is also based on a three-layer decomposition of predicates. The mathematical results of the work can be used in systems for automatic processing of textual information (effective support and implementation of databases, knowledge bases, expert systems, etc.), as well as in automated design of new information technologies.

6. Applying multi-layer predicate decomposition in the example of modeling linguistic relations

It is well known that predicate logic is a natural and convenient tool for modeling natural language relations. This tool satisfies all the requirements imposed on language formalizations. Moreover, all types of language processing are reduced to solving algebraic equations with different input data. Predicate logic is highly formalized and well-studied. It is designed to describe a very limited part of semantics, the one that deals with the truth or falsity of statements. Nevertheless, its elements – logical connectives, quantifiers, and especially predicates - allow for a broader range of applications.

To enable a computer to understand natural language, it is not only necessary to break down the language into its basic elements and input this information into the computer, but also to create a complete system for natural language processing [13–15].

Special importance in word inflection is played by the endings of word forms (flexional morphemes or simply flexions). In language morphology, there exists a certain dependency (relationship) between flexion and the surrounding text. The task is to mathematically describe the existing dependency, i.e. formalize the concept of flexion. The text surrounding the ending is heterogeneous with respect to it. We will distinguish between the proximate text (bordering the ending directly in the word form) and the distant text (bordering the word form). According to the principle of unambiguousness, the ending always unambiguously depends on its meaning. This principle can be interpreted as a requirement for completeness of the set of features used to select the ending. We will call the set of features complete if it ensures unambiguousness of the selection of the corresponding flexion for any feature values. A set of features that satisfies the completeness requirement will be considered meaningful.

Let us describe the mathematical formulation of the task of flexional processing of complete non-possessive adjectives in the Ukrainian language. In other words, the task is to formally describe the morphological predicate $P\ (X, Y, Z)$, which is the model of flexional processing of Ukrainian adjectives. Thus, it is necessary to form a three-letter ending $Z=z_1z_2z_3$ of the word form X depending on the set of grammatical features Y. In the Ukrainian language, there are 24 endings of complete non-possessive adjectives.

ий, им, их, ими, і, ій, ім, іх, іми, ого, ому, ої, ою, а, я, у, ю, е, є, ї, їй, їм, їх, їми.

The influence of the word X on the first, second, and third letters of the ending (z_1, z_2, z_3) can be unambiguously characterized by the set of features $X=(x_1, x_2, x_3)$, where x_1 is the feature of the last letter of the stem with values of *б, в, г, д, ж, з, к, л, м, н, п, р, с, т, у, ф, х, ц, ч, ш, щ, ь;* x_2 is the feature of the stress on the stem with values of *у* – stressed, *б* – unstressed; x_3 is the feature of softening of the stem with values of *м* – soft, *т* – hard.

Let's write the domains of the introduced variables:

$$
\begin{gathered}
z_1^{а} \vee z_1^{е} \vee z_1^{є} \vee z_1^{и} \vee z_1^{і} \vee z_1^{ї} \vee z_1^{о} \vee z_1^{у} \vee z_1^{ю} \vee z_1^{я} = 1, \\
z_2^{г} \vee z_2^{ї} \vee z_2^{й} \vee z_2^{м} \vee z_2^{х} \vee z_2^{ю} = 1, \\
z_3^{и} \vee z_3^{о} \vee z_3^{у} \vee z_3^{-} = 1, \\
x_1^{б} \vee x_1^{в} \vee x_1^{г} \vee x_1^{д} \vee x_1^{ж} \vee x_1^{з} \vee x_1^{к} \vee x_1^{л} \vee x_1^{м} \vee x_1^{н} \vee \\
\vee x_1^{п} \vee x_1^{р} \vee x_1^{с} \vee x_1^{т} \vee x_1^{ф} \vee x_1^{х} \vee x_1^{ц} \vee x_1^{ч} \vee x_1^{ш} \vee x_1^{щ} \vee x_1^{ь} = 1, \\
x_2^{у} \vee x_2^{б} = 1, \\
x_3^{м} \vee x_3^{т} = 1.
\end{gathered} \tag{22}
$$

Linguistic studies [16, 17] have shown that it is necessary to introduce four grammatical features $Y=(y_1, y_2, y_3, y_4)$, that unambiguously characterize the influence of the distant text Y on the ending Z, where y_1 is the case with values *н* – nominative, *р* – genitive, *д* – dative, *з* – accusative, *о* – instrumental, *м* – locative; y_2 is gender with values *м* – masculine, *ж* – feminine, *с* – neuter; y_3 is number with values *е* – singular, *м* – plural; y_4 is animacy with values *о* – animate, *н* – inanimate. Let us write the domains of definition for the introduced variables:

$$
\begin{gathered}
y_1^{н} \vee y_1^{р} \vee y_1^{д} \vee y_1^{з} \vee y_1^{о} \vee y_1^{м} = 1, \qquad y_2^{м} \vee y_2^{ж} \vee y_2^{с} = 1, \\
y_3^{е} \vee y_3^{м} = 1, \quad y_4^{н} \vee y_4^{о} = 1,
\end{gathered} \tag{23}
$$

The morphological predicate $P\ (X, Y, Z)$ corresponds to the function $Z=P\ (X, Y)$. In order to express this function in a compact form, we will use the first-order two-layer decomposition. Let us introduce the functions $a=\xi(x)$ and $t=\eta(y)$, and then write the desired function as $z=\varphi(a, t)$. Now let us write the function $a=\xi(x)$, but first, to avoid cluttering the notation, we introduce some symbols:

$$\text{Пригол}(x_1)= x_1^{\text{б}} \vee x_1^{\text{в}} \vee x_1^{\text{д}} \vee x_1^{\text{з}} \vee x_1^{\text{л}} \vee x_1^{\text{м}} \vee x_1^{\text{н}} \vee x_1^{\text{п}} \vee x_1^{\text{р}} \vee x_1^{\text{с}} \vee x_1^{\text{т}} \vee x_1^{\text{ф}}; \tag{24}$$
$$\text{Зяз}(x_1) = x_1^{\text{г}} \vee x_1^{\text{к}} \vee x_1^{\text{х}};$$
$$\text{Шипл}(x_1) = x_1^{\text{ж}} \vee x_1^{\text{ч}} \vee x_1^{\text{ш}} \vee x_1^{\text{щ}};$$
$$\text{Гол}(x_1)= x_1^{\text{а}} \vee x_1^{\text{е}} \vee x_1^{\text{є}} \vee x_1^{\text{и}} \vee x_1^{\text{і}} \vee x_1^{\text{о}} \vee x_1^{\text{у}} \vee x_1^{\text{ю}} \vee x_1^{\text{я}}.$$

It should be noted that in the Ukrainian language, there are no adjectives with stems that end in *-е*, *- и*, *-ї*, *-о*, *-у*, *-ю* or a soft consonant in an unstressed syllable. Taking into account the aforementioned fact and the introduced notations, the function a will be expressed as follows:

$$(x_1^{\text{ц}} \vee \text{Сог}(x_1))\, x_3^{\text{т}} \vee \text{Зяз}(x_1) \vee \text{Шипл}(x_1) \vee x_1^{\text{й}} x_2^{\text{у}} = a_1; \tag{25}$$
$$\text{Зяз}(x_1) \vee \text{Шипл}(x_1)\, x_2^{\text{б}} = a_2;$$
$$(\text{Пригол}(x_1) \vee x_1^{\text{ь}})\, x_2^{\text{у}} x_3^{\text{м}} = a_3;$$
$$(\text{Гол}(x_1) \vee x_1^{\text{й}})\, x_2^{\text{у}} = a_4;$$
$$\text{Пригол}(x_1) x_3^{\text{т}} \vee x_1^{\text{ц}} x_3^{\text{т}} x_2^{\text{б}} = a_5.$$
$$\text{Шип}(x_1)\, x_2^{\text{у}} = a_6.$$

The function $t=\eta(y)$ will have the following form:

$$(y_1^{\text{н}} \vee y_1^{\text{з}} y_4^{\text{н}})\, y_2^{\text{м}} y_3^{\text{е}} = t_1; \tag{26}$$
$$(y_1^{\text{р}} \vee y_1^{\text{з}} y_4^{\text{о}})\, y_2^{\text{м}} y_3^{\text{е}} \vee y_1^{\text{р}} y_2^{\text{с}} y_3^{\text{е}} = t_2;$$
$$y_1^{\text{д}} (y_2^{\text{м}} \vee y_2^{\text{с}})\, y_3^{\text{е}} = t_3;$$
$$y_1^{\text{о}} (y_2^{\text{м}} \vee y_2^{\text{с}})\, y_3^{\text{е}} \vee y_1^{\text{д}} y_3^{\text{м}} = t_4;$$
$$y_1^{\text{м}} (y_2^{\text{м}} \vee y_2^{\text{с}})\, y_3^{\text{е}} = t_5;$$
$$y_1^{\text{н}} y_2^{\text{ж}} y_3^{\text{е}} = t_6;$$
$$(y_1^{\text{д}} \vee y_1^{\text{о}} \vee y_1^{\text{м}})\, y_2^{\text{ж}} y_3^{\text{е}} = t_7;$$
$$y_1^{\text{з}} y_2^{\text{ж}} y_3^{\text{е}} = t_8;$$
$$y_1^{\text{о}} y_2^{\text{ж}} y_3^{\text{е}} = t_9;$$
$$(y_1^{\text{н}} \vee y_1^{\text{з}})\, y_2^{\text{с}} y_3^{\text{е}} = t_{10};$$
$$(y_1^{\text{н}} \vee y_1^{\text{з}} y_4^{\text{н}})\, y_3^{\text{м}} = t_{11};$$
$$(y_1^{\text{р}} \vee y_1^{\text{з}} y_4^{\text{о}} \vee y_1^{\text{м}})\, y_3^{\text{м}} = t_{12};$$
$$y_1^{\text{о}} y_3^{\text{м}} = t_{13};$$
$$y_1^{\text{р}} y_2^{\text{ж}} y_3^{\text{е}} = t_{14}.$$

Formulas (25) and (26) show that a word affects the endings of the full form of non-possessive adjectives in 6 different ways, and a text in 14 different ways.

As a result, we get a function $z=\varphi(a, t)$ of the following form:

$$z_1^{\text{и}} z_2^{\text{й}} z_3^{-} = a_1 t_1; \tag{27}$$
$$z_1^{\text{и}} z_2^{\text{х}} z_3^{-} = a_1 t_{12};$$
$$z_1^{\text{и}} z_2^{\text{м}} z_3^{-} = a_1 t_4;$$
$$z_1^{\text{и}} z_2^{\text{м}} z_3^{\text{и}} = a_1 t_{13};$$
$$z_1^{\text{і}} z_2^{\text{й}} z_3^{-} = (a_2 \vee a_3)(t_1 \vee t_7);$$
$$z_1^{\text{і}} z_2^{-} z_3^{-} = (a_2 \vee a_3 \vee a_6)\, t_{11};$$
$$z_1^{\text{і}} z_2^{\text{м}} z_3^{-} = (a_2 \vee a_3)(t_4 \vee t_5);$$
$$z_1^{\text{і}} z_2^{\text{х}} z_3^{-} = (a_2 \vee a_3)\, t_{12};$$
$$z_1^{\text{і}} z_2^{\text{м}} z_3^{\text{и}} = (a_2 \vee a_3)\, t_{13};$$
$$z_1^{\text{о}} z_2^{\text{г}} z_3^{\text{о}} = (a_1 \vee a_3)\, t_2;$$
$$z_1^{\text{о}} z_2^{\text{м}} z_3^{\text{у}} = (a_1 \vee a_3)(t_3 \vee t_5);$$
$$z_1^{\text{о}} z_2^{\text{ї}} z_3^{-} = (a_2 \vee a_5)\, t_{14};$$
$$z_1^{\text{о}} z_2^{\text{ю}} z_3^{-} = (a_2 \vee a_5)\, t_9;$$
$$z_1^{\text{ї}} z_2^{-} z_3^{-} = a_4 t_{11};$$
$$z_1^{\text{ї}} z_2^{\text{м}} z_3^{-} = a_4 t_4;$$
$$z_1^{\text{ї}} z_2^{\text{м}} z_3^{\text{и}} = a_4 t_{13};$$
$$z_1^{\text{ї}} z_2^{\text{х}} z_3^{-} = a_4 t_{12};$$
$$z_1^{\text{ї}} z_2^{\text{й}} z_3^{-} = a_4 t_1;$$
$$z_1^{\text{а}} z_2^{-} z_3^{-} = a_1 t_6;$$
$$z_1^{\text{у}} z_2^{-} z_3^{-} = a_1 t_8;$$
$$z_1^{\text{е}} z_2^{-} z_3^{-} = a_1 t_{10};$$
$$z_1^{\text{я}} z_2^{-} z_3^{-} = (a_3 \vee a_4)\, t_6;$$
$$z_1^{\text{ю}} z_2^{-} z_3^{-} = (a_3 \vee a_4)\, t_8;$$

$$z_1^{\epsilon} z_2^{-} z_3^{-} = (a_3 \vee a_4)\ t_{10}.$$

Thus, formulas (23) – (27) form a model of the flexion of adjectives in the Ukrainian language. It is evident that processing such a system of equations is extremely difficult and it is necessary to further decompose the original morphological predicate *P(X, Y, Z)* as described above, i.e., to perform its binarization and then exclude uninformative variable pairs from consideration. Similarly, it is possible to describe the declension of all adjectives, nouns, pronouns, and numerals, as well as the conjugation of verbs. Naturally, formalizing the concept of flexion for each part of speech presents certain problems.

The results obtained in this work can find broad applications in various areas of human activity related to computer and information technologies. The most promising direction is the creation of a new generation computer based on the principles of parallel information processing [18]. Clearly, if various regularities of natural language are described using the two-layer decomposition method of 1st and 2nd order predicates and microprocessors are built on this basis, they will be able to perform the functions of certain structures of human intelligence that participate in the implementation of corresponding aspects of human language activity.

7. Conclusions

The scientific problem of developing algebraic methods for predicate decomposition for the formal analysis of information processes, particularly for the formal representation of natural language text semantics, has been solved in this work.

A series of theorems have been considered and proven, characterizing the method of comparator identification, which is a method suitable for studying and modeling subjective states of a person. By representing the predicate in its general form of the first kind (9), we have achieved that the maximum possible amount of information carried by the predicate R about the relationships between elements of the sets A_1 and A_2 has been transferred to the functions f_1 and f_2, while the comparator L bears a minimal load. This general form of the predicate is a new model of comparator identification. The relationship between the types of isomorphisms of the model of comparator identification (equivalence predicate) and the practical features of measuring input signals has been investigated, which allowed the theory of comparator identification to be developed for a wider class of objects.

We were also able to present the predicate in its general form of the 2nd kind, which gives the researcher even more opportunities for formalizing any relations in logic algebra. The combination of the two-layer decomposition of the 1st and 2nd order made it possible to obtain a three-layer decomposition of any predicate of any dimensionality. This representation of the predicate, and therefore the relation, allows for parallel processing of information, which significantly speeds up the process of formalization and brings it closer to the workings of the human brain.

The development of formal representation methods for arbitrary relations and their subsequent schematic implementation contributes to the development of artificial intelligence systems and the improvement of the process of automated design of digital devices, which can, in particular, be part of an intelligent interface, computer-aided design and learning systems, expert systems, decision support systems, etc.

8. References

[1] Jan von Plato, The Great Formal Machinery Works: Theories of Deduction and Computation at the Origins of the Digital Age, Princeton University Press, 2017.

[2] Stanislas Dehaene, How we learn. Why brains learn better than any machine... for now, New York, 2020.

[3] Advancing Neural Machine Translation with Meta-Learning" by Xiangpeng Wei et al., published in February 2022 in the journal arXiv:2202.02527.

[4] B. Hitis, K. Yu. Hudkova, Metody shtuchnoho intelektu, Donbas. derzh. mashynobud. akad. (DDMA), 2018.

[5] K. Petrov, EH. Petrov, Komparatornaya identifikaciya modelej mnogofaktornogo ocenivaniya, Palmarium academic publishing, 2014.

[6] Bondarenko M.F., Shabanov-Kushnarenko S.YU. Teoriya cvetovogo zreniya, Khar'kov: KHNUREH, 2002.
[7] O. S. Bulhakova, V. V. Zosimov, V. O. Pozdieiev, Metody ta systemy shtuchnoho intelektu: teoriia ta praktyka, Kherson: OLDI-PLIuS, 2020
[8] O. Ye. Lytvynenko ta in. Dyskretna matematyka, Kyiv: NAU, 2017.
[9] Christopher Topalian, True Artificial Intelligence: A.I. Achieved by Utilizing a Complete Logical Boolean System, independently published, 2020.
[10] Spatial Interpretation of the Notion of Relation and Its Application in the System of Artificial Intelligence» /Ganna Proniuk, Nataliia Geseleva, Iryna Kyrychenko, Glib Tereshchenko // 3rd International Conference on Computational Linguistics and Intelligent Systems (COLINS-2019), Kharkiv, Ukraine, April 18–19, 2019, CEUR Workshop Proceedings, Vol. 2362.
[11] V. P. Semerenko, Tekhnolohii paralelnykh obchyslen, Vinnytsia: VNTU, 2018
[12] Madhu Jain, Dinesh K Sharma, Rakhee Kulshrestha, H.S. Hota, Applications of Mathematical Modeling, Machine Learning, and Intelligent Computing for Industrial Development, CRC Press, 2023.
[13] Dotsenko S. I. Liudyno-mashynnyi interfeis, Kharkiv: UkrDUZT, 2022. – 135 s.
[14] S. M. Amelina, R. O. Tarasenko, Kompiuterna leksykohrafiia i pereklad, Kyiv: Komprynt, 2018 Using natural language processing methods to improve the applicability of altmetrics in academic evaluation Yuan-Cheng Lin; Chieh Liu; Wen-Hao Chiu, 2021 IEEE International Conference on Social Sciences and Intelligent Management (SSIM).
[15] Ukrainskyi pravopys, NAN Ukrainy, In-t movoznavstva im. O.O. Potebni, In-t ukrainskoi movy, K.: Naukova dumka, 2015.
[16] Slovnyk ukrainskoi movy: u 20 t., nauk. kerivnyk proektu V. A. Shyrokov, Kyiv: Naukova dumka, 2018.
[17] Shubin, I., Snisar, S., & Litvin, S. Categorical analysis of logical networks in application to intelligent radar systems. 2020 IEEE International Conference on Problems of Infocommunications Science and Technology, PIC S&T`2020 – Proceedings, pp. 235–238. ISBN 978-172819177-5. doi:10.1109/PICST51311.2020.9467893.
[18] K. Smelyakov, A. Chupryna, D. Sandrkin and M. Kolisnyk, "Search by Image Engine for Big Data Warehouse," 2020 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream), Vilnius, Lithuania, 2020, pp. 1-4, doi: 10.1109/eStream50540.2020.9108782.