

DATA CLUSTERING BASED ON INDUCTIVE LEARNING OF NEURO-FUZZY NETWORK WITH DISTANCE HASHING

Subbotin S. A. – Dr. Sc., Professor, Head of the Department of Software Tools, National University “Zaporizhzhia Polytechnic”, Zaporizhzhia, Ukraine.

ABSTRACT

Context. Cluster analysis is widely used to analyze data of various nature and dimensions. However, the known methods of cluster analysis are characterized by low speed and are demanding on computer memory resources due to the need to calculate pairwise distances between instances in a multidimensional feature space. In addition, the results of known methods of cluster analysis are difficult for human perception and analysis with a large number of features.

Objective. The purpose of the work is to increase the speed of cluster analysis, the interpretability of the resulting partition into clusters, as well as to reduce the requirements of cluster analysis to computer memory.

Method. A method for cluster analysis of multidimensional data is proposed, which for each instance calculates its hash based on the distance to the conditional center of coordinates, uses a one-dimensional coordinate along the hash axis to determine the distances between instances, considers the resulting hash as a pseudo-output feature, breaking it into intervals, which matches the labels pseudo-classes – clusters, having received a rough crisp partition of the feature space and sample instances, automatically generates a partition of input features into fuzzy terms, determines the rules for referring instances to clusters and, as a result, forms a fuzzy inference system of the Mamdani-Zadeh classifier type, which is further trained in the form of a neuro-fuzzy network to ensure acceptable values of the clustering quality functional. This makes it possible to reduce the number of terms and features used, to evaluate their contribution to making decisions about assigning instances to clusters, to increase the speed of data cluster analysis, and to increase the interpretability of the resulting data splitting into clusters.

Results. The mathematical support for solving the problem of cluster data analysis in conditions of large data dimensions has been developed. The experiments confirmed the operability of the developed mathematical support have been carried out.

Conclusions. . The developed method and its software implementation can be recommended for use in practice in the problems of analyzing data of various nature and dimensions.

KEYWORDS: cluster analysis, neuro-fuzzy network, hash, fuzzy inference, data analysis.

ABBREVIATIONS

NFN is a neuro-fuzzy network.

NOMENCLATURE

α is a user-specified coefficient;

β is a user-specified coefficient;

ε is an error threshold value specified by the user;

$\langle a_{j,q}, b_{j,q}, c_{j,q}, d_{j,q} \rangle$ are adjustable parameters of fuzzy term membership function;

C is a set of adjustable parameters;

C^k is a k -th cluster parameters;

E is a classification error;

F is a criterion of a model quality (user-specified);

$F_{1,2}$ is a quotient from division of F for the first and second models;

$G_{1,2}$ is a quotient from division of I_G coefficients for first and second models;

$GF_{1,2}$ is a quotient from division of I_{GF} coefficients for first and second models

i is a feature number;

I_j is an indicator of individual informativity of the j -th feature;

$I_{i,p,j,q}$ is a pairwise equivalence estimate for the terms of different input features;

\bar{I}^t is an average estimate of the relationship between the terms of different input features;

\bar{I} is an average estimate of the relationship of features;

$I(i,j)$ is an estimate of the pairwise relationship of the i -th and j -th input features;

$I_{j,q}$ is an indicator of the individual informativity of the q -th term of j -th feature for the entire set of classes;

I_G is a generalization indicator;

I_{GF} is a generalization of features indicator;

j is a feature number;

K is a number of subsets (clusters, pseudo-classes);

$l_{j,q}$ is a left boundary of the q -th interval of j -th feature;

$L_{j,q}$ is a length of the q -th interval of values of the j -th feature;

N is a number of features characterizing instances;

N' is a number of features after reduction;

n is a sample dimensionality;

N_j is a number of intervals into which the range of values of the j -th feature is divided;

$N_{j,q}$ is a number of times the q -th term of j -th feature was used in the rules;

$N_{j,q \rightarrow k}$ is a number of instances of k -th class belonging to the q -th term of j -th feature;

N_w is a number of parameters of the clustering model;

opt is a formal designation of the optimum;

q^s is a number of the interval, in which s -th instance hit according to the hash value x_*^s ;

q_j^s is a number of the term of j -th feature to which s -th instance belongs (hit in);

q_j^s is a number of the interval in which s -th instance hit on j -th feature;

R_c is a conflict set of rules;
 $|R_c|$ is a power (the number of rules in) of the conflict set;
 $r_{j,q}$ is a right boundary of the q -th interval of j -th feature;
 s is an instance number;
 (s) is a rule number;
 S' is a number of instances in a reduced set;
 S is a number of instances in the sample;
 t is a running time of the method;
 $t_{1,2}$ is a quotient from division of t for the first and second methods;
 $m_{1,2}$ is a quotient from division of m for the first and second methods;
 w^s is a weight of s -th rule;
 x is a sample of observations;
 x^s is a s -th instance of a sample;
 x_j^s is a value of j -th feature for s -th instance of a sample;
 x^s is a hash value for the instance x^s ;
 y^s is a label of the output feature (pseudo-class or cluster) for s -th instance;
 y^* is a calculated class number for the recognized instance x^s .

INTRODUCTION

The cluster analysis [1–17] is widely used to analyze data of various nature and dimensions. The purpose of cluster analysis is to split the initial sample of observations (instances) into compactly located groups of instances or to identify compact areas of grouping instances – clusters in the feature space that describe the sample instances.

The **object of study** is the process of cluster analysis of data samples.

There are two groups of cluster analysis methods: crisp [1–17] and fuzzy methods [18–26]. Unlike crisp methods, which provide a coarser separation of instances, fuzzy methods allow more adaptive selection of clusters in the feature space.

The **subject of study** is the methods of fuzzy cluster data analysis.

Fuzzy cluster analysis methods [18–26] are highly adaptive and require a large number of parameters to be adjusted. Also, the well-known methods of fuzzy and crisp cluster analysis are characterized by low speed and are demanding on computer memory resources due to the need to calculate pairwise distances between instances in a multidimensional feature space. In addition, the results of known methods of cluster analysis [1–29] are difficult for human perception and analysis with a large number of features.

The **purpose of the work** is to increase the speed of cluster analysis, the interpretability of the resulting partition into clusters, as well as to reduce the requirements of cluster analysis to computer memory.

1 PROBLEM STATEMENT

Suppose given the sample of observations $x = \{x^s\}$, $s = 1, 2, \dots, S$, $x^s = \{x_j^s\}$, $j = 1, 2, \dots, N$, then the problem of cluster analysis of sample x is to determine the splitting of the sample x into K subsets (clusters, pseudo-classes) with parameters $C = \{C^k\}$:

$$x = \bigcup_{k=1}^K \{x^s \mid x^s \in C^k, s = 1, 2, \dots, S\},$$
$$F(x, C) \rightarrow opt.$$

As a rule, such a criterion should ensure the minimization of distances between instances within the same cluster and the maximization of inter-cluster distances of instances [1, 12–14]. Here, the distance between instances in the feature space is considered as a measure of their similarity.

That is, for a given sample x , we need to constructively determine F and to find the optimal (or acceptable) values of C for it.

2 REVIEW OF THE LITERATURE

According to the type of membership functions used for instances to clusters, the known methods of cluster analysis are divided into crisp and fuzzy methods.

Well-known methods of crisp cluster analysis [1–17, 27–29] assume that the initial sample of observations is divided into clusters in such a way that each instance belongs to only one cluster, and the partition is formed iteratively from the initial random or user-defined partition to the final partition that satisfies the specified criterion quality. In fact, the main differences between the well-known methods of crisp cluster analysis [1–17, 27–29] are the method of calculating the distance, the quality criterion of the partition, the method of generating the initial partition (set of initial partitions), the method of generating a new partition (set of partitions) of the sample based on the existing (or previously considered), search termination criteria. In this case, the partition quality criterion is a function determined on the basis of pairwise distances between sample instances, as well as distances from instances to cluster centers in the feature space. The calculation of such distances for samples of large dimensions is a computationally expensive task and also requires significant memory costs to load the entire sample of observations into memory. Additionally, the task is complicated by the need for pairwise enumeration of distances between instances.

The well-known methods of fuzzy cluster analysis [18–26] assume that each sample instance belongs to all clusters, but with different values of the membership function, the splitting of instances into clusters is formed iteratively from the initial random or user-specified split to the final split that satisfies given quality criteria. In fact, the main differences between the known methods of fuzzy cluster analysis, as well as for the methods of crisp cluster analysis, are the method of calculating the distance, the quality criterion of the partition, the method of

generating the initial partition (set of initial partitions), the method of generating a new partition (set of partitions) of the sample based on the existing (or previously considered), criteria for terminating the search. At the same time, in contrast to crisp cluster analysis, fuzzy methods operate with cluster memberships calculated on the basis of the distance function of instances to cluster centers. Like crisp methods in fuzzy cluster analysis, the partition quality criterion is a function determined on the basis of pairwise distances between sample instances in the feature space. The calculation of such distances for samples of large dimensions is a computationally expensive task and also requires significant memory costs to load the entire sample of observations into memory. Additionally, the task is complicated by the need for pairwise enumeration of distances between instances, as well as the need to recalculate the belonging of instances to clusters.

Crisp methods of cluster analysis [1–17, 27–29] are obviously more accurate (provide a specific result), but at the same time coarse and less adaptive. Fuzzy methods [18–26] give a fuzzy assessment of the membership of an instance to a cluster and are less accurate (specific in the assessment of membership), but at the same time they are more adaptive compared to crisp methods, but also more expensive in terms of the amount of calculations and the required memory.

Depending on the method of forming a partition into clusters, cluster analysis methods can be divided into non-hierarchical [1–17], in which clusters are not subordinated, and hierarchical [27–29], in which partitioning is carried out sequentially by forming nested clusters. In fact, non-hierarchical methods implement breadth-first search, while hierarchical methods implement depth-first search.

It should also be noted that most of the known methods of cluster analysis are dependent on a set of features specified by the user and do not allow one to evaluate their significance. This leads to an excessive partition, an increase in the number of calculations, and also reduces the possibility for the perception of the resulting partition by a person.

Also, if a set of features contains interrelated, duplicate, similar features, or features that are discrete of a time-distributed value, traditional cluster analysis methods will generate an extremely complex, redundant, and uninterpretable partition. However, they will not be able to identify such signs and eliminate or use them more effectively.

Therefore, there is a need to eliminate the shortcomings of crisp and fuzzy methods by developing a fuzzy clustering method taking into account crisp partitioning and search acceleration heuristics.

3 MATERIALS AND METHODS

Unlike most cluster analysis methods [1–29], which involve calculating the distances between all instances in the feature space, it is proposed to calculate the hash distance from it to the conditional common center of coordinates for each instance, replacing the N -dimensional in-

stance coordinate vector with one coordinate, and then determine the distance between instances in one-dimensional space. This will allow for large samples to load into memory only individual instances (minimally – one by one in turn), reducing the amount of calculations and the minimum amount of memory required.

Also, unlike the traditional methods of cluster analysis [1–29], it is proposed to consider the obtained hash feature [30–49] as a pseudo-output feature, dividing it into intervals, which can be compared with labels of pseudo-classes – clusters. This will allow replacing the enumeration of pairs of compared feature distances with an ordered set of one-dimensional coordinates of instances along the hash axis, thus reducing the amount of calculations.

Further, having received a rough crisp splitting of the sample instances, it is proposed for them to set the splitting of the input features into fuzzy terms, to determine on their basis and splitting the instances the rules for referring instances to clusters.

In contrast to the traditional metric approach to cluster analysis [1–17], which involves the use of the entire set of initial features, it is proposed to evaluate the informativity of features [50, 51] and fuzzy terms [52, 53] and exclude non-informative terms, as well as non-informative features, while maintaining an acceptable level of quality criterion.

Then we may determine the fuzzy inference system of the Mamdani-Zadeh classifier type, which in the form of a neuro-fuzzy network can be further trained by means of optimization methods [54–56] to adjust the parameters of membership functions to fuzzy terms and weights of rules that provide acceptable values of the clustering quality function.

The above mentioned feature and term reduction will reduce the complexity of calculations, reduce the amount of memory, the complexity of the neuro-fuzzy system, reduce the number of configurable network parameters and, as a result, increase its level of data generalization, as well as interpretability.

Formally, a method for constructing a NFN for data cluster analysis that implements the ideas described above can be represented as follows.

The initialization stage. Specify a sample of observations $x = \{x^s\}$ and used defined values $\varepsilon \geq 0$, $0 < \alpha \leq 1$, and $0 < \beta \leq 1$.

The hash calculation stage. Using one of the hash calculation methods [30–49] determine the hashes $\{x_*^s\}$ for the sample instances. Order sample instances along the hash value axis x_*^s .

The stage of a crisp division of the feature space. Split the range of hash values x_*^s into intervals, numbering them sequentially. For each sample instance x^s , fix the number of the interval q_*^s , in which it fell according to the hash value x_*^s , as the label of the output feature – the pseudo-class (cluster) $y^s = q_*^s$.

A crisp division of the ranges of feature values into intervals (selection of terms) can be done in various ways.

The simplest way is to divide the range of values of each feature into an equal number of intervals that have the same length for the corresponding feature. With such a partition, the space of input features will be divided by a uniform grid, the parameters of the intervals of which are easy to determine the number of intervals into which the range of values of the j -th feature is divided N_j ($N_j \geq 2$). It is also reasonable to provide the restriction $N_j \geq K$. It is also desirable that $N_j \ll S$. For a given number of intervals N_j define $L_{j,q}$ the length of the q -th interval of values of the j -th feature:

$$L_{j,q} = \frac{x_j^{\max} - x_j^{\min}}{N_j},$$

$$x_j^{\min} = \min_{s=1,2,\dots,S} \{x_j^s\},$$

$$x_j^{\max} = \max_{s=1,2,\dots,S} \{x_j^s\},$$

on the basis of which we calculate the left $l_{j,q}$ and right $r_{j,q}$ boundaries of the intervals:

$$l_{j,q} = x_j^{\min} + (q-1)L_{j,q},$$

$$r_{j,q} = x_j^{\min} + qL_{j,q}.$$

For the s -th instance, the number of the interval q , in which it falls according to the j -th feature, we determine as:

$$q_j^s = 1 + \left\lfloor \frac{x_j^s - x_j^{\min}}{L_{j,1}} \right\rfloor$$

or as

$$q_j^s = \{q | l_{j,q} \leq x_j^s \leq r_{j,q}, q = 1, 2, \dots, N_j\}.$$

Since the dimension of the sample is $n = NS$, then to ensure the generalizing properties of the model, it is important that $3N_jN \leq n$, that is, $3N_j \leq S$. As a result, we can recommend setting: $K \leq N_j \leq S/3$. If $S/3$ is less than K , then set $N_j = K$.

In the cells of such a grid, in the general case, instances of different pseudo-classes will fall, since the hash feature is not taken into account. Also unknown is the number of intervals into which it is necessary to divide the ranges of feature values in order to achieve an acceptable accuracy of approximation of the cluster boundaries. This partition will be computationally the fastest and simplest, but it will contain an uncertainty in the choice of the number of intervals, and it will also not allow us to accurately select clusters.

It is more difficult to divide the range of feature values into intervals, when a different number of intervals are allocated on the axis of each feature and the pseudo-class

number determined by the hash feature [30–49] is taken into account. To do this, we need to project labels of instances (pseudo-class numbers) one by one onto the axis of the j -th feature in ascending order of its values.

In this case, a situation may arise when for the same value of the coordinate along the axis of the j -th feature, there are several instances with different labels of pseudo-classes. In this case, instances with a label equal to the label of the instance with a lower coordinate preceding the group should be placed first, and instances with a label equal to the label of the instance with a larger coordinate following the group should be placed last. After that, it is necessary to select intervals of values of the j -th feature $\{<l_{j,q}, r_{j,q}>\}$ such that within one interval there are instances with the same value of the hash pseudoclass number, and instances of adjacent intervals of feature values have different hash pseudoclass numbers. Here the situations are possible when adjacent intervals can touch and partially overlap if the left and right boundaries of adjacent intervals have the same coordinates. It is also possible that there will be voids between adjacent intervals in which there are no instances.

After the formation of such a partition, the number of intervals into which the range of values of the feature is divided N_j can be used to determine the information content of the features.

The more intervals of changing the class number the range of the feature is divided into, the more complex and non-linear the classification is, i.e. the lower the individual informativity of this feature. The fewer intervals of feature values (ideally, one) correspond to a specific pseudo-class number, the more valuable this interval is for this pseudo-class. This can be quantified by the indicator of individual informativity of the j -th feature:

$$I_j = \frac{K}{N_j}, N_j \geq K.$$

This indicator will tend to zero with a lower individual information content of the feature and to one – with a higher one.

The rule formation stage. Convert each instance of the sample into a crisp rule of the form:

$$(s) : \text{if } \bigcup_{j=1}^N \left\{ \bigcap_{q=1}^{N_j} x_j^s \in [l_{j,q}, r_{j,q}] \right\},$$

then $y^s = q^s$ with a weight $w^s=1$.

Here (s) is a rule number.

To simplify program processing, such rules can be represented as a set $R: \{(s) : \{q_j^s\} \rightarrow q^s, w^s\}$.

The stage of assessing the quality of a partition and a set of rules. To assess the quality of the generated partition and set of rules, it is possible to use the classification error E .

To do this, for each s -th instance of the sample x^s , $s=1, 2, \dots, S$:

- determine its belonging to each term $\{q_j^s\}$;
- from the set of rules R select those rules that correspond to the recognized instance on the left side (form a conflict set of rules R_c and estimate its power (the number of rules in the conflict set $|R_c|$));
- define as the calculated class number for the recognized instance x^s and the conflict set R_c the value:

$$y_*^s = \arg \max_{q=1,2,\dots,K} \left\{ \sum_{p=1}^{|R_c|} w^p \right\}.$$

For a sample of recognized instances, the classification error can be estimated as:

$$E = \sum_{s=1}^S \{1 | y^s \neq y_*^s\}.$$

If the error value is unacceptable ($E > \varepsilon$), then it is possible to revise the generated partition by increasing the number of intervals into which the feature value ranges are divided and / or change the hash calculation method.

The rule reduction stage. All rules should be sorted by the value $y=\{q^s\}$, then by the values $\{q_j^s\}$. Set $S' = S$. Looking through the rules sequentially $s=1, 2, \dots, S'-1$: for two rules (s) and ($s+1$) consecutive in y , if their right parts are the same ($q^s = q^{s+1}$) and the left parts are the same ($\forall j=1,2,\dots,N : q_j^s = q_j^{s+1}$), then keep the first (s -th) rule, increasing its weight by the weight of the ($s+1$ -th) rule: $w^s = w^s + w^{s+1}$, then remove the second ($(s+1)$ -th) rule, and decrease S' : $S' = S' - 1$.

The stage of reduction of terms and features. For each term of each input feature, using the set of generated rules for each k -th pseudo-class, determine the number of times the term was used in the rules, taking into account their weights:

$$N_{j,q,k} = \sum_{s=1}^{S'} \{w^s | q_j^s = q, q_*^s = k\},$$

$$N_{j,q} = \sum_{s=1}^{S'} \{w^s | q_j^s = q\}.$$

The greater the value of $N_{j,q,k}$, the more strongly the q -th term of the j -th feature is involved in making decisions about assigning an instance to the k -th pseudoclass.

Let's define the indicator of the individual informativity of the term for the entire set of classes:

$$I_{j,q} = \frac{\max_{k=1,2,\dots,K} \{N_{k,j,q}\}}{N_j}.$$

This indicator will take values in the range from zero to one. The smaller its value, the less informative is the q -th term of the j -th feature. The greater its value, the more significant is the q -th term of the j -th feature.

In ascending order, the estimates of the individual informativities of features I_j are sequentially for the current considered j -th feature:

- exclude it (all its terms) from all rules;
- estimate the error in determining the class number for sample instances according to the current set of rules and terms E ;
- if the error E is acceptable ($E \leq \varepsilon$), then consider the j -th feature as non-informative and remove it from further consideration, and also remove its terms and their membership functions;
- if the error E is unacceptable ($E > \varepsilon$), then return the deleted feature and terms to the rules and stop further revision of the features.

Looking through the terms of the features in order from the least frequently used in the rules to the most frequently used (i.e. in ascending order of the value of $I_{j,q}$), sequentially for each term:

- exclude it from all rules;
- estimate the error in determining the class number for sample instances according to the current set of rules and terms E ;
- if the error is acceptable ($E \leq \varepsilon$), then consider the q -th term of the j -th feature as non-informative and remove it and its membership function from further consideration.
- if the error is unacceptable ($E > \varepsilon$), then return the deleted term to the rules and stop further revision of the terms.

The stage of identifying similarities and reducing similar features. For $i=1, 2, \dots, N, j=i+1, i+2, \dots, N$, determine the estimate of the pairwise relationship of the i -th and j -th input features $I(i,j) = I(j,i)$. It is possible to implement this on the basis of indicators of individual informativity of features [48–51], meaning the input feature is the i -th feature, and the output feature is the j -th feature.

Based on pairwise estimates $\{I(i,j)\}$, determine the average estimate of the relationship of features:

$$\bar{I} = \frac{1}{0.5N(N-1)} \sum_{i=1}^N \sum_{j=i+1}^N I(i,j).$$

Divide features into groups, such that the features of one group have an estimate of the relationship greater than the average, multiplied by a user-specified coefficient α , $0 < \alpha \leq 1$. To do this, first enter into the set of unconsidered features all the features that are present in the rules of the current set of rules. Then, while the set of unconsidered features is non-empty, repeat:

- choose from the set of individually unconsidered the most informative feature (in relation to the output feature) and form a new group of features for it;
- from the set of unconsidered features, select all the features that are related with the feature of the new group stronger than the average relation, taking into account the coefficient α : $I(i,j) \geq \alpha \bar{I}$, transfer them to the group of this feature.

From the features of each group containing two or more features:

- leave in the rules only one feature that is most closely related to the output feature (has the highest estimate of individual informativity I_j).
- estimate the classification error E for the current set of rules and terms;
- if the error is acceptable ($E \leq \varepsilon$), then remove the excluded features and their terms from further consideration, otherwise, return all the features of this group.

An alternative option is to sequentially remove from each group individually the least significant feature (with a lower value of I_j) until the error remains acceptable and the number of remaining features in the group is at least one.

Also, in a number of problems where it is assumed that a number of input features are indirect observations of a hidden factor or some of the input features are discrete samples of a distributed signal, then instead of or in addition to this stage, it is possible to include not the removal, but the combination of the primary features into the calculated one. In this case, after selecting groups of features for each group, based on all its features, it is necessary to calculate the values of artificial convolution features (in the simplest case, this can be the sum, average value, maximum, minimum, and the product of the values of the primary features of the group), and then evaluate for each convolution its connection with the output feature. If not a single convolution exceeds the individual informativity of the primary features of the group in terms of the value of individual informativity, then for this group it should be limited to choosing one primary feature with the highest individual informativity, otherwise all primary features should be excluded from the group, replacing them into an artificial convolution feature, determine the terms and their parameters for this feature, adjusting the generated partition and its parameters, as well as the set of rules.

The stage of revealing the similarity and reduction of similar terms of different features. Determine the pairwise equivalence estimate for the terms of different input features:

$$I_{i,p,j,q} = I_{j,q,i,p} = \frac{\sum_{s=1}^S \{1 | l_{i,p} \leq x_i^s \leq r_{i,p}, l_{j,q} \leq x_j^s \leq r_{j,q}\}}{\max\{N_{i,p}, N_{j,q}\}},$$

$$i = 1, 2, \dots, N, j = i + 1, i + 2, \dots, N,$$

$$p = 1, 2, \dots, N_i, q = 1, 2, \dots, N_j.$$

Determine the average estimate of the relationship between the terms of various input features:

$$\bar{I}^t = \frac{\sum_{i=1}^N \sum_{j=i+1}^N \sum_{p=1}^{N_i} \sum_{q=1}^{N_j} I_{i,p,j,q}}{0,5N(N-1) \left(\sum_{i=1}^N N_i \right)^2}.$$

Divide the terms into groups, such that the terms of different features of the same group have an estimate of the relationship greater than the average multiplied by a user-specified coefficient β , $0 < \beta \leq 1$.

To do this, first enter into the set of unconsidered terms all the terms that are present in the rules of the current set of rules. Then, while the set of unconsidered terms is non-empty, repeat:

- choose from the set of individually unconsidered the most informative term and form a new group of terms for it;
- from the set of unconsidered terms excluding other terms of the feature same as feature of a group forming term, select all the terms that are related with the term of the new group stronger than the average relation, taking into account the coefficient β : $I_{i,p,j,q} \geq \beta \bar{I}^t$, transfer them to the group of this term.

From the terms of each group containing two or more terms:

- leave in the rules only one term, the feature of which is most closely related to the output feature (it has the biggest value of individual informativity I_j);
- estimate the classification error E for the current set of rules and terms;
- if the error is acceptable ($E \leq \varepsilon$), then remove the excluded terms from further consideration, otherwise, return all the terms of this group.

An alternative option is to sequentially remove from each group individually the least significant term (with a lower value of I_j) until the error remains acceptable and the number of remaining terms in the group is at least one.

The stage of fuzzy terms formation. On the basis of the parameters of the intervals of the values of the features selected during the formation of a crisp partition and the terms and features selected in the process of reduction, it is possible to determine the membership functions for fuzzy terms. For this it is possible to use different types of elementary membership functions [21, 22]. For crisp intervals in which two or more instances fell into, it is proposed to use the following functions: trapezoidal, bell-shaped, Gaussian, Π -shaped. For point intervals, where only one instance fell, it is proposed to use the following functions: triangular, bell-shaped, Gaussian, Π -shaped function. Each of these functions $\mu_{j,q}(x_j^s)$ for a specific fuzzy term (the q -th interval of values on the axis of the j -th feature) will have adjustable parameters $\langle a_{j,q}, b_{j,q}, c_{j,q}, d_{j,q} \rangle$, such that: $a_{j,q} \leq b_{j,q} \leq c_{j,q} \leq d_{j,q}$. The values of the parameters of the membership functions can be determined based on the parameters of a crisp partition.

For example, for trapezoidal and Π -shaped functions, the parameters can be defined as:

- for splitting into intervals equal in length:

$$a_{j,q} = b_{j,q} = l_{j,q},$$

$$c_{j,q} = d_{j,q} = r_{j,q};$$

– for splitting into intervals of different classes:

$$a_{j,q} = \begin{cases} l_{j,q}, q = 1; \\ \frac{r_{j,q-1} + l_{j,q}}{2}, q > 1; \end{cases}$$

$$b_{j,q} = l_{j,q},$$

$$c_{j,q} = r_{j,q},$$

$$d_{j,q} = d_{j,q} = \begin{cases} r_{j,q}, q = N_j; \\ \frac{r_{j,q} + l_{j,q+1}}{2}, q < N_j. \end{cases}$$

The stage of NFN formation for clustering. Map the generated knowledge base into a fuzzy logical inference system, which is conveniently represented in the neural network basis as a NFN.

The network structure can be determined based on the Mamdani-Zadeh approximator [57]. The nodes of the input layer of the network will correspond to the input features x_j for the recognized instance $x^s = \{x_j^s\}$. Thus,

the input layer will have N nodes (hereinafter, we mean not the initial values of the number of features and terms, but after reduction).

The nodes of the first hidden layer of the network will correspond to fuzzification blocks, i.e. will determine the values of membership functions for terms of input features $\mu_{j,q}(x_j^s)$. On the first hidden layer there will be

$\sum_{j=1}^N N_j$ nodes. The input of each node of the first hidden

layer receives a value from the output of only the input layer node corresponding to its feature.

The nodes of the subsequent second hidden layer will combine the membership functions of terms into the membership functions of the antecedents (left parts) of the rules, combining the outputs of the nodes of the first layer, the corresponding terms of which are included in the corresponding antecedents. The second hidden layer will have S' fuzzy “AND” nodes.

The nodes of the third layer will combine the rules into pseudo-classes, implementing a fuzzy “OR”. The third hidden layer will have K nodes.

The single node of the output layer will defuzzify the result, giving the number of the cluster (pseudo-class) according to the formula:

$$y^s = \left[\frac{\sum_{k=1}^K k \mu_k(x^s)}{\sum_{k=1}^K \mu_k(x^s)} \right] \text{ or}$$

$$y^s = \arg \max_{k=1,2,\dots,K} \{\mu_k(x^s)\}.$$

The network parameters will be determined on the basis of the previously formed crisp partition and set of rules.

The stage of additional training and optimization of the neuro-fuzzy clusterizer. Let evaluate the performance quality of a NFN (the quality of data clustering) based on a given functional F , which can be determined based on a wide class of metrics [1, 2, 7, 8, 12–15]. Using the methods of evolutionary optimization [54, 55], we can select such values of the network term parameters that will improve the value of the optimized functional F . The final model will be a neuro-fuzzy clusterizer optimized by the number of features used and the functional F .

4 EXPERIMENTS

To study the practical applicability of the proposed method, it was implemented in software and used to solve a set of practical problems of different nature and dimension. The characteristics of the initial data samples for practical tasks are given in Table 1.

Table 1 – Characteristics of initial samples for cluster analysis

Task name	Task acronym	Source	N	S	n
Low Resolution Spectrometer	LRS	https://archive.ics.uci.edu/ml/datasets/Low+Resolution+Spectrometer	102	531	54162
Musk (Version 2)	Mv2	https://archive.ics.uci.edu/ml/datasets/Musk+%28Version+2%29	168	6598	1108464
Urban Land Cover	ULC	https://archive.ics.uci.edu/ml/datasets/Urban+Land+Cover	148	168	24864
Iris	IRIS	https://archive.ics.uci.edu/ml/datasets/Iris	4	150	600
Heart Disease	HD	https://archive.ics.uci.edu/ml/datasets/Heart+Disease	75	303	22725
Breast Cancer Wisconsin (Diagnostic)	BCWD	https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29	32	569	18208
Arrhythmia	ART	https://archive.ics.uci.edu/ml/datasets/Arrhythmia	279	452	126108
Crop mapping using fused optical-radar	CMFOR	https://archive.ics.uci.edu/ml/datasets/Crop+mapping+using+fused+optical-radar+data+set	175	325834	57020950
Sensorless Drive Diagnosis	SDD	https://archive.ics.uci.edu/ml/datasets/Dataset+for+Sensorless+Drive+Diagnosis	49	58509	2866941

For each task in the experiments, various hash generation methods and various values of the parameters that regulate the operation of the proposed method were used.

To evaluate the results of the cluster analysis methods, we used the error E , the F value for the resulting model, the

running time of the method t , the memory size required by the method m , the number of parameters of the clustering model N_w , and the number of features after reduction N' . It is also suggested that, in addition to the characteristics described above, the following metrics to be used.

The generalizing properties of the resulting clustering models compared to the original data dimension can be characterized by the generalization coefficient:

$$I_G = \frac{NS}{N_w} = \frac{n}{N_w}, N_w \geq 1.$$

This coefficient will take a value in the range from zero to n . The more parameters the model has, the lower the level of generalization it has relative to the dimension of the initial data, the lower the value of the generalization coefficient.

Alternatively, the generalization may be characterized at the equal number of instances as the ratio of the number of features in the primary set N to the number of features used in the reduced set of the final model, N' :

$$I_{GF} = \frac{N}{N'}, N \geq N' \geq 1.$$

This coefficient will take a value in the range from zero to n . The more parameters the model has, the lower the level of generalization it has relative to the dimension of the initial data, the lower the value of the generalization coefficient.

When comparing pairwise the resulting models 1 and 2 for the same initial data sample, their generalization with acceptable values of the criterion F may be characterized by the relations:

$$G_{1,2} = \frac{I_{G1}}{I_{G2}} = \frac{NS}{N_{w1}} \frac{N_{w2}}{NS} = \frac{N_{w2}}{N_{w1}}, N_{w1} \geq 1, N_{w2} \geq 1, N \geq 1, S \geq 1.$$

$$GF_{1,2} = \frac{I_{GF1}}{I_{GF2}} = \frac{N}{N'_1} \frac{N'_2}{N} = \frac{N'_2}{N'_1}, N \geq 1, N'_1 \geq 1, N'_2 \geq 1.$$

For two models we also can define:

$$t_{1,2} = t_1 / t_2,$$

$$m_{1,2} = m_1 / m_2,$$

$$F_{1,2} = F_1 / F_2.$$

5 RESULTS

Table 2 presents the values of the indicators $G_{1,2}$, $GF_{1,2}$, $t_{1,2}$, $m_{1,2}$, and $F_{1,2}$ to compare the proposed method with the fuzzy c-means method [24–26], in which the initial partition is formed randomly.

Table 2 – Resulting values of indicators for clustering model comparison

Task acronym	$GF_{1,2}$	$G_{1,2}$	$t_{1,2}$	$m_{1,2}$	$F_{1,2}$
LRS	1.3	1.5	0.8	0.7	0.9
Mv2	1.4	2.4	0.9	0.4	1.1
ULC	1.5	1.7	0.9	0.6	0.9
IRIS	1.3	1.4	1	0.8	1
HD	1.2	1.5	0.9	0.6	1.1
BCWD	1.1	1.7	1.1	0.6	0.9
ART	1.3	2.2	0.9	0.5	0.9
CMFOR	1.2	1.9	0.8	0.5	1.1
SDD	1.2	1.7	0.9	0.6	0.9

It is easy to see from the Table 1 and Table 2, that the proposed method allows for the same data sample to significantly improve the generalizing properties of the model, to reduce time and computer memory costs, and also provide a better or acceptable value of the quality functional. This is explained by the fact that the proposed method non-randomly generates a partition of the feature space, selects and reduces non-informative terms and features, seeking to reduce the complexity of the model. At the same time, the proposed method does not require the calculation of distances between instances due to the use of a locally sensitive hash.

The generalized dependences between key characteristics of the proposed method obtained in experiments are schematically shown in Fig. 1–Fig. 9.

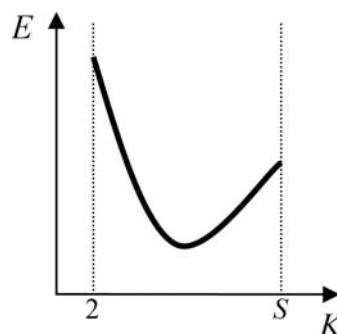


Figure 1 – Schematic graph of the averaged dependence E from K

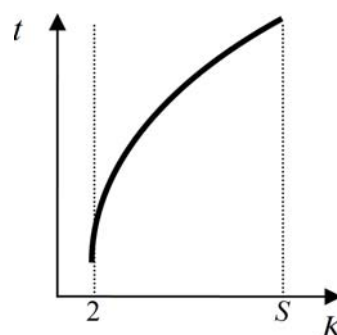


Figure 2 – Schematic graph of the averaged dependence t from K

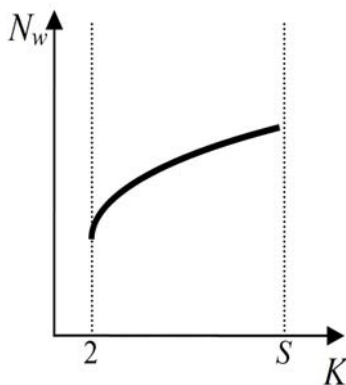


Figure 3 – Schematic graph of the averaged dependence N_w from K

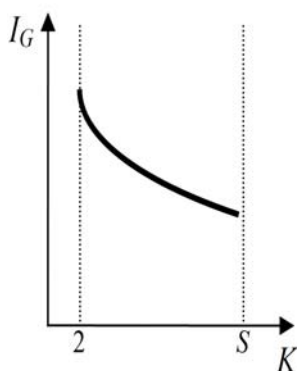


Figure 4 – Schematic graph of the averaged dependence I_G from K

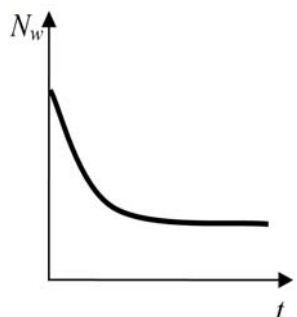


Figure 5 – Schematic graph of the averaged dependence N_w from t

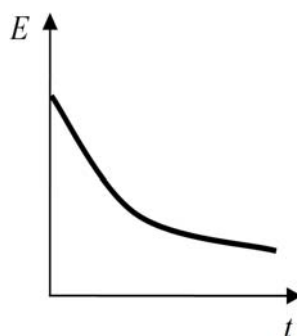


Figure 6 – Schematic graph of the averaged dependence E from t

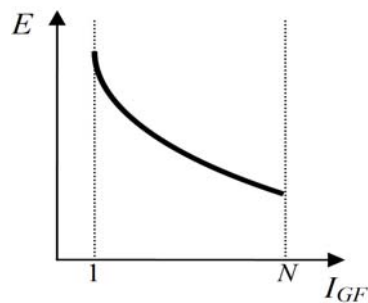


Figure 7 – Schematic graph of the averaged dependence E from I_{GF}

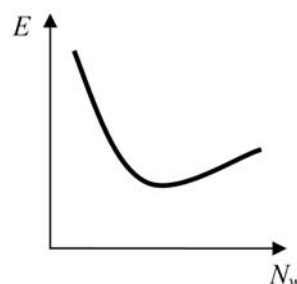


Figure 8 – Schematic graph of the averaged dependence E from N_w

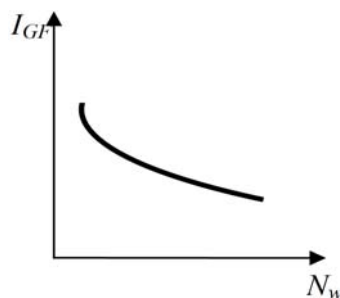


Figure 9 – Schematic graph of the averaged dependence I_{GF} from N_w

As can be seen from the Fig. 1, with an increase in the value of K , the value of the error E decreases to a certain level, after which it begins to grow. The decrease in the error E with increasing K is explained by the increase in the accuracy of approximating the boundaries of clusters due to the refinement of the partition of the output feature. An increase in the error E with a further increase in the value of K is explained by an increase in the uncertainty in the approximation of the boundaries of clusters with an excessively detailed partition of the output feature due to the selection of a large number of small clusters. Therefore, it is recommended to iteratively select such a value of K in the process of executing the method, when using a uniform partition of the feature space, at which the smallest error value will be achieved. The use of non-uniform partitioning into intervals with different class numbers makes it possible to automate this process and avoid the growth of the error. Generally, with an increase in the number of clusters, the accuracy of the partition increases,

but the cost of calculations and memory requirements also increase.

As can be seen from the Fig. 2 and Fig. 3, with an increase in the number of classes K , the running time of the method t and the number of parameters of the resulting model N_w are increased significantly. This is explained by the fact that with an increase in the number of pseudo-classes (pseudo-clusters) K , the number of selected terms and their parameters will increase, which will require a significant amount of time to calculate their information content indicators and reduction, as well as optimization tuning of the increased number parameters of the NFN model.

As can be seen from the Fig. 4, with an increase in the number of pseudo-clusters K , a decrease in the generalization indicator of the model is observed. This is explained by a significant increase in the number of model parameters due to a more detailed approximation of the cluster boundaries due to an increase in their number.

As can be seen from the Fig. 5, with an increase in the time spent in the process of the method's work t , the number of model parameters N_w is reduced. This is explained by the fact that the number of features and terms is reduced due to the removal of non-informative and duplicate features and terms. The more iterations will be in the process of identifying and reducing non-informative terms and features, the more time will be spent by the method, but the smaller will be the number of parameters of the resulting model.

As can be seen from the Fig. 6, with an increase in the time of the method t , a decrease in the model error E is observed. This is explained by the fact that the adjustment of the model parameters makes it possible to increase the accuracy (reduce the error) of the model. Also, the increase in time costs can be explained by an iterative search for the optimal partition of the feature space, which will eventually lead to a decrease in the error of the resulting model.

As can be seen from the Fig. 7, with an increase in the value of the I_{GF} indicator, a decrease in the error of the model E is observed. This is explained by the fact that with a very high generalization, the number of features used will be less and, accordingly, the approximation of the partition of the feature space will be rougher, which will lead to an increase in the error E . With the lowest value of the feature generalization index, more features will be used and the feature space will be split in more detail, which will reduce the error value.

As can be seen from the Fig. 8, with an increase in the number of model parameters N_w , there is a drop in the error value to a certain value. This is explained by the fact that the detailing of the division of the feature space due to the increase in the number of pseudoclusters makes it possible to more accurately approximate the boundaries of the clusters. The further growth of the error in the process of increasing the value of N_w is explained by the fact that the excessive selection of clusters leads to a lack of generalization, which is gradually reflected in the

growth of the error value E . However, this is typical mainly for the uniform partition of the feature space.

As can be seen from the Fig. 9, with an increase in the number of model parameters N_w , a decrease in the generalization index of features I_{GF} is observed. This is explained by the fact that the detailing of the division of the feature space leads to the forming of a larger number of non-informative terms and features, which makes it possible to exclude non-informative features. On the other hand, an increase in the number of model parameters N_w can be explained by an increase in the number of features in the original feature set for the task N , which in turn may indicate a greater proportion of non-informative features, i.e. about the reduction in the number of features.

At the Fig. 10 the schematic graphs of averaged dependencies of E , I_G , t , and N_w from the α and β values are shown.

As it can be seen from the Fig. 10 the bigger the value of α or β the bigger will be values of t and N_w and the lower the E and I_G values. If it is assumed that the features are of a different nature, then the value of the coefficients α and β is recommended to be set the bigger. If it is assumed that the features are ordered readings of a certain value, then the values of the coefficients α and β are recommended to be set smaller.

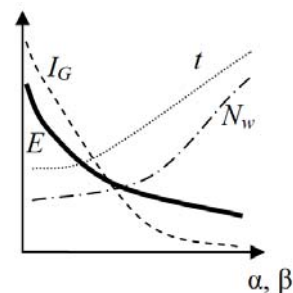


Figure 9 – Schematic graphs of the averaged dependences E , I_G , t , and N_w from the α and β values

6 DISCUSSION

The operability and practical applicability of the proposed method and the developed software were confirmed as a result of the analysis of experimentally obtained data.

The proposed method combines the ideas of crisp and fuzzy cluster analysis. At first, it forms a crisp partition of the feature space, but then, due to fuzzification, it transforms it into a fuzzy partition. The crisp partition is used to automate the selection of the number of clusters, as well as to speed up the selection of feature terms. The proposed method generates a crisp partition automatically (without human intervention), and the proposed method is faster than the traditional iterative (optimization) formation of a fuzzy partition [13, 21–23], which is inherent in most fuzzy clustering methods.

Unlike the traditionally used metric methods of cluster analysis [1–3], which involve the use of the entire primary set of features in the final model, the proposed method

selects the minimum subset of features necessary for clustering, thereby reducing the structural and parametric complexity of the model, increasing its generalization properties and interpretability (explainability), and can also reduce the number of features by combining primary features into artificial calculated ones, thereby further increasing the generalizing properties of the model and its interpretability. In addition, the proposed method is more adaptive due to fuzzification, does not require the initial setting of the number of clusters and the initial splitting of the sample into clusters, as well as the user setting metrics for clusters.

Unlike hierarchical methods of cluster analysis [27–29], which subordinate features and form a splitting hierarchy, which can have a large depth, the proposed method does not subordinate features, but at the same time, removes non-informative features and terms, and the hierarchy of its depth checks does not exceed three levels. At the same time, the proposed method significantly exceeds hierarchical methods in terms of parallelization of calculations, which is achieved due to a smaller depth compared to hierarchical methods. However, the proposed method makes it possible to obtain as an additional result the estimates of the informativity of features and terms, to form artificial features by replacing the original ones, to adaptively adjust the shape and parameters of clusters due to membership functions, and also to automatically generate the number of clusters.

CONCLUSIONS

The problem of multidimensional data cluster analysis is considered. Within this problem the cluster formation speed is increased, the complexity of the clustering model is reduced, and its interpretability is increased.

The scientific novelty of obtained results is that for the first time a method of cluster analysis of multidimensional data is proposed, which for each instance calculates its hash based on the distance to the conditional center of coordinates, uses a one-dimensional coordinate along the hash axis to determine the distances between instances, considers the resulting hash as a pseudo output a feature, dividing it into intervals, to which it compares labels of pseudo-classes-clusters, having received a rough crisp partition of the feature space and sample instances, automatically generates a partition of input features into fuzzy terms, determines the rules for referring instances to clusters and, as a result, forms a fuzzy inference system of the Mamdani-Zadeh classifier, which is retrained in the form of a neuro-fuzzy network to ensure acceptable values of the clustering quality functional. This makes possible to reduce the number of terms and features used, to evaluate their contribution to making decisions about assigning instances to clusters, to increase the speed of data cluster analysis, and to increase the interpretability of the resulting data splitting into clusters.

The practical significance of obtained results is that mathematical support allowing to solve the problem of cluster data analysis in conditions of large data dimensionality has been developed. The experiments confirmed

the operability of the developed software have been carried out. They allow to recommend it for use in practice in problems of data analysis of various nature and dimensions.

The prospects for further research are to study the application of the proposed method on a wide range of practical problems of various dimensions and nature, to study the influence of various metrics on the results of the method (accuracy and speed of building NFN, computational complexity), to develop a parallel implementation of the method, to study questions of method integration with evolutionary and multi-agent search methods.

ACKNOWLEDGEMENTS

The work is supported by the state budget scientific research project “Intellectual methods for diagnosing and prediction the state of the complex objects” (state registration number 0122U000972) of the National University “Zaporizhzhia Polytechnic” with partial information support of international projects “Cross-domain competences for healthy and safe work in the 21st century” (WORK4CE, Project Reference: 619034-EPP-1-2020-1-UA-EPPKA2-CBHE-JP) of the Erasmus+ program of the European Union and “EuroPIM Virtual Master School Ukraine” (EU-ViMUK) of the program “Ukraine digital: Ensuring study success in times of crisis (2022)” of the German Academic Exchange Service DAAD.

REFERENCES

1. Everitt B., Landau S., Morven L. et al. Cluster analysis. Chichester, Wiley, 2011, 330 p.
2. Aggarwal C., Reddy C., Chandan K. eds. Data Clustering : Algorithms and Applications. New York, Chapman and Hall/CRC, 2016, 652 p.
3. Huang Z. Extensions to the k-means algorithm for clustering large data sets with categorical values, *Data Mining and Knowledge Discovery*, 1998, Vol. 2, Issue 3, pp. 283–304. DOI:10.1023/A:1009769707641.S2CID 11323096.
4. Ng R., Han J. Efficient and effective clustering method for spatial data mining, *20th International Conference on Very Large Data Bases (VLDB'94), September 12–15, 1994, Santiago, Chile, proceedings*. Burlington, Morgan Kaufmann, 1994, pp. 144–155.
5. Bailey K. D. Typologies and Taxonomies: An Introduction to Classification Techniques. London, Sage Publications, 1994, 96 p.
6. Gordon A.D. Classification. Boca Raton, Chapman & Hall/CRC, 1999, 256 p.
7. Romesburg C. H. Cluster Analysis for Researchers. Belmont, Lifetime Learning Publications, 1984, 334 p.
8. Aldenderfer M. S., Blashfield R. K. Cluster Analysis. London, Sage Publications, 1984, 88 p.
9. Meilă, M. Comparing Clusterings by the Variation of Information, *Lecture Notes in Computer Science*, 2003, Vol. 2777, pp. 173–187. DOI:10.1007/978-3-540-45167-9_14.
10. Kraskov A., Stögbauer H., Andrzejak R. G., Grassberger P. Peter Hierarchical Clustering Based on Mutual Information, [Electronic resource]. Access mode: <https://arxiv.org/abs/q-bio/0311039>.

11. Frey B. J., Dueck D. Clustering by Passing Messages Between Data Points, *Science*, 2007, Vol. 315, № 5814, pp. 972–976. DOI: 10.1126/science.1136800.
12. Pfützner D., Leibbrandt R., Powers D. Characterization and evaluation of similarity measures for pairs of clusterings, *Knowledge and Information Systems*, 2009, Vol. 19, № 3, pp. 361–394. DOI:10.1007/s10115-008-0150-6.
13. Dunn J. Well separated clusters and optimal fuzzy partitions, *Journal of Cybernetics*, 1974, № 4, pp. 95–104. DOI:10.1080/01969727408546059.
14. Rand W. M. Objective criteria for the evaluation of clustering methods, *Journal of the American Statistical Association*, 1971, Vol. 66 (336), pp. 846–850. DOI:10.2307/2284239.
15. Hubert L., Arabie P. Comparing partitions, *Journal of Classification*, 1985, Vol. 2, pp. 193–218. DOI:10.1007/BF01908075.S2CID189915041
16. Di Marco A., Navigli R. Clustering and Diversifying Web Search Results with Graph-Based Word Sense Induction, *Computational Linguistics*, 2013, Vol. 39, № 3, pp. 709–754. DOI:10.1162/COLI_a_00148. S2CID 1775181.
17. Arnott R. D. Cluster Analysis and Stock Price Comovement, *Financial Analysts Journal*, 1980, Vol. 36, № 6, pp. 56–62. DOI: 10.2469/faj.v36.n6.56. ISSN 0015-198X.
18. Dunn J. C. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters, *Journal of Cybernetics*, 1973, Vol. 3, Issue 3, pp. 32–57. DOI:10.1080/01969727308546046.
19. Ahmed M., Yamany S., Mohamed N., Farag A., Moriarty T. A Modified Fuzzy C-Means Algorithm for Bias Field Estimation and Segmentation of MRI Data, *IEEE Transactions on Medical Imaging*, 2002, Vol. 21, № 3, pp. 193–199. DOI:10.1109/42.996338.
20. Abonyi J., Feil B. Cluster Analysis for Data Mining and System Identification. Berlin, Birkhäuser Verlag, 2007, 306 p. DOI: 10.1007/978-3-7643-7988-9
21. Höppner F., Klawonn F., Kruse R., Runkler T. Fuzzy Cluster Analysis: Methods for Classification, Data Analysis and Image Recognition. Chichester, John Wiley & Sons, 1999, 304 p.
22. Miyamoto S. Fuzzy Sets in Information and Retrieval and Cluster Analysis. Dordrecht, Kluwer Academic Publishers, 1990, 274 p.
23. Banerjee T. Day or Night Activity Recognition From Video Using Fuzzy Clustering Techniques, *IEEE Transactions on Fuzzy Systems*, 2014, Vol. 22, Issue 3, pp. 483–493. DOI: 10.1109/TFUZZ.2013.2260756.
24. Valente de Oliveira J., Pedrycz W. eds. Advances in Fuzzy Clustering and its Applications. Chichester, John Wiley & Sons, 2007, 454 p.
25. Bezdek J. C. Pattern Recognition with Fuzzy Objective Function Algorithms. New York, Plenum Press, 1981, 272 p. DOI: 10.1007/978-1-4757-0450-1
26. Dumitrescu D., Lazzerini B., Jain L. C. Fuzzy Sets and Applications to Clustering and Training. Boca Raton, CRC Press, 2000, 664 p.
27. Achtert E., Böhm C., Kriegel H.-P., Kröger P., Müller-Gorman I., Zimek A. Finding Hierarchies of Subspace Clusters, *Lecture Notes in Computer Science*, 2006, Vol. 4213, pp. 446–453. DOI:10.1007/11871637_42. ISBN 978-3-540-45374-1.
28. Achtert E., Böhm C., Kriegel H. P., Kröger P., Müller-Gorman I., Zimek A. Detection and Visualization of Subspace Cluster Hierarchies, *Lecture Notes in Computer Science*, 2007, Vol. 4443, pp. 152–163. DOI:10.1007/978-3-540-71703-4_15.
29. Johnson S. C. Hierarchical clustering schemes, *Psychometrika*, 1967, Vol. 32, № 3, pp. 241–254. DOI:10.1007/BF02289588
30. Jafari O., Maurya P., Nagarkar P., Islam K. M., Crushev C. A Survey on Locality Sensitive Hashing Algorithms and their Applications [Electronic resource]. Access mode: <https://arxiv.org/pdf/2102.08942>
31. Buhler J. Efficient large-scale sequence comparison by locality-sensitive hashing, *Bioinformatics*, 2001, Vol. 17, № 5, pp. 419–428.
32. Zhao K., Lu H., Mei J. Locality Preserving Hashing, *Twenty-Eighth AAAI Conference on Artificial Intelligence, 27–31 July 2014, Québec, proceedings*. Palo Alto, AAAI Press, 2014, pp. 2874–2880.
33. Tsai Y.-H., Yang M.-H. Locality preserving hashing, *2014 IEEE International Conference on Image Processing (ICIP), Paris, 27–30 of October 2014, proceedings*. Los Alamitos, IEEE, 2014, pp. 2988–2992. DOI: 10.1109/ICIP.2014.7025604.
34. Weinberger K., Dasgupta A., Langford J., et al. Feature Hashing for Large Scale Multitask Learning, *26th Annual International Conference on Machine Learning (ICML '09) Montreal, June 2009, proceedings*. New York, ACM, 2009, pp. 1113–1120. DOI: 10.1145/1553374.1553516
35. Wolfson H. J., Rigoutsos I. Geometric Hashing: An Overview, *IEEE Computational Science and Engineering*, 1997, Vol. 4, No 4, pp. 10–21.
36. Fast supervised discrete hashing / [J. Gui, T. Liu, Z. Sun et al.] // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. – 2017. – Vol. 40, No 2. – P. 490–496. DOI: 10.1109/TPAMI.2017.2678475
37. Aluç, G., Özsü M., Daudjee K. Building self-clustering RDF databases using Tunable-LSH, *The VLDB Journal*, 2018, Vol. 28, № 2, pp. 173–195. DOI:10.1007/s00778-018-0530-9
38. Pauleve L., Jegou H., Amsaleg L. Locality sensitive hashing: A comparison of hash function types and querying mechanisms, *Pattern Recognition Letters*, 2010, Vol. 31, № 11, pp. 1348–1358. DOI:10.1016/j.patrec.2010.04.004.
39. Andoni A., Indyk P. Near-Optimal Hashing Algorithms for Approximate Nearest Neighbor in High Dimensions, *Communications of the ACM*, 2008, Vol. 51, № 1, pp. 117–122. DOI:10.1145/1327452.1327494.
40. Salakhutdinov R., Hinton G. Semantic hashing, *International Journal of Approximate Reasoning*, 2008, Vol. 50, № 7, pp. 969–978. DOI:10.1016/j.ijar.2008.11.006.
41. Dahlgaard S., Knudsen M., Thorup M. Fast similarity sketching, *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS), 15–17 October 2017, Berkeley*. Los Alamitos: IEEE, 2017, pp. 663–671. DOI: 10.1109/FOCS.2017.67
42. Chin A. Locality-preserving hash functions for general purpose parallel computation, *Algorithmica*, 1994, Vol. 12, Issue 2–3, pp. 170–181. DOI: 10.1007/BF01185209. S2CID 18108051.
43. Subbotin S. A. Methods and characteristics of locality-preserving transformations in the problems of computational intelligence, *Radio Electronics, Computer Science, Control*, 2014, № 1, pp. 120–128. DOI: 10.15588/1607-3274-2014-1-17
44. Subbotin S. A., Blagodarev A. Yu., Gofman Ye. A. The neuro-fuzzy diagnostic model synthesis with hashed transformation in the sequence and parallel mode, *Radio Elec-*

- tronics, Computer Science, Control*, 2017, No. 1, pp. 56–65. DOI: 10.15588/1607-3274-2017-1-7
45. Subbotin S. A. The polar coordinates based hashing for data dimensionality reduction, *Radio Electronics, Computer Science, Control*, 2020, № 4, pp. 118–128. DOI: 10.15588/1607-3274-2020-4-12
46. Subbotin S. A., Oleynik A. A. Analiz preobrazovaniy dlya proyetsirovaniya dannykh na obobshchennuyu os' v zadachakh raspoznavaniya obrazov, *Shtuchniy intelekt*, 2010, № 1, pp. 114–121.
47. Subbotin S. A. Constructed features for automatic classification of stationary timing signals, *Radio Electronics, Computer Science, Control*, 2012, № 1, pp. 96–103. DOI: 10.15588/1607-3274-2012-1-19
48. Subbotin S. A. The complex data dimensionality reduction for diagnostic and recognition model building on precedents, *Radio Electronics, Computer Science, Control*, 2016, No. 4, pp. 70–76. DOI: 10.15588/1607-3274-2016-4-9
49. Subbotin S. A. Evaluation of informativity and selection of instances based on hashing, *Radio Electronics, Computer Science, Control*, 2020, № 3, pp. 129–137. DOI: 10.15588/1607-3274-2020-3-12
50. Oliinyk A., Subbotin S., Lovkin V., Blagodariv O., Zaiko T. The system of criteria for feature informativeness estimation in pattern recognition, *Radio Electronics, Computer Science, Control*, 2017, № 4, pp. 85–96. DOI: 10.15588/1607-3274-2017-4-10
51. Subbotin S. eds.: Bris R., Majernik J., Pancercz K., Zaitseva E. The instance and feature selection for neural network based diagnosis of chronic obstructive bronchitis, *Applications of Computational Intelligence in Biomedical Technology*. Cham, Springer, 2016, pp. 215–228. (Studies in Computational Intelligence, Vol. 606).
52. Subbotin S. The neuro-fuzzy network synthesis and simplification on precedents in problems of diagnosis and pattern recognition, *Optical Memory and Neural Networks (Information Optics)*, 2013, Vol. 22, № 2, pp. 97–103.
53. Subbotin S., Oliinyk A. The Fully-Defined Neuro-Fuzzy Model Synthesis, *Data Stream Mining & Processing (DSMP), 2016 IEEE First International Conference*, Lviv, 23–27 August 2016, proceedings. Lviv: NU “Lvivska Politechnika”, 2016. – P. 9–14.
54. Oliinyk A. O., Skrupsky S. Yu., Subbotin S. A. Using parallel random search to train fuzzy neural networks, *Automatic Control and Computer Sciences*, 2014, Vol. 48, №. 6, pp. 313–323.
55. Subbotin S. The method of a structural-parametric synthesis of neuro-fuzzy diagnostic model based on the hybrid stochastic search, *The experience of designing and application of CAD systems in microelectronics : XI International conference CADSM–2011, Polyana-Svalyava, 23–25 February 2011, proceedings*. Lviv, NU “Lviv Polytechnic”, 2011, pp. 248–249.
56. Subbotin S. A. Building a fully defined neuro-fuzzy network with a regular partition of a feature space based on large sample, *Radio Electronics, Computer Science, Control*, 2016, № 3, pp. 47–53. DOI: 10.15588/1607-3274-2016-3-6
57. Halgamuge S. K. A trainable transparent universal approximator for defuzzification in Mamdani-type neuro-fuzzy controllers, *IEEE Transactions on Fuzzy Systems*, Vol. 6, № 2, pp. 304–314. DOI: 10.1109/91.669031.

Received 10.09.2022.
Accepted 01.11.2022.

УДК 004.93

НЕЙРО-НЕЧІТКА МЕРЕЖА ДЛЯ КЛАСТЕРИЗАЦІЇ ДАНИХ З ХЕШУВАННЯМ ВІДСТАНЕЙ ТА САМОНАВЧАННЯМ

Субботін С. О. – д-р техн. наук, професор, завідувач кафедри програмних засобів Національного університету «Запорізька політехніка», Запоріжжя, Україна.

АНОТАЦІЯ

Актуальність. Для аналізу даних різної природи та розмірності широко застосовують кластерний аналіз. Однак відомі методи кластер-аналізу характеризуються низькою швидкістю та є вимогливими до ресурсів пам'яті ЕОМ внаслідок необхідності розрахунку попарних відстаней між екземплярами у багатовимірному просторі ознак. Крім того, результати відомих методів кластер-аналізу складні для сприйняття та аналізу людиною при великій кількості ознак.

Мета – підвищення швидкості кластер-аналізу, інтерпретабельності одержуваного розбиття на кластери, а також зниження вимог кластер-аналізу до пам'яті ЕОМ.

Метод. Запропоновано метод кластер-аналізу багатовимірних даних, який для кожного екземпляра обчислює його хеш на основі відстані до умовного центру координат, використовує одновимірну координату по осі хешу для визначення відстаней між екземплярами, розглядає отриманий хеш як псевдовихідну ознаку, розбивши її на інтервали, яким співставляє мітки псевдокласів-кластерів, отримавши грубе чітке розбиття простору ознак і екземплярів вибірки, автоматично формує розбиття вхідних ознак на нечіткі терми, визначає правила віднесення екземплярів до кластерів і в результаті формує систему нечіткого виведення типу класифікатора Мамдані-Заде, який у вигляді нейро-нечіткої мережі донавчається для забезпечення прийняттого значення функціоналу якості кластеризації. Це дозволяє скоротити кількість використовуваних термів і ознак, оцінити їх внесок у прийняття рішень про віднесення екземплярів до кластерів, підвищити швидкість кластер-аналізу даних, а також підвищити інтерпретабельність отриманого розбиття даних на кластери.

Результати. Розроблено математичне забезпечення, що дозволяє вирішувати завдання кластерного аналізу даних в умовах великої розмірності даних, проведено експерименти, що підтвердили працездатність розробленого математичного забезпечення.

Висновки. Розроблений метод та його програмна реалізація можуть бути рекомендовані для використання практиці у завданнях аналізу даних різної природи та розмірності.

КЛЮЧОВІ СЛОВА: кластер-аналіз, нейро-нечітка мережа, хеш, нечітке виведення, аналіз даних.

ЛІТЕРАТУРА / LITERATURE

1. Everitt B. Cluster analysis / [B. Everitt, S. Landau, L. Morven et al.]. – Chichester : Wiley, 2011. – 330 p.
2. Data Clustering: Algorithms and Applications / eds.: C. Aggarwal, C. Reddy, K. Chandan. – New York : Chapman and Hall/CRC, 2016. – 652 p.
3. Huang Z. Extensions to the k-means algorithm for clustering large data sets with categorical values / Z. Huang // Data Mining and Knowledge Discovery. – 1998. – Vol. 2, Issue 3. – P. 283–304. DOI:10.1023/A:1009769707641.S2CID 11323096.
4. Ng R. Efficient and effective clustering method for spatial data mining / R. Ng, J. Han // 20th International Conference on Very Large Data Bases (VLDB'94), September 12–15, 1994, Santiago, Chile : proceedings. – Burlington : Morgan Kaufmann, 1994. – P. 144–155.
5. Bailey K. D. Typologies and Taxonomies: An Introduction to Classification Techniques / K. D. Bailey. – London: Sage Publications, 1994. – 96 p.
6. Gordon A. D. Classification / A. D. Gordon. – Boca Raton : Chapman & Hall/CRC, 1999. – 256 p.
7. Romesburg C. H. Cluster Analysis for Researchers / C. H. Romesburg. – Belmont : Lifetime Learning Publications, 1984. – 334 p.
8. Aldenderfer M. S. Cluster Analysis / M. S. Aldenderfer, R. K. Blashfield. – London : Sage Publications, 1984. – 88 p.
9. Meilă M. Comparing Clusterings by the Variation of Information / M. Meilă // Lecture Notes in Computer Science. – 2003. – Vol. 2777. – P. 173–187. DOI:10.1007/978-3-540-45167-9_14.
10. Hierarchical Clustering Based on Mutual Information / [A. Kraskov, H. Stögbauer, R. G. Andrzejak, P. Grassberger, Peter]. [Electronic resource]. – Access mode: <https://arxiv.org/abs/q-bio/0311039>.
11. Frey B. J. Clustering by Passing Messages Between Data Points / B. J. Frey, D. Dueck // Science. – 2007. – Vol. 315, № 5814. – P. 972–976. DOI: 10.1126/science.1136800.
12. Pfitzner D. Characterization and evaluation of similarity measures for pairs of clusterings / D. Pfitzner, R. Leibbrandt, D. Powers // Knowledge and Information Systems. – 2009. – Vol. 19, № 3. – P. 361–394. DOI:10.1007/s10115-008-0150-6.
13. Dunn J. Well separated clusters and optimal fuzzy partitions / J. Dunn // Journal of Cybernetics. – 1974. – № 4. – P. 95–104. DOI:10.1080/01969727408546059.
14. Rand W. M. Objective criteria for the evaluation of clustering methods / W. M. Rand // Journal of the American Statistical Association. – 1971. – Vol. 66 (336). – P. 846–850. DOI:10.2307/2284239.
15. Hubert L. Comparing partitions / L. Hubert, P. Arabie // Journal of Classification. 1985. – Vol. 2. – P. 193–218. DOI:10.1007/BF01908075.S2CID189915041
16. Di Marco A. Clustering and Diversifying Web Search Results with Graph-Based Word Sense Induction / A. Di Marco, R. Navigli // Computational Linguistics. – 2013. – Vol. 39, № 3. – P. 709–754. DOI:10.1162/COLI_a_00148. S2CID 1775181.
17. Arnott R. D. Cluster Analysis and Stock Price Comovement / R. D. Arnott // Financial Analysts Journal. – 1980. – Vol. 36, № 6. – P. 56–62. DOI: 10.2469/faj.v36.n6.56. ISSN 0015-198X.
18. Dunn J. C. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters / J. C. Dunn // Journal of Cybernetics. – 1973. – Vol. 3, Issue 3. – P. 32–57. DOI:10.1080/01969727308546046.
19. A Modified Fuzzy C-Means Algorithm for Bias Field Estimation and Segmentation of MRI Data / [M. Ahmed S. Yamany, N. Mohamed, A. Farag, T. Moriarty] // IEEE Transactions on Medical Imaging. – 2002. – Vol. 21, № 3. – P. 193–199. DOI:10.1109/42.996338.
20. Abonyi J. Cluster Analysis for Data Mining and System Identification / J. Abonyi, B. Feil. – Berlin : Birkhäuser Verlag, 2007. – 306 p. DOI: 10.1007/978-3-7643-7988-9
21. Fuzzy Cluster Analysis: Methods for Classification, Data Analysis and Image Recognition / [F. Höppner, F. Klawonn, R. Kruse, T. Runkler]. – Chichester: John Wiley & Sons, 1999. – 304 p.
22. Miyamoto S. Fuzzy Sets in Information and Retrieval and Cluster Analysis / S. Miyamoto. – Dordrecht : Kluwer Academic Publishers, 1990. – 274 p.
23. Banerjee T. Day or Night Activity Recognition From Video Using Fuzzy Clustering Techniques / T. Banerjee // IEEE Transactions on Fuzzy Systems. – 2014. – Vol. 22, Issue 3. – P. 483–493. DOI:10.1109/TFUZZ.2013.2260756.
24. Advances in Fuzzy Clustering and its Applications / eds.: J. Valente de Oliveira, W. Pedrycz. – Chichester : John Wiley & Sons, 2007. – 454 p.
25. Bezdek J. C. Pattern Recognition with Fuzzy Objective Function Algorithms / J. C. Bezdek. – New York : Plenum Press, 1981. – 272 p. DOI: 10.1007/978-1-4757-0450-1
26. Dumitrescu D. Fuzzy Sets and Applications to Clustering and Training / D. Dumitrescu, B. Lazzerini, L. C. Jain. – Boca Raton: CRC Press, 2000. – 664 p.
27. Finding Hierarchies of Subspace Clusters / [E. Achtert, C. Böhm, H.-P. Kriegel et al.] // Lecture Notes in Computer Science. – 2006. – Vol. 4213. – P. 446–453. DOI:10.1007/11871637_42. ISBN 978-3-540-45374-1.
28. Detection and Visualization of Subspace Cluster Hierarchies / [E. Achtert, C. Böhm, H. P. et al.] // Lecture Notes in Computer Science. – 2007. – Vol. 4443. – P. 152–163. DOI:10.1007/978-3-540-71703-4_15.
29. Johnson S. C. Hierarchical clustering schemes / S. C. Johnson // Psychometrika. – 1967. – Vol. 32, № 3. – P. 241–254. DOI:10.1007/BF02289588
30. A Survey on Locality Sensitive Hashing Algorithms and their Applications [Electronic resource] / [O. Jafari, P. Maurya, P. Nagarkar et al.]. – Access mode: <https://arxiv.org/pdf/2102.08942>
31. Buhler J. Efficient large-scale sequence comparison by locality-sensitive hashing / J. Buhler // Bioinformatics. – 2001. – Vol. 17, № 5. – P. 419–428.
32. Zhao K. Locality Preserving Hashing / K. Zhao, H. Lu, J. Mei // Twenty-Eighth AAAI Conference on Artificial Intelligence, 27–31 July 2014, Québec : proceedings. – Palo Alto: AAAI Press, 2014. – P. 2874–2880.
33. Tsai Y.-H. Locality preserving hashing / Y.-H. Tsai, M.-H. Yang // 2014 IEEE International Conference on Image Processing (ICIP), Paris, 27–30 of October 2014: proceedings. – Los Alamitos: IEEE, 2014. – P. 2988–2992. DOI: 10.1109/ICIP.2014.7025604.
34. Feature Hashing for Large Scale Multitask Learning / [K. Weinberger, A. Dasgupta, J. Langford et al.] // 26th Annual International Conference on Machine Learning (ICML '09) Montreal, June 2009 : proceedings. – New York : ACM, 2009. – P. 1113–1120. DOI: 10.1145/1553374.1553516
35. Wolfson H. J. Geometric Hashing: An Overview / H. J. Wolfson, I. Rigoutsos // IEEE Computational Science and Engineering. – 1997. – Vol. 4, No 4. – P. 10–21.

36. Fast supervised discrete hashing / [J. Gui, T. Liu, Z. Sun et al.] // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. – 2017. – Vol. 40, No 2. – P. 490–496. DOI: 10.1109/TPAMI.2017.2678475
37. Aluç, G. Building self-clustering RDF databases using Tunable-LSH / G. Aluç, M. Özsu, K. Daudjee // *The VLDB Journal*. – 2018. – Vol. 28, № 2. – P. 173–195. DOI:10.1007/s00778-018-0530-9
38. Pauleve L. Locality sensitive hashing: A comparison of hash function types and querying mechanisms / L. Pauleve, H. Jegou, L. Amsaleg // *Pattern Recognition Letters*. – 2010. – Vol. 31, № 11. – P. 1348–1358. DOI:10.1016/j.patrec.2010.04.004.
39. Andoni A. Near-Optimal Hashing Algorithms for Approximate Nearest Neighbor in High Dimensions / A. Andoni, P. Indyk // *Communications of the ACM*. – 2008. – Vol. 51, № 1. – P. 117–122. DOI:10.1145/1327452.1327494.
40. Salakhutdinov R. Semantic hashing / R. Salakhutdinov, G. Hinton // *International Journal of Approximate Reasoning*. – 2008. – Vol. 50, № 7. – P. 969–978. DOI:10.1016/j.ijar.2008.11.006.
41. Fast similarity sketching / [S. Dahlgaard, M. Knudsen, M. Thorup] // *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, 15–17 October 2017, Berkeley. – Los Alamitos : IEEE, 2017. – 663–671. DOI: 10.1109/FOCS.2017.67
42. Chin A. Locality-preserving hash functions for general purpose parallel computation / A. Chin // *Algorithmica*. – 1994. – Vol. 12, Issue 2–3. – P. 170–181. DOI:10.1007/BF01185209. S2CID 18108051.
43. Subbotin S. A. Methods and characteristics of locality-preserving transformations in the problems of computational intelligence / S. A. Subbotin // *Radio Electronics, Computer Science, Control*. – 2014. – № 1. – С. 120–128. DOI: 10.15588/1607-3274-2014-1-17
44. Субботин С. А. Синтез нейро-нечетких диагностических моделей с хэширующим преобразованием в последовательном и параллельном режимах / С. А. Субботин, А. Ю. Благодарев, Е. А. Гофман // *Radio Electronics, Computer Science, Control*. – 2017. – № 1. – С. 56–65. DOI: 10.15588/1607-3274-2017-1-7
45. Субботин С.А. Хэширование на основе полярных координат для сокращения размерности данных / С.А.Субботин // *Radio Electronics, Computer Science, Control*. – 2020. – № 4. – P. 118–128. DOI: 10.15588/1607-3274-2020-4-12
46. Субботин С. А. Анализ преобразований для процирования данных на обобщённую ось в задачах распознавания образов / С. А. Субботин, А. А. Олейник // *Штучний інтелект*. – 2010. – № 1. – С. 114–121.
47. Субботин С. А. Конструируемые признаки для автоматической классификации распределенных во времени стационарных сигналов / С. А. Субботин // *Radio Electronics, Computer Science, Control*. – 2012. – № 1. – С. 96–103. DOI: 10.15588/1607-3274-2012-1-19
48. Субботин С. А. Комплексное сокращение размерности данных для построения диагностических и распознающих моделей по прецедентам / С. А. Субботин // *Radio Electronics, Computer Science, Control*. – 2016. – № 4. – С. 70–76. DOI: 10.15588/1607-3274-2016-4-9
49. Субботин С.А. Оценка информативности и отбор экземпляров на основе хэширования / С.А. Субботин // *Radio Electronics, Computer Science, Control*. – 2020. – № 3. – С. 129–137. DOI: 10.15588/1607-3274-2020-3-12
50. Oliinyk A. The system of criteria for feature informativeness estimation in pattern recognition / [A. Oliinyk, S. Subbotin, V. Lovkin et al.] // *Radio Electronics, Computer Science, Control*. – 2017. – № 4. – С. 85–96. DOI: 10.15588/1607-3274-2017-4-10
51. Subbotin S. The instance and feature selection for neural network based diagnosis of chronic obstructive bronchitis / S. Subbotin // *Applications of Computational Intelligence in Biomedical Technology* / eds.: R. Bris, J. Majernik, K. Panczer, E. Zaitseva. – Cham : Springer, 2016: – P. 215–228. – (Studies in Computational Intelligence, Vol. 606).
52. Subbotin S. The neuro-fuzzy network synthesis and simplification on precedents in problems of diagnosis and pattern recognition / S. Subbotin // *Optical Memory and Neural Networks (Information Optics)*. – 2013. – Vol. 22, № 2. – P. 97–103.
53. Subbotin S. The Fully-Defined Neuro-Fuzzy Model Synthesis / S. Subbotin, A. Oliinyk // *Data Stream Mining & Processing (DSMP): 2016 IEEE First International Conference, Lviv, 23–27 August 2016 : proceedings*. – Lviv : NU “Lvivska Politechnika”, 2016. – P. 9–14.
54. Oliinyk A. O. Using parallel random search to train fuzzy neural networks / A. O. Oliinyk, S. Yu. Skrupsky, S. A. Subbotin // *Automatic Control and Computer Sciences*. – 2014. – Vol. 48. – №. 6. – P. 313–323.
55. Subbotin S. The method of a structural-parametric synthesis of neuro-fuzzy diagnostic model based on the hybrid stochastic search / S. Subbotin // *The experience of designing and application of CAD systems in microelectronics : XI International conference CADSM-2011, Polyana-Svalyava, 23–25 February 2011 : proceedings*. – Lviv : NU “Lviv Polytechnic”, 2011. – P. 248 – 249.
56. Субботин С. А. Построение полностью определенных нейро-нечетких сетей с регулярным разбиением пространства признаков на основе выборок большого объема / С. А. Субботин // *Radio Electronics, Computer Science, Control*. – 2016. – № 3. – С. 47–53. DOI: 10.15588/1607-3274-2016-3-6
57. Halgamuge S. K. A trainable transparent universal approximator for defuzzification in Mamdani-type neuro-fuzzy controllers / S. K. Halgamuge // *IEEE Transactions on Fuzzy Systems*. – Vol. 6, № 2. – P. 304–314. DOI: 10.1109/91.669031.

ПРОГРЕСИВНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

PROGRESSIVE INFORMATION TECHNOLOGIES

ПРОГРЕССИВНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

UDK [625.7:528.854]:004.9

INFORMATION TECHNOLOGY OF TRANSPORT INFRASTRUCTURE MONITORING BASED ON REMOTE SENSING DATA

Danshyna S. Yu. – Dr. Sc., Professor of Dept. of Geo-information Technologies and Space Monitoring of the Earth, National Aerospace University “KhAI”, Kharkiv, Ukraine.

Nechausov A. S. – PhD, Associate Professor of the Dept. of Geo-information Technologies and Space Monitoring of the Earth, National Aerospace University “KhAI”, Kharkiv, Ukraine.

Andriev S. M. – PhD, Associate Professor of the Dept. of Geo-information Technologies and Space Monitoring of the Earth, National Aerospace University “KhAI”, Kharkiv, Ukraine.

ABSTRACT

Context. In the light of current road network monitoring practices, this study aims to explore the capability of remote sensing technologies to solve the problems of increasing the objectivity of preliminary evaluations of the condition of the infrastructure as a whole. The object of the study was to process the monitoring of transport infrastructure (TI) to find ways to improve it in the implementation of development projects.

Objective. The goal of the work is to increase objectivity of decision-making on the evaluation, reconstruction, development of the transport network structure due to the visual presentation and disclosure of open data for monitoring the transport value.

Method. Existing approaches to TI monitoring and evaluating its condition are analyzed. The identified shortcomings, as well as the development of remote sensing technologies, open up prospects for the use of remote sensing data in the TI monitoring process. A set-theoretic model of the monitoring process information flows is proposed, the consistent refinement of the elements of which made it possible to develop information technology (IT). Formation of a set of input and output parameters of IT, the set of its operations, their representation with IDEFX-models set explains how a set of heterogeneous (graphic, text, digital, cartographic, etc.) data about TI elements coming from different sources are processed and presented to support decision-making on the survey of existing infrastructure and its improvement. The developed IT makes it possible to obtain complex indicators for analyzing the TI of a particular area, to solve the problems of inventorying objects, TI and its elements modeling, taking into account the physical and geographical location, which makes it possible to consider it as an auxiliary tool that complements existing methods of TI monitoring.

Results. The developed IT was studied in solving the problem of monitoring the TI section of the Kharkiv region using satellite imagery of medium (Sentinel-2) and high (SuperView-1) resolution and the results of laser survey of the road bridge across the river Mzha (as an element of infrastructure).

Conclusions. The conducted experiments confirmed the operability of the proposed information technology and showed expediency of its practical use in solving the problems of obtaining generalizing characteristics of the infrastructure, inventory of TI objects and their modeling. This opens up opportunities for substantiating project decisions for the reconstruction of the transport network and planning procedures for examining its condition. Prospects for further research may include: creating reference models of TI objects, expanding the table of decryption signs of road transport infrastructure objects, integrating remote data, survey results of TI sections and engineering surveys of objects to obtain evaluations of the condition of TI in general.

KEYWORDS: model of information flows of the process, IDEFX-models, mapping and 3D-modeling.

ABBREVIATIONS

IT is an information technology;
GIS is a geoinformation systems;
GPR is ground penetrating radar;
RMC are road-maintenance companies;
TI is a transport infrastructure.

NOMENCLATURE

φ is an update function;

ψ is an output function, that generates an output data;
 A is a set of operations that implement the transport infrastructure monitoring process;
 H is a population size, thousand people;
 I_Pr is a model of information flows of the transport infrastructure monitoring process;
 J_k are junctions, that define the interaction logic of IDEF3-model operations;

$K(E)$ is an Engel coefficient;
 $K(G)$ is a Goltz coefficient;
 L is the length of roads in the given territory, km;
 N is the number of settlements;
 O is a set of output data the TI monitoring process;
 r is the correlation coefficient;
 S is the area of the territory, km²;
 V is a set of input data, incoming into an input of an IT in the TI monitoring process;
 Z is a set of documents regulating the TI monitoring process.

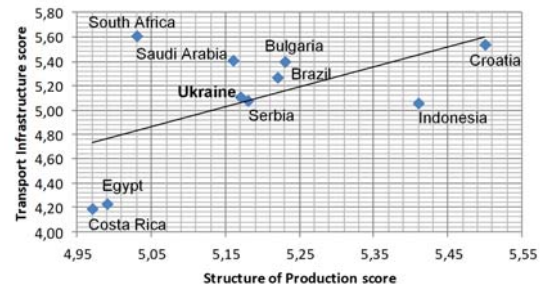
INTRODUCTION

The development of countries, the strengthening of international relations, the activation of globalization processes increases the importance of transport as a factor in economic and social development [1]. Thus, experts from the World Economic Forum note that for countries and economies are assigned to nascent archetypes, is a medium-pronounced dependence (with a correlation coefficient $r = 0.567$) between indicators of TI and the structure of production components, reflecting the complexity and scale of the country's current production base (Fig. 1a). A more pronounced relationship ($r = 0.637$) between indicators of TI and drivers of production, that position a country to capitalize on emerging technologies and opportunities in the future of production (Fig. 1b) [2]. Thus, transport and the transport sector have an impact on the location and efficiency of production, on the formation of local and national markets, and on the solution of socio-economic problems [1, 3].

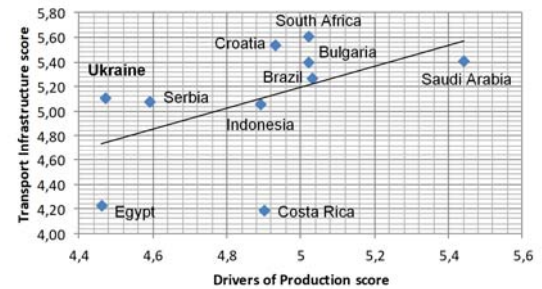
Determining the cost of production, on the one hand, TI turns out to be a measure of economic activity; on the other hand, it becomes its reflection, forcing the business to coordinate the development of its enterprises with the development plans of TI [2–4]. In this context, improving the efficiency of the existing TI becomes the main goal of government programs and development projects, and understanding the quantity, quality of roads and the entire infrastructure as a whole is necessary for making strategic decisions.

The object of study is the process of monitoring transport infrastructure in order to find ways to improve it when implementing development projects.

One of the most promising and powerful monitoring tools for TI is considered to be methods based on remote sensing of its elements, which traditionally focus on monitoring the condition of the road surface [5]. However, it is noted that TI is a broader concept that combines not only roads as elements of communication networks for all types of transport, but also the corresponding set of structures to meet the needs of the population and production in transportation [6]. Therefore, in order to increase the objectivity of TI evaluation in general, it is necessary to expand the use of remote zoning by implementing models and methods based on the analysis of spatially distributed data about its elements.



a



b

Figure 1 – The scatterplot (for the top 10 countries of the nascent archetypes) that illustrates dependence between transport infrastructure components and indicators: a – structure of production; b – drivers of production (According to the results of the analysis of work [2])

The subject of study is the information technology for transport infrastructure monitoring, which uses remote sensing data to analyze it and find ways to improve it.

Increasingly, urban planners and transport engineers note that there is a need to create scientific and methodological support that combines spatial data with evaluations of TI to provide information support for its monitoring process while making decisions on ways to improve it. The growing number of vehicles, in particular, on highways, increases the need to step up efforts to implement IT of TI monitoring as a set of methods for searching, processing and presenting heterogeneous data in the implementation of development projects. Therefore, the known sampling methods [5–16], which, unfortunately, are focused on the remote evaluation of only the road surface, have excellent prospects for using the entire road infrastructure in IT for monitoring.

The purpose of the work is to increase the objectivity of the decisions made on the survey, reconstruction, development of the existing transport network through the representation and visualization of spatial data for monitoring of the TI.

1 PROBLEM STATEMENT

Assume there is a set of data $V = \{v_i\}, i = \overline{1, n}$, obtained during the TI monitoring process and incoming on the input of IT. Next, this data is transformed into a set $O = \{o_j\}, j = \overline{1, m}$ – a set of output data for analysis and

search for ways to improve the TI. The rules for this transformation are set by the function

$$f: V \rightarrow O, \forall v_i \in V \exists o_j \in O: v_i = f(o_j). \quad (1)$$

It is necessary to define actions within the framework of information technology for TI monitoring, which implements the mapping $V \xrightarrow{f} O$ in expression (1).

2 THE LITERATURE REVIEW

Among economists, the generalizing Engel or Goltz coefficients are widely used as a tool for evaluating the condition of the transport component. These coefficients, based on statistical data, show the level of provision of the population with a transport network or the level of development of this network between settlements, taking into account the length of roads, the area of the analyzed territory, the population and the number of settlements. To calculate the level of provision of the population of the analyzed territory with the transport network, we use Engel's formula [6, 7]:

$$K(E) = \frac{L}{\sqrt{SH}}. \quad (2)$$

and to calculate transport network development level – Goltz's formula [7]:

$$K(G) = \frac{L}{\sqrt{SN}}. \quad (3)$$

These coefficients make it possible, with a certain reliability, to judge the level of development of transport networks in relation to their main users and to determine the main differences in development in the study areas. However, the coefficients useful for the system analysis of the transport component of the territories do not take into account the configuration of TI, the condition of the road surface, its objects, etc. [6, 7]. Therefore, more and more often, urban planners and transport engineers come to the need to create indicators for sustainable planning and making informed decisions that combine spatial data with evaluations of TI [8, 9].

The regulatory database for evaluating the condition of roads in Ukraine is regulated by a number of state, industry and departmental standards (for example, GOST 8747:2017, BC B.2.3-4-2007, BC B.2.3-218-534:2011, DBC H.1-218-530.2006 etc.). For employees of road maintenance organizations (RMC), the standards define the rules for expert-visual or visual-instrumental control of the road surface, all structures and elements of TI. However, due to the high operating costs of specialized inspection vehicles, conventional monitoring systems suffer from a limited amount of data collected from periodic inspections, which makes it impossible to fully evaluate the condition and form a clear plan for upgrading roads, structures and TI elements. This makes road survey work cumbersome and inefficient [6, 10]: some checks

are redundant, and some lead to belated detection of problems. At the same time, visual methods are resource-intensive (in terms of human, time and financial resources), not promptly (do not allow you to clearly and quickly obtain data on the condition of TI, develop project documentation, etc.) and subjective (significantly depend on the experience and qualifications of employees exercising control roads) [10, 11].

It is precisely because of the outdated and poor functioning of TI monitoring systems that today RMC are reforming their observational methods and, supporting world trends, are moving to new competitive technologies that can detect and analyze the condition of TI in a short time and with high accuracy [11].

An analysis of the development of approaches to TI monitoring (A. Shtayat, et al. [11]) shows the presence of trends in the use of remote methods to solve the problems of evaluating its condition. The existing monitoring methods are conditionally divided into two groups: static methods – focused on the use of stationary sensors or instruments; dynamic methods – require certain actions to collect data on the condition of TI. At the same time, the growing appearance on the market of drones, video cameras, GPR, laser scanners, etc. as a constant and periodic source of information, contributes to monitor the condition of roads in real time (D. Gura, et al. [5]).

Static remote methods implement the ideas of non-destructive control to obtain accurate and valuable information about TI objects without direct physical contact with them. They have proven themselves well in the problems of determining local surface and structural deformations in linear transport networks (F. Tosti, et al. [12]). With low operating costs, non-contact information, and less labor required, these approaches enable early inspection of pavements, optimizing maintenance and repair practices, reducing maintenance costs, and extending pavement life (M. Rasol, et al. [13]). However, it is fair to say that the potential of these methods in TI monitoring systems is still not well understood, and they are not suitable for large-scale studies of surface deformation of the road network [5, 11, 12].

Even more theoretical are studies related to the use of dynamic methods for TI monitoring. The few studies in this area are focused on a dynamic monitoring system using portable equipment. For example, Khoudeir et al. [14] experimentally prove that, in comparison with other monitoring systems, the use of digital images of the road surface can increase the efficiency of its evaluation and operational safety. In addition, digital images can be used to evaluate any type of pavement, as well as unpaved roads. For example, Amarasiri S., et al. [15] based on the evaluation of the change in the brightness level of the image of TI objects, conclusions are drawn about the potential possibility of monitoring the degradation of the macrotecture of the road surface. Evaluating the prospects of dynamic monitoring methods, many experts consider them the most effective and recommend them, in particular, for information systems for evaluating the condition of concrete pavement [11]. We also note that in the case

of using spatial data, an important point is the ability to track the dynamics of TI development, highlighting common features and main geographical patterns [8, 16].

Thus, regardless of the aspects of the monitoring problem considered, the TI literature review results confirm the following. The effectiveness of TI status evaluation is reduced due to the large size of the geographical areas that need to be controlled, the limited amount of human and financial resources available for RMC [10, 11, 17]. At the same time, studies of the possibility of using remote methods confirm the reduction of socio-economic costs due to such unique advantages [10, 17]:

- a large amount of data increases the accuracy of predicting the condition of TI;

- the cumulative measurement of the current condition of the object increases the accuracy of the inspection, contributes to the early detection of problems and reduces the number of visits of the TI facilities with an inspection;

- reducing the time of road maintenance work while complying with RMC policies based on continuous monitoring of the TI condition.

Therefore, in order to increase the objectivity of monitoring results based on the analysis of independent heterogeneous data obtained by different methods from different sources, it is necessary to implement IT that combines visual control of the TI condition with remote sensing data. Implementing the concept of non-destructive control to obtain accurate and timely information about the condition of TI, such technologies [11, 12, 17]:

- allow you to create lists of works for maintenance, renovation and / or reconstruction of TI facilities, reducing the total cost of maintenance;

- help RMC prioritize finding ways to improve TI by formulating plans and investment strategies for development projects;

- make it possible to track the dynamics, establish common features and patterns of TI development, etc.

3 MATERIALS AND METHODS

Today, the legislation of Ukraine in the field of evaluating the condition of roads is in the process of formation, only some operations of the TI monitoring process are regulated at the level of industry standards (for example, [18, 19]). Traditionally, the standards prescribe to collect data (the initial elements of the set $V = \{v_i\}$) by expert-visual and visual-instrumental methods, the choice of which is determined by road-maintenance priorities, available resources and geographical limitations [17]. The global practice of RMC “transfer of responsibility” embedded in the regulatory framework can significantly reduce the objectivity of the obtained evaluations of the TI condition [6, 15], causes delays in problems detection [17], increases overall maintenance costs [20], reduce road safety and quality of TI [12]. In order to eliminate the possible listed problems, it is proposed to expand the set V by adding data obtained by remote sensing methods.

When monitoring the TI, an idea is formed about the actual condition of the road surface and infrastructure elements to find ways to improve it, i.e., in accordance

with expression (1), the resulting elements of the set V must be transformed by IT into a set $O = \{o_j\}$ for evaluating the TI condition.

At the same time, like any information process, the TI monitoring process under consideration requires algorithmizing and formalization to eliminate chaos, determination of a clear sequence of operations, ease of control over their implementation, etc., as well as for systematizing the data, requirements, norms and rules in one document arising from the provisions of various regulatory documents [21].

To determine the procedure for actions that implement the mapping $V \xrightarrow{f} O$, using the author’s methodology for studying information processes, we will conceptually present the process of TI monitoring as a set-theoretic model of its information flows [21–23]

$$I_Pr = (V, O, A, \psi, Z, \varphi). \quad (4)$$

Based on the requirements of the standards [18, 19] and taking into account global trends [5, 17], we will form the set V of the model (4) from the following elements: v_1 – satellite imagery of the TI study area; v_2 – survey results of the TI section; v_3 – results of engineering surveys of the TI section objects; v_4 – base map. At the output of the process, a set O is formed, which combines the following elements: o_1 – TI section objects database; o_2 – the list of objects of problematic TI section; o_3 – TI section condition evaluations. It is possible by implementing the operations defined by the set A : a_1 – to define the objects of the TI section; a_2 – to check the condition of the TI section objects; a_3 – to form TI section condition evaluations. To normalize the internal content of information flows during TI monitoring, documents that specify the requirements for TI objects, the rules for their definition, control and operation are used. This means that the set Z (a set of documents regulating the process) will combine the elements: z_1 – table of decryption signs of objects; z_2 – requirements of the Building Code; z_3 – TI section objects evaluation requirements. Also, in model (4) ψ means the output function, which uniquely determines the rules for the formation of elements of the set O ; φ – update function, the execution of which is necessary to clarify the input data of the process in accordance with the requirements of the elements of the set Z .

TI monitoring is carried out discretely in time, which means that the inputs and outputs will change depending on the requirements of the regulatory documents, the initial data at the input and the set of operations that implement the process:

$$\begin{cases} V(t) = \varphi(V(t-1), Z(t)); \\ O(t) = \psi(V(t), A(t)), \end{cases} \quad (5)$$

that is, the input of the process at the present $V(t)$ depends on the input at the previous moment of time and the set of documents regulating the process, $Z(t)$; the output of the process $O(t)$ is determined by the set $V(t)$ and the operations $A(t)$ performed in the present.

The output function – displaying the view:

$$\psi : A \times V \rightarrow O, \quad (6)$$

which depends on the complexity of the process, can be presented in tabular, graph and graphical view [21–23].

Defining the mapping (6) in the model (4), we present the function ψ in graph form (Fig. 2) in accordance with the rules formed in [22].

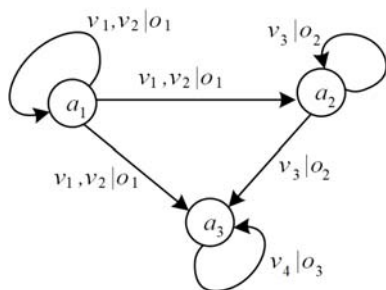


Figure 2 – Output function graph view (6)

Output function graph view (6) – it is a directed graph representation (Fig. 2), whose vertices correspond to the operations of the set $A = \{a_g\}, g = 1, \dots, 3$, and edges –

possible transitions from one operation of the process to another. Each edge has a weight – specifying an element of the set $V = \{v_i\}, i = 1, \dots, 4$, on which there is a transition from the execution of one operation a_g to another, and an element of the set $O = \{o_j\}, j = 1, \dots, 3$ as a result a_g , necessary to perform subsequent operations [22]. For example, at time t , the arrival of the operation at the input a_3 element v_4 allows to form an output element o_3 . But at the same time, an output o_1 is needed, resulting from an operation a_1 by combining an input data v_1 and v_2 , as well as the exit o_2 , as a result of operation a_2 .

For clarity and to facilitate the perception of data and operations of the I_Pr process, while maintaining the rigor and formality of the presentation, we depict the graph (Fig. 2) in the form of an IDEF0-model (Fig. 3) [22, 23]. Based on the provisions of the functional modeling methodology, IDEF0-model determines the structure of IT monitoring of the TI.

In the IDEF0-model, GIS is considered as a means of performing a set of operations A, and spatial analysis (or GIS analysis) of TI monitoring tasks solved with its help contains a set of methods and tools for combining geospatial data with the arguments of decision makers [3, 17, 23].

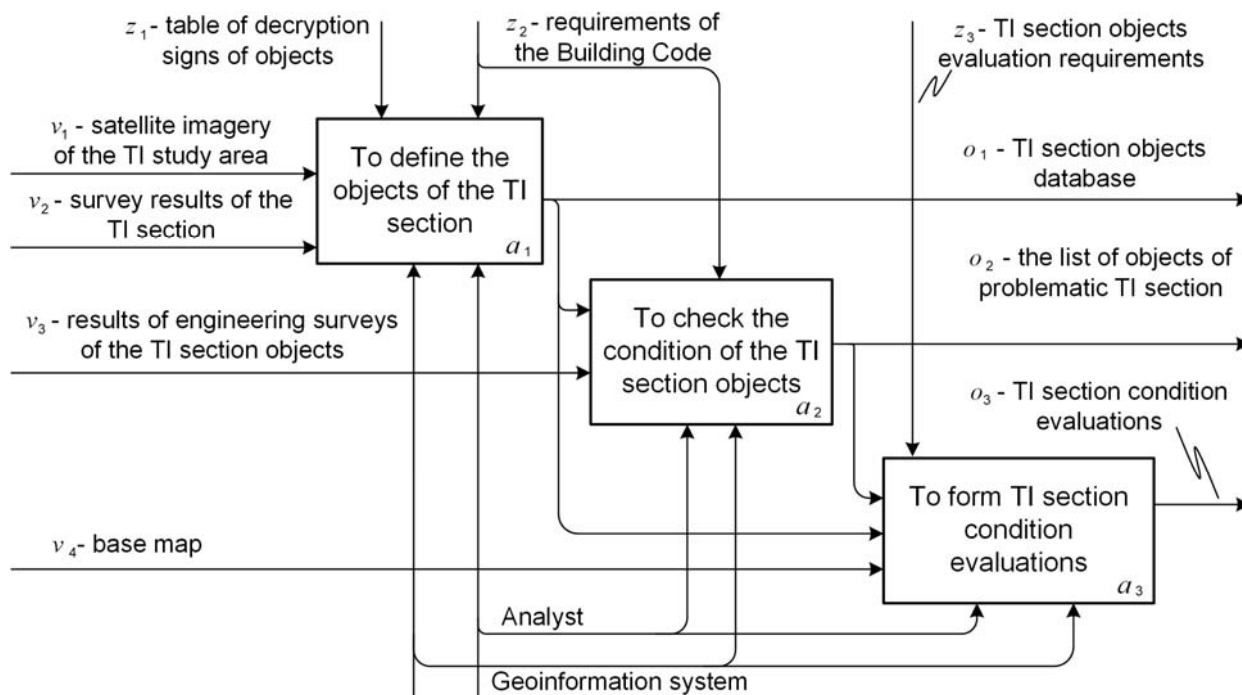


Figure 3 – IDEF0-model as a structure of information technology of transport infrastructure monitoring, that explains graphical view of function ψ realizations

Executing the update function – displaying the view [21–23]:

$$\varphi: V \times Z \rightarrow V \quad (7)$$

is necessary to refine the input data in accordance with the requirements of the documents of the set Z .

The simplest approach to the definition of function (7) is enumeration, when the expert associates each element of the set V with an implementation of the view:

$$v_i = \varphi(v_i, z_l) \quad \forall v_i \in V, i = \overline{1,4} \quad \forall z_l \in Z, l = \overline{1,3}. \quad (8)$$

Taking into account the dynamic nature of the TI monitoring process, based on the first expression of the system (5), experts need to discretely form a set of implementations (8) in time. On the other hand, in the context of digitalization of TI monitoring and its information support, the expert approach to the formation of implementations (8) contradicts the principles of standard procedures, adherence to uniform approaches and data processing technologies, formed, for example, in [5, 12, 15]. Dynamic nature of the function φ can be explained by using the IDEF3 methodology. It has been chosen due to the following reasons [21, 24]:

– IDEF3 reveals the implementation logic of set A operations, representing their dynamic sequence in the form

of a scenario that is implemented by IT in a finite time in accordance with the first expression of the system (5);

– IDEF3 is an extension of the IDEF0 standard, which is used to represent the function of outputs ψ ;

– IDEF3 provides a tool for creating a set of graphical models that reveal the mechanism for generating realization (8) of the update function φ of the process, while providing simplicity, clarity and ease of perception of the dynamic character of information flows;

– IDEF3 as a part of a structural analysis, has certain semantics for describing information processes, which facilitates their full understanding by developers and end users.

All of the above, on the one hand, makes it possible to take into account the dynamic nature of the TI monitoring process and, in particular, the update function φ according to (5), on the other hand, reveals the mechanism for obtaining realizations (8) in the form of a final sequence of process operations. A graphical representation of the implementations of the function φ based on the IDEF3 methodology, which explains the TI monitoring mechanism in accordance with the expressions of system (5), is shown in Fig. 4. The resulting IDEF3-model reveals the IT monitoring structure of TI for decision-making when looking for ways to improve it [24].

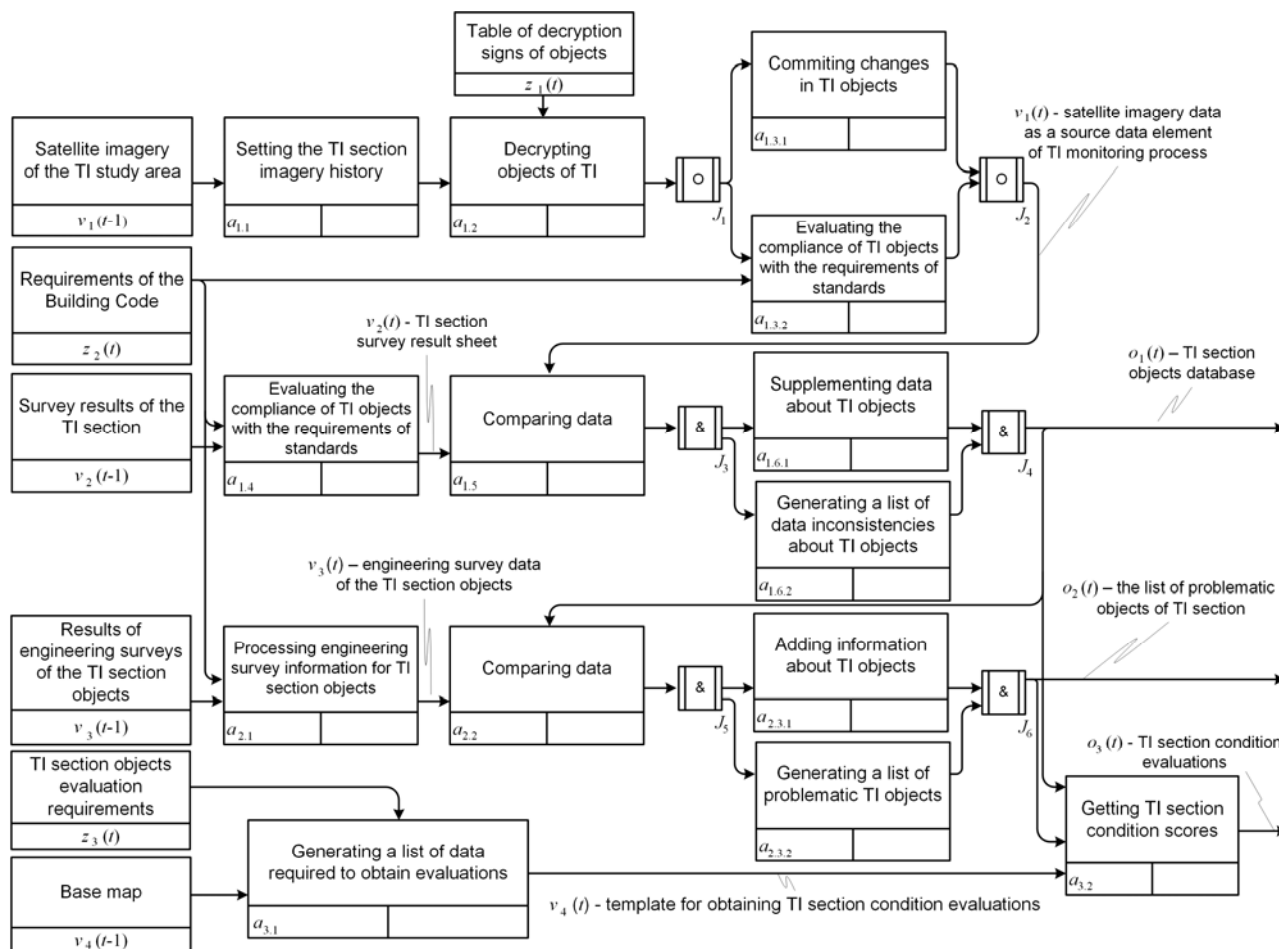


Figure 4 – IDEF3-model, that explains graphical view of function φ realizations

The sequence of actions on Fig. 4 is formed by summarizing the results of works [5, 12, 15, 17] and the recommendations of current standards [18, 19]. At the same time, the operations of the set $A = \{a_g\}, g = 1, \dots, 3$ are refined and revealed as the function $f: V \rightarrow O$ is implemented. This is explained by the corresponding identifiers of the unit of work of IDEF3-model (for example, the identifier $a_{3,1}$ denotes the first sub-operation of the operation a_3 , etc.). Also here, a temporary designation of the elements of the sets V and Z is additionally introduced to explain the dynamics of their change (which is not typical for the IDEF3 methodology).

The IDEF3-model uses typical definitions of industry standards [18, 19], for example, the TI section survey results sheet, etc. This is a document that organizes the results of a survey of a road surface by a visual or visual-instrumental method in order to determine its technical condition. Thus, the output o_1 , obtained from the junction J_4 , is a multidimensional database that combines in a single description heterogeneous information about static (TI object name, coordinates, geometric and operational characteristics, etc.) and dynamic (a snapshot of the object, the area of damage to the coverage on the date of the survey, the strength of concrete relative to the norms, etc.) characteristics of the objects obtained during TI monitoring. The algorithm for the formation of such a database on the example of another object of study is considered in the work of the authors [23]. The same data are input to operations a_2 and a_3 , where, in one case, they are considered as the basis for the formation of evaluations of the condition of the TI site, in the other, these data are refined by comparison with the results of engineering surveys of TI objects.

Thereby, the consistent refinement of the conceptual model of information flows of the TI monitoring process makes it possible to form the structure of IT and present it as a set of IDEFX-models that explain how a set of heterogeneous (graphic, text, digital, cartographic, etc.) data about TI elements obtained from different sources are processed and presented to support decision-making on ways to improve it.

4 EXPERIMENTS

The possibility of using the proposed IT for TI monitoring is considered on the example of the Kharkiv district of the Kharkiv region near the villages of Rokytno, Pavlivka, Vatutine, Vilkhuvatka, Ordivka, Krynychky with a total population of 3602 people (Fig. 5).

The purpose of the experiment was to evaluate the possibility of using remote data in TI monitoring to solve the following problems:

- obtaining generalizing coefficients for the analysis of TI of a separate area for choosing directions for the strategic development of territories;
- inventorying the TI objects;
- modeling the TI and its elements;



Figure 5 – Study area (TI section) on the Sentinel-2 imagery

- researching the TI properties to substantiate design decisions during the reconstruction, development of the existing transport network;
- planning a procedure for surveying the condition of highways and other TI facilities.

The following were used as initial data:

- satellite images of the study area from Sentinel-2 and SuperView-1;
- table of decryption signs of TI objects;
- results of laser survey with a Leica RTC 360 scanner;
- results of tachometric survey in the SK-63 coordinate system using the Leica FlexLine TS03 device.

The implementation of proposed IT is possible by using the ArcGIS software suite by Esri. When validating the evaluations, they were compared for compliance with the results obtained from the statistics department of the Kharkiv region, which made it possible to conduct an experiment to study the effectiveness of IT in real conditions and on real objects.

As a result of the experiment, conclusions were drawn about the possibility of using remote data as auxiliary tools that complement the existing expert-visual and visual-instrumental methods for TI monitoring.

5 RESULTS

An analysis of a snapshot of the road infrastructure in the study area (Fig. 5) showed that it covers an area of $S=61.25 \text{ km}^2$, the total length of roads of various types is $L=17.8 \text{ km}$. Using formulas (2), (3), we find generalizing coefficients $K(E) \approx 0.038$ and $K(G) \approx 0.93$. To validate the obtained data, using statistical data, for the Kharkiv region we will obtain $S=31418 \text{ km}^2$, $L=9672.8 \text{ km}$, $H=2633834$ people, $N=1755$; hence the values of the coefficients for Kharkiv region are $K(E) \approx 0.034$ and $K(G) \approx 1.3$.

Fig. 6 shows the result of solving the problem of inventorying objects and TI cartographic modeling of the

study area: all TI objects were found and plotted on a cartographic base to fix spatial data, systematize and visualize the information received about the road transport infrastructure.

Artificial structures, and primarily bridges, are the most complex and expensive structures in the transport infrastructure [25]. The cartographic model (Fig. 6) shows several bridges that have been identified in the study area. Further, among the bridges presented in Fig. 6, a bridge over the river Mzha, located on the territory of the Rokytno village, which is on the local road that connects the M-18 and M-29 highways, was selected for analysis.

Traditionally, evaluation of the condition of bridges is based on visual inspections and surveys using geotechnical field instruments and geodetic equipment. Naturally, the results of such engineering surveys significantly depend on subjective factors, including the level of training of specialists [17, 25]. However, laser scanning technology is increasingly attracting the attention of inspectors and researchers in the field of bridge management, allowing to obtain detailed bridge geometry. Being a kind of remote methods, laser scanning of objects increases the reliability of the survey results by increasing the sample of measurements, and improves the quality of project documentation [5, 25].

The results of laser survey of the bridge obtained as initial data made it possible to build an information 3D-model of bridge structures (Fig. 7) with an accuracy of 1 mm. Its integration into a 3D relief model improves the perception of TI objects as natural elements located in real conditions of the natural environment of the study area.

6 DISCUSSION

The results of the experiment confirmed that, based on remote data, it is easy to find the generalizing Engel and Goltz coefficients for analyzing the transport component of the area. In the context of the ongoing administrative-territorial reform in Ukraine, when statistical data on the region may not be available, remote data becomes the only means of obtaining the necessary information. Note that the found values of the coefficients for the study area are comparable with the statistical values for the region. At the same time, the level of provision of the population with motor transport routes slightly exceeds (by 12.7%) the statistical data for the Kharkiv region, and attention should be paid to the development of the road network between villages: compared to the region, this indicator is lower by 28.7%.

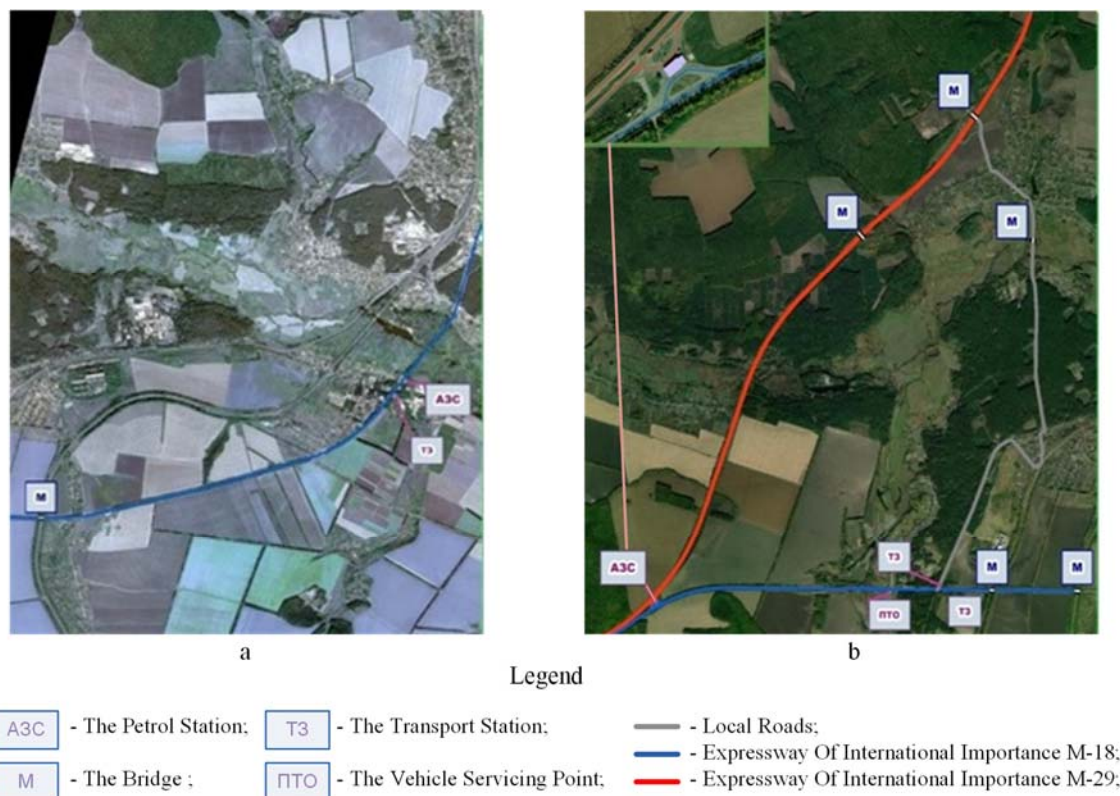


Figure 6 – An example of solving the problem of object inventory and TI cartographic modeling based on:
 a – satellite images from SuperView-1; b – satellite images from Sentinel-2

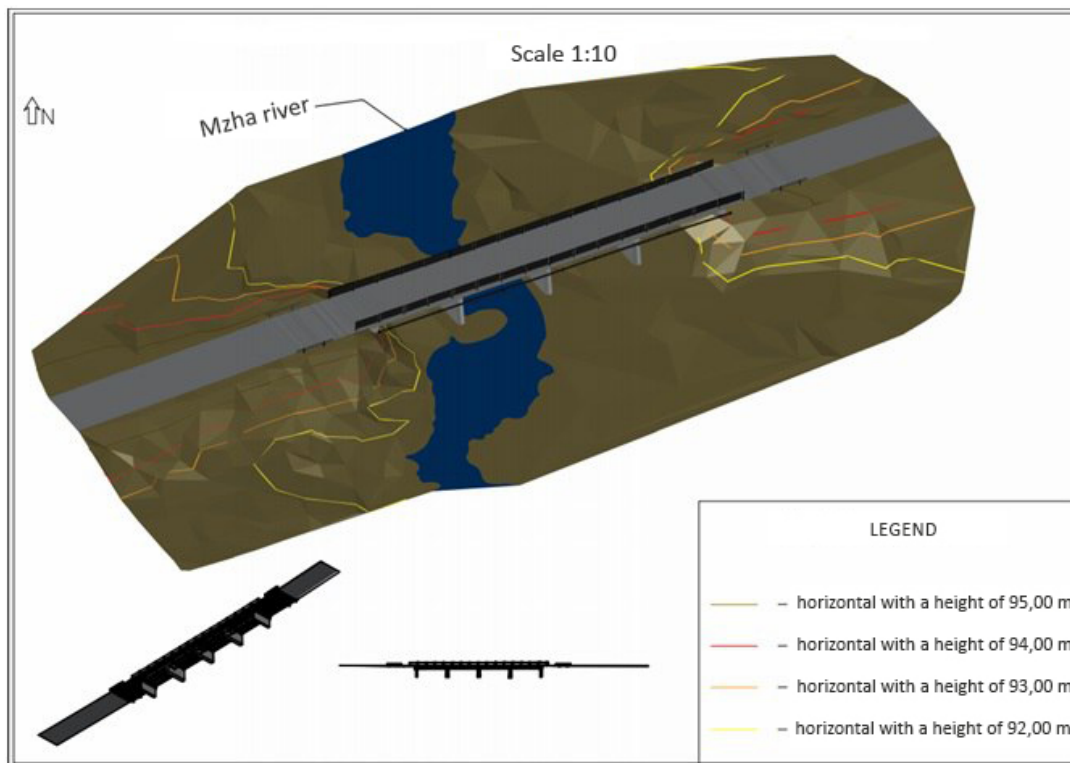


Figure 7 – An example of solving the problem of 3D-modeling of TI objects based on the results of laser scanning

The results of solving the problem of inventory of objects and cartographic modeling of the TI of the study area (Fig. 6) showed that the graphical interpretation of different types of objects and the possibility of combining different images on the model improves the visual perception of the condition of the TI section. At the same time, the high resolution of images does not affect the speed of their processing by ArcGIS.

Analyzing the image from the SuperView-1 satellite (Fig. 6a), we note that its resolution (0.5 m per pixel) makes it possible to apply the results of a visual inspection of the highways concrete pavement condition to the model using the gradation provided by the Building Code B.2.3-218-534:2011. This significantly expands the scope of the developed IT and in the future creates an opportunity for substantiating design solutions for the reconstruction of the transport network and planning procedures for examining its condition, which is also confirmed by the conclusions obtained in [17]. However, due to the lack of results of a real survey of TI in the study area, the scope of this study is limited only to evaluate the possibility of using the developed IT in solving such problems. It should be noted that the low-resolution, free images from the Sentinel-2 satellite (Fig. 6b) do not allow solving such problems. But the resulting cartographic models will help RMC to identify vulnerabilities in the investigated road infrastructure, facilitate decision-making when planning procedures for examining the condition of TI objects, etc.

Interesting prospects are opened by the integrated use of remote data and the results of engineering surveys of

TI objects, collected using modern geodetic equipment. The 3D-model of the bridge built based on the results of laser scanning makes it possible not only to obtain accurate design documentation, but also makes it possible to conduct a comprehensive analysis of the bridge, taking into account the physical and geographical location and predict its behavior without fieldwork. However, the lack of a reference model makes it difficult to solve the problem of evaluating the condition of the bridge: research is primarily focused on its representation as a whole, it lacks a clear linkage of changes to specific characteristics, which means that it is difficult to understand what data should be used as a reference for the condition evaluating for a certain period of time.

CONCLUSIONS

The actual problem of developing scientific and methodological support of information support for the process of monitoring the transport infrastructure in order to find ways to improve it in the implementation of development projects has been solved.

The scientific novelty of the obtained results is that the methodology for researching information processes has been further developed by adapting it to solve TI monitoring problems by refining the set-theoretic model of information flows process. Based on the requirements of current standards, taking into account global trends, a set of input and output parameters of the model, as well as a set of process operations, are formed. As a result of the consistent refinement of this model, for the first time, an information technology for monitoring the transport infrastructure was proposed, including a set of IDEFX-models

that explain how a set of heterogeneous (graphic, text, digital, cartographic, etc.) data about TI elements upcoming from different sources are processed and presented to support decision-making process on the survey and improvement of the existing infrastructure.

The practical significance of the obtained results is that the representation of the IT structure based on the IDEF functional modeling standard makes it easy to move to the creation of information monitoring systems for TI based on remote data. The conducted experiment on studying the capabilities of the proposed IT showed its effectiveness in solving the classical problems of complex analysis of TI based on generalizing coefficients, and also outlined the range of tasks where it can be used as an addition to the existing expert visual and visual-instrumental methods of TI monitoring. The results of the experiment make it possible to recommend the developed IT for use in practice, as well as to determine the effective conditions for its application.

Prospects for further research are the combining of remote data, survey results of TI sections and engineering surveys of objects on these sections to form evaluations of the TI condition. On the other hand, the direction associated with the refinement and expansion of the table of decryption signs for road transport infrastructure objects, as well as obtaining reference models of TI objects for monitoring the road surface and infrastructure as a whole, becomes interesting.

Author Contributions: Review and analysis of references, S. Andriev, S. Danshyna; development of conceptual provisions and methodology of research, S. Danshyna; validation, A. Nechausov; analysis of research results, A. Nechausov, S. Danshyna; data curation, S. Andriev; writing—original draft, S. Danshyna; writing—review and editing, A. Nechausov; project administration, S. Danshyna. All authors have read and agreed to the published version of the manuscript.

ACKNOWLEDGEMENTS

The work is supported by the state budget scientific research projects of National Aerospace University “KhAI” “Methodological bases of distributed systems for monitoring environmental objects creating” (state registration number 0122U002298), “Development of the preliminary design of basic model of multifunction family of unmanned aviation complexes of operational-tactical class” (state registration number 0121U109480) and with the support of the Regional Center of Space Monitoring of the Earth “Slobozhanshchina”.

REFERENCES

1. Skorobogatova O., Kuzmina-Merlino I. Transport Infrastructure Development Performance [Text], *Reliability and Statistics in Transportation and Communication “RelStat – 2017” : 17th International science conference*. Tianjin, 2017 : proceeding, 2017, Vol. 178, pp. 319–329. DOI: <https://doi.org/10.1016/j.proeng.2017.01.056>.
2. Readiness for the Future of Production Report 2018; ed. A. T. Kearney. Switzerland, 2018.
3. Danshyna S. Yu., Nechausov A. S. Solution of the problem of placing medical facilities in city development projects, *Radio*

- Electronics, Computer Science, Control*, 2020, No. 3 (54), pp. 138–149. DOI: <https://doi.org/10.15588/1607-3274-2020-3-12>.
4. Uskov V., Kharchenko O. Regulating the Development of Transport Infrastructure in Megacities of the Russian Federation, in *Proc. of Int. Scien. Siberian transport forum on Transportation Research*, 2021, Vol. 54, pp. 645–653. DOI: <https://doi.org/10.1016/j.trpro.2021.02.117>.
5. Gura D., Markovskii I., Khushn N., Rak I., and Pshidatok S. A Complex for Monitoring Transport Infrastructure Facilities Based on Video Surveillance Cameras and Laser Scanners, in *Proc. of Int. Scien. Siberian transport forum on Transportation Research*, 2021, Vol. 54, pp. 775–782. DOI: <https://doi.org/10.1016/j.trpro.2021.02.130>.
6. Sotnychenko L. L. Doslidzhennia stanu infrastrukturnoho zabezpechennia rehioniv Ukrainy, *Ekonomika i organizaciya upravlinnya*, 2014, No. 1 (17)–2(18), pp. 255–263.
7. Borisov A. I., Gnatyuk I. G. Assessment of Transport Accessibility of the Arctic Regions of the Republic of Sakha (Yakutia), *Transport Infrastructure: Territory Development and Sustainability “TITDS-XII” : XII International conference : proceeding*, 2022, Vol. 61, pp. 289–293. DOI: <https://doi.org/10.1016/j.trpro.2022.01.048>.
8. Burghardt K., Uhl J. H., Lerman K., Leyk S. Road network evolution in the urban and rural United States since 1900, *Computers, Environment and Urban Systems*, 2022, Vol. 95. Available at: <https://www.sciencedirect.com/science/article/abs/pii/S0198971522000473>. DOI: <https://doi.org/10.1016/j.compenvurbsys.2022.101803> (Accessed 21 June 2022).
9. Ahmadzai F., Lakshmana Rao K. M., Ulfat Sh. Assessment and modelling of urban road networks using Integrated Graph of Natural Road Network (a GIS-based approach), *Journal of Traffic and Transportation Engineering*, 2019, Vol. 8, Issue 1, pp. 109–125. DOI: <https://doi.org/10.1016/j.jum.2018.11.001>.
10. Shon H., Cho Ch.-S., Byon Y.-J., Lee J. Autonomous condition monitoring-based pavement management system, *Automation on Construction*, 2022, Vol. 138. Available at: <https://www.sciencedirect.com/science/article/abs/pii/S0926580522000954>. DOI: <https://doi.org/10.1016/j.autcon.2022.104222> (Accessed 29 June 2022).
11. Shtayat A., Moridpour S., Best B., Shroff A., Raol D. A review of monitoring systems of pavement condition in paved and unpaved roads, *Journal of Traffic and Transportation Engineering*, 2020, Vol. 7, Issue 5, pp. 629–638. DOI: <https://doi.org/10.1016/j.tte.2020.03.004>.
12. Tosti F., Gagliardi V., D’Amico F., Alani A. M. Transport infrastructure monitoring by data fusion of GPR and SAR imagery information, *Transportation Research : International science Siberian transport forum : proceeding*, 2020, Vol. 45, pp. 771–778. DOI: <https://doi.org/10.1016/j.trpro.2020.02.097>.
13. Rasol M., Pais J. C., Perez-Gracia V., Solla M., Fentul S., Fernandes F. M. GPR monitoring for road transport infrastructure: A systematic review and machine learning insights, *Construction and Building Materials*, 2022, Vol. 324. Available at: <https://www.sciencedirect.com/science/article/abs/pii/S0950061822003774>. DOI: <https://doi.org/10.1016/j.conbuildmat.2022.126686> (Accessed 27 June 2022).
14. Khoudeir M., Brochard J., Legeay V., Do M.-T. Roughness Characterization through 3D Textured Image Analysis: Contribution to the Study of Road Wear Level, *Computer-Aided Civil and Infrastructure Engineering*, 2004, Vol. 19, Issue 2, pp. 93–104. DOI: <https://doi.org/10.1111/j.1467-8667.2004.00340.x>.
15. Amarasiri S., Gunaratne M., Sarkar S. Use of Digital Image Modeling for Evaluation of Concrete Pavement Macrotecture and Wear, *Journal of Transportation Engineering*, 2012, Vol. 138, Issue 5, pp. 589–611. DOI: [https://doi.org/10.1016/\(ASCE\)TE.1943-5436.0000347](https://doi.org/10.1016/(ASCE)TE.1943-5436.0000347).

16. Butenko O., Horelyk S., Zynuk O. Geospatial Data Processing Characteristics for Environmental Monitoring Tasks, *Architecture Civil Engineering Environment*, 2020, No. 13 (1), pp. 103–114. DOI: <https://doi.org/10.21307/ACEE-2020-008>.
17. Ozden A., Faghri A., Li M., Tabrizi K. Evaluation of Synthetic Aperture Radar Satellite Remote Sensing for Pavement and Infrastructure Monitoring, *Procedia Engineering*, 2016, Vol. 145, pp. 752–759. DOI: <https://doi.org/10.1016/j.proeng.2016.04.098>.
18. Sporudi transportu. Avtomobil'ni dorogi, DBN V.2.3-4-2007. [Chinnij vid 2007-07-01]. Kyiv, Derzhbud Ukraïni, 2007, 85 p. (Galuzevij standart Ukraïni).
19. Sporudi transportu. Ocinyuvannya stanu betonogo pokryttya avtomobil'nix dorog: GBN V.2.3-218-534:2011. [Chinnij vid 2011-01-18]. Kyiv, Derzhavna sluzhba av-tomobil'nix Ukraïni, 2011, 24 p. (Galuzevij standart Ukraïni).
20. Özcan A.-H., Ünsalan C. Probabilistic object detection and shape extraction in remote sensing data, *Computer Vision and Image Understanding*, 2020, Vol. 195. Available at: <https://www.sciencedirect.com/science/article/abs/pii/S1077314220300357>. DOI: <https://doi.org/10.1016/j.cviu.2020.102953> (Accessed 27 June 2022).
21. Danshyna S. General approach to project material resources management, *Innovate technologies and scientific solutions for industry*, 2021, No. 1 (15), pp. 43–52. DOI: [10.30837/ITSSI.2021.15.043](https://doi.org/10.30837/ITSSI.2021.15.043).
22. Danshyna S., Fedorovich O., Djakons D. Formalization of the processes of projects for the development of high-tech enterprises, *Intelligent computer-integrated information technology in project and program management: collective monograph*, edited by I. Linde, I. Chumachenko, V. Timofeyev. Riga, ISMA University of Applied Science, 2020. pp. 23–38. DOI: <https://doi.org/10.30837/MMP.2020.023>.
23. Danshyna S., Nechausov A., Andrieiev S., Cheranovskiy V. Information technology for analysis of waste management objects infrastructure, *Radioelectronic and Computer Systems*, 2022, No. 2, pp. 97–107. DOI: <https://doi.org/10.32620/refs.2022.2.08>.
24. Butler K. A., Bahrami A., Esposito Ch., Hebron R. Conceptual models for coordinating the design of user work with the design of information systems, *Data & Knowledge Engineering*, 2000, Vol. 33, Issue 2, pp. 191–198. DOI: [https://doi.org/10.1016/S0169-023X\(99\)00051-8](https://doi.org/10.1016/S0169-023X(99)00051-8).
25. Oskouie1 P., Becerik-Gerber B. and Soibelman L. Automated measurement of highway retaining wall displacements using terrestrial laser scanners, *Automation in Construction*, 2016, Vol. 65, pp. 86–101. DOI: <https://doi.org/10.1016/j.autcon.2015.12.023>.

Received 19.08.2022.
Accepted 09.11.2022.

УДК [625.7:528.854]:004.9

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ МОНІТОРИНГУ ТРАСПОРТНОЇ ІНФРАСТРУКТУРИ НА ОСНОВІ ДАНИХ ДИСТАНЦІЙНОГО ЗОНДУВАННЯ

Даншина С. Ю. – д-р техн. наук, професор кафедри геоінформаційних технологій та космічного моніторингу Землі, Національний аерокосмічний університет «Харківський авіаційний інститут», Харків, Україна.

Нечаусов А. С. – канд. техн. наук, доцент кафедри геоінформаційних технологій та космічного моніторингу Землі, Національний аерокосмічний університет «Харківський авіаційний інститут», Харків, Україна.

Андрєєв С. М. – канд. техн. наук, доцент кафедри геоінформаційних технологій та космічного моніторингу Землі, Національний аерокосмічний університет «Харківський авіаційний інститут», Харків, Україна.

АНОТАЦІЯ

Актуальність. На тлі існуючої практики моніторингу автодорожньої мережі це дослідження спрямоване на вивчення можливостей технологій дистанційного зондування для вирішення завдань підвищення об'єктивності одержаних оцінок стану транспортної інфраструктури в цілому. Об'єктом дослідження являється процес моніторингу транспортної інфраструктури для пошуку шляхів її вдосконалення при реалізації проєктів розвитку. Мета роботи – шляхом наочного подання та візуалізації просторових даних моніторингу транспортної інфраструктури підвищити об'єктивність рішень, що приймаються, відносно планів обстеження, реконструкції та розвитку існуючої транспортної мережі.

Метод. Проаналізовано існуючі підходи до моніторингу транспортної інфраструктури (ТІ) та оцінювання її стану. Виділені недоліки, а також тенденції розвитку технологій дистанційного зондування відкривають перспективи з використання дистанційних даних у процесі моніторингу ТІ. Запропоновано теоретико-множинну модель інформаційних потоків процесу моніторингу, послідовне уточнення елементів якої дало змогу запропонувати інформаційну технологію (ІТ). Формування множин вхідних і вихідних параметрів ІТ, множини її операцій, подання їх у нотації IDEFX-моделей пояснює як сукупність різномірних (графічних, текстових, цифрових, картографічних тощо) даних про елементи ТІ, які надходять з різних джерел, обробляються та надаються для підтримки прийняття рішень щодо обстеження існуючої інфраструктури та її вдосконалення. Розроблена ІТ дає змогу отримати комплексні показники для аналізу ТІ окремого району, вирішувати завдання інвентаризації об'єктів інфраструктури, картографічного моделювання ТІ та її елементів з урахуванням фізико-географічного розташування, що дає змогу розглядати її як допоміжний засіб, що доповнює існуючі методи моніторингу ТІ.

Результати. Розроблена ІТ досліджена при вирішенні завдань моніторингу ТІ ділянки Харківського району з використанням супутникових знімків середньої (Sentinel-2) і високої (SuperView-1) роздільної здатності та результатів лазерної зйомки дорожнього мосту через р. Мжу (як елементу інфраструктури).

Висновки. Проведені експерименти підтверджують працездатність запропонованої ІТ і дають змогу рекомендувати її для використання на практиці при вирішенні завдань з отримання узагальнюючих характеристик інфраструктури, інвентаризації об'єктів ТІ та їх моделювання. Це відкриває можливості щодо обґрунтування проєктних рішень реконструкції транспортної мережі та планування процедур обстеження її стану. Перспективи подальших досліджень можуть полягати: у створенні еталонних моделей об'єктів ТІ, розширенні таблиць дешифрувальних ознак об'єктів дорожньо-транспортної інфраструктури, комплексуванні дистанційних даних, результатів обстеження ділянок ТІ й інженерних вишукувань об'єктів для отримання оцінок стану ТІ в цілому.

КЛЮЧОВІ СЛОВА: модель інформаційних потоків процесу, IDEFX-моделі, картографічне та 3D-моделювання.

ЛІТЕРАТУРА / LITERATURE

1. Skorobogatova, O. Transport Infrastructure Development Performance [Text] / O. Skorobogatova, I. Kuzmina-Merlino // Reliability and Statistics in Transportation and Communication "RelStat – 2017" : 17th International science conference, Tianjin, 2017 : proceeding. – 2017. – Vol. 178. – P. 319–329. DOI: <https://doi.org/10.1016/j.proeng.2017.01.056>.
2. Readiness for the Future of Production Report 2018 [Text] / ed. A. T. Kearney. – Switzerland, 2018. – 266 p.
3. Danshyna S. Yu. Solution of the problem of placing medical facilities in city development projects [Text] / S. Yu. Danshyna, A. S. Nechausov // Radio Electronics, Computer Science, Control. – 2020. – № 3 (54). – P. 138–149. DOI: <https://doi.org/10.15588/1607-3274-2020-3-12>.
4. Uskov V. Regulating the Development of Transport Infrastructure in Megacities of the Russian Federation [Text] / V. Uskov, O. Kharchenko // Transportation Research : International science Siberian transport forum : proceeding. – 2021. – Vol. 54. – P. 645–653. DOI: <https://doi.org/10.1016/j.trpro.2021.02.117>.
5. A Complex for Monitoring Transport Infrastructure Facilities Based on Video Surveillance Cameras and Laser Scanners [Text] / [D. Gura, I. Markovskii, N. Khusht et al.] // Transportation Research : International science Siberian transport forum : proceeding. – 2021. – Vol. 54. – P. 775–782. DOI: <https://doi.org/10.1016/j.trpro.2021.02.130>.
6. Сотниченко Л. Л. Дослідження стану інфраструктурного забезпечення регіонів України [Текст] / Л. Л. Сотниченко // Економіка і організація управління. – 2014. – № 1(17) – 2 (18). – С. 255–263.
7. Borisov A. I. Assessment of Transport Accessibility of the Arctic Regions of the Republic of Sakha (Yakutia) [Text] / A. I. Borisov, G. A. Gnatyuk // Transport Infrastructure: Territory Development and Sustainability "TITDS-XII" : XII International conference : proceeding. – 2022. – Vol. 61. – P. 289–293. DOI: <https://doi.org/10.1016/j.trpro.2022.01.048>.
8. Road network evolution in the urban and rural United States since 1900 [Electronic resource] / [K. Burghardt, J. H. Uhl, K. Lerman, S. Leyk // Computers, Environment and Urban Systems. – 2022. – Vol. 95. Access mode: <https://www.sciencedirect.com/science/article/abs/pii/S0198971522000473>. DOI: <https://doi.org/10.1016/j.compenvurbsys.2022.101803>
9. Ahmadzai, F. Assessment and modelling of urban road networks using Integrated Graph of Natural Road Network (a GIS-based approach) [Text] / F. Ahmadzai, K. M. Lakshmana Rao, Sh. Ulfat // Journal of Traffic and Transportation Engineering. – 2019. – Vol. 8, Issue 1. – P. 109 – 125. DOI: <https://doi.org/10.1016/j.jum.2018.11.001>.
10. Autonomous condition monitoring-based pavement management system [Electronic resource] / H. Shon, Ch.-S. Cho, Y.-J. Byon, J. Lee // Automation on Construction. – 2022. – Vol. 138. – Access mode: <https://www.sciencedirect.com/science/article/abs/pii/S0926580522000954>. DOI: <https://doi.org/10.1016/j.autcon.2022.104222>.
11. A review of monitoring systems of pavement condition in paved and unpaved roads [Text] / [A. Shtayat, S. Moridpour, B. Best et al.] // Journal of Traffic and Transportation Engineering. – 2020. – Vol. 7, Issue 5. – P. 629 – 638. DOI: <https://doi.org/10.1016/j.tte.2020.03.004>.
12. Transport infrastructure monitoring by data fusion of GPR and SAR imagery information [Text] / F. Tosti, V. Gagliardi, F. D'Amico, A. M. Alani // Transportation Research : International science Siberian transport forum : proceeding. – 2020. – Vol. 45. – P. 771 – 778. DOI: <https://doi.org/10.1016/j.trpro.2020.02.097>.
13. GPR monitoring for road transport infrastructure: A systematic review and machine learning insights [Electronic resource] / [M. Rasol, J. C. Pais, V. Perez-Gracia et al.] // Construction and Building Materials. – 2022. – Vol. 324. – Access mode: <https://www.sciencedirect.com/science/article/abs/pii/S0950061822003774>. DOI: <https://doi.org/10.1016/j.conbuildmat.2022.126686>.
14. Roughness Characterization through 3D Textured Image Analysis: Contribution to the Study of Road Wear Level [Text] / M. Khoudeir, J. Brochard, V. Legeay, M.-T. Do // Computer-Aided Civil and Infrastructure Engineering. – 2004. – Vol. 19, Issue 2. – P. 93 – 104. DOI: <https://doi.org/10.1111/j.1467-8667.2004.00340.x>.
15. Amarasiri, S. Use of Digital Image Modeling for Evaluation of Concrete Pavement Macrotexture and Wear [Text] / S. Amarasiri, M. Gunaratne, S. Sarkar // Journal of Transportation Engineering. – 2012. – Vol. 138, Issue 5. – P. 589 – 611. DOI: [https://doi.org/10.1016/\(ASCE\)TE.1943-5436.0000347](https://doi.org/10.1016/(ASCE)TE.1943-5436.0000347).
16. Butenko, O. Geospatial Data Processing Characteristics for Environmental Monitoring Tasks [Text] / O. Butenko, S. Horelyk, O. Zynuk // Architecture Civil Engineering Environment. – 2020. – № 13 (1). – P. 103 – 114. DOI: <https://doi.org/10.21307/ACEE-2020-008>.
17. Evaluation of Synthetic Aperture Radar Satellite Remote Sensing for Pavement and Infrastructure Monitoring [Text] / A. Ozden, A. Faghri, M. Li, K. Tabrizi // Procedia Engineering. – 2016. – Vol. 145. – P. 752–759. DOI: <https://doi.org/10.1016/j.proeng.2016.04.098>.
18. Споруди транспорту. Автомобільні дороги : ДБН В.2.3-4-2007. – [Чинний від 2007-07-01]. – К. : Держбуд України, 2007. – 85 с. – (Галузевий стандарт України).
19. Споруди транспорту. Оцінювання стану бетонного покриття автомобільних доріг: ГБН В.2.3-218-534:2011. – [Чинний від 2011-01-18]. – К. : Державна служба автомобільних України, 2011. – 24 с. – (Галузевий стандарт України).
20. Özcan A.-H. Probabilistic object detection and shape extraction in remote sensing data [Electronic resource] / A.-H. Özcan, C. Ünsalan // Computer Vision and Image Understanding. – 2020. – Vol. 195. – Access mode: <https://www.sciencedirect.com/science/article/abs/pii/S1077314220300357>. DOI: <https://doi.org/10.1016/j.cviu.2020.102953>.
21. Danshyna S. General approach to project material resources management [Text] / S. Danshyna // Innovate technologies and scientific solutions for industry. – 2021. – № 1 (15). – С. 43–52. DOI: [10.30837/ITSSI.2021.15.043](https://doi.org/10.30837/ITSSI.2021.15.043).
22. Danshyna S. Formalization of the processes of projects for the development of high-tech enterprises [Text] / S. Danshyna, O. Fedorovich, D. Djakons // Intelligent computer-integrated information technology in project and program management: collective monograph, edited by I. Linde, I. Chumachenko, V. Timofeyev. – Riga: ISMA University of Applied Science, 2020. – P. 23–38. DOI: <https://doi.org/10.30837/MMP.2020.023>.
23. Information technology for analysis of waste management objects infrastructure [Text] / [S. Danshyna, A. Nechausov, S. Andrieiev, V. Cheranovskiy] // Radioelectronic and Computer Systems. – 2022. – № 2. – P. 97–107. DOI: <https://doi.org/10.32620/reks.2022.2.08>.
24. Conceptual models for coordinating the design of user work with the design of information systems [Text] / [K. A. Butler, A. Bahrami, Ch. Esposito, R. Hebron] // Data & Knowledge Engineering. – 2000. – Vol. 33, Issue 2. – P. 191–198. DOI: [https://doi.org/10.1016/S0169-023X\(99\)00051-8](https://doi.org/10.1016/S0169-023X(99)00051-8).
25. Oskouie1 P. Automated measurement of highway retaining wall displacements using terrestrial laser scanners [Text] / P. Oskouie1, B. Becerik-Gerber, L. Soibelman // Automation in Construction. – 2016. – Vol. 65. – P. 86–101. DOI: <https://doi.org/10.1016/j.autcon.2015.12.023>.

DETERMINATION OF INHERITANCE RELATIONS AND RESTRUCTURING OF SOFTWARE CLASS MODELS IN THE PROCESS OF DEVELOPING INFORMATION SYSTEMS

Kungurtsev O. B. – PhD, Professor of the Software Engineering Department, Odessa Polytechnic National University, Odessa, Ukraine.

Vytnova A. I. – Student of the Software Engineering Department, Odessa Polytechnic National University, Odessa, Ukraine.

ABSTRACT

Context. The implementation of different use-cases may be performed by different development teams at different times. This results in a poorly structured code. The problem is exacerbated when developing medium and large projects in a short time.

Objective. Since inheritance is one of the effective ways to structure and improve the quality of code, the aim of the study is to determine possible inheritance relationships for a variety of class models.

Method. It is proposed to select from the entire set of classes representing the class model at a certain design stage, subsets for which a common parent class (in a particular case, an abstract class) is possible. To solve the problem, signs of the generality of classes have been formulated. The mathematical model of the conceptual class has been improved by including information about the responsibilities of the class, its methods and attributes. The connection of each class with the script items for which it is used has been established. A system of data types for class model elements is proposed. Description of class method signatures has been extended. A method for restructuring the class model, which involves 3 stages, has been developed. At the first stage, the proximity coefficients of classes are determined. At the second, subsets of possible child classes are created. At the third stage, an automated transformation of the class structure is performed, considering the identified inheritance relationships.

Results. A software product for conducting experiments to identify possible inheritance relationships depending on the number of classes and the degree of their similarity has been developed. The results of the conducted tests showed the effectiveness of the decisions made.

Conclusions. The method uses an algorithm for forming subsets of classes that can have one parent and an algorithm for automatically creating and converting classes to build a two-level class hierarchy. An experiment showed a threefold reduction in errors in detecting inheritance and a multiple reduction in time in comparison with the existing technology.

KEYWORDS: class model, class attribute, class method, data types, use case, inheritance.

ABBREVIATIONS

UC is a use-case;
OOP is an object-oriented programming;
OOA is an object-oriented analysis;
SP is a software product;
OOT is an object-oriented technologies.

NOMENCLATURE

Cp_i is a parent class;
 C_{jq} is q -th descendant class that passed common attributes and methods to the parent class;
 $cHead$ is a class header;
 $mMeth$ is a set of functions (methods) of the class;
 $mAttr$ is a set of class attributes;
 $cName$ is a name of the class;
 $mResp$ is a set of class responsibilities;
 $uName$ is the name of the UC and the number of the point where the class was created, or a function was added to the class;
 nP is a class responsibility, represented by a single phrase;
 $abstract$ is an abstract class;
 $cName1$ is a name of the parent class for the $cName$ class;
 $mChildCl$ is a set of child classes (filled only for an abstract class);
 $Numb$ is a number format;

$Bool$ is a boolean value;
 $Text$ is any text;
 $Void$ is a function does not return the value;
 $NameS$ is a name of the type;
 $NameFi$ and $Typei$ are the name and type of the i -th field;
 $NameL$ is a name of the type;
 $NameE$ is a name of the list element;
 $CPName$ is a type name (class name).
 $attrName$ is an identifier of the attribute;
 $attrResp$ is an attribute responsibility;
 $attrType$ is an attribute type;
 $fName$ is a name of the method;
 $fRespo$ is a responsibility of the method;
 mRC_i is a set of class C_i responsibilities;
 $C_i mRC_i$ is a number of class C_i responsibilities;
 mRC_j is a set of class C_j responsibilities;
 $C_j mRC_j$ is a number of class C_j responsibilities;
 $mArgs$ is a set of method arguments;
 $returnVal$ is a function return value;
 $mRsArgs$ is a set of arguments that return the result of the calculation;
 CA_i is an abstract class;
 $mChildC_i$ is the set of its child classes;
 CAS is the concatenation of the names of all classes that are included in the set $mChildC_r$ (hereinafter, the name is edited by the expert).

INTRODUCTION

The theory of OOP and OOA was elaborated in detail in the works of G. Booch and his colleagues [1] and continues to be developed and promoted [2, 3]. However, the practice of applying theoretical principles in the development of SPs faces many unsolved problems. The use of flexible technologies significantly speeds up the process of designing software products [4], however, it is possible to perform OOA to full extent only within the framework of the cascade model of the software life cycle [5]. In most OOT for creating software products functional requirements are written in the form of use cases (UC) [6]. UML is used to create UC diagrams, interaction diagrams, and class specifications. Stages of compiling the text of UC, class analysis, defining possible hierarchical relationships between them are not usually supported by design tools. The implementation of all main design stages within one iteration, which is typical for flexible technologies, allows carrying out a detailed OOA only for some fragments of the subject area. This creates a number of problems for the project [7], including defects in the architecture and structure of the class model. As a result, the program code requires detailed refactoring [8]. This is especially evident for medium and large projects, when teams of developers work in parallel to solve different problems (Fig. 1.). Under such conditions, there is a high probability that a possible “kinship” between classes will go unnoticed or will not cover all potential members of the hierarchy.

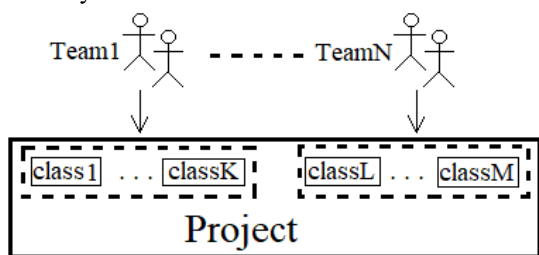


Figure 1 – Parallel development of the class structure

The purpose of the study is to select from the set of classes that represent the class model at a certain design stage, subsets for which a common parent class and automated restructuring of all classes related by inheritance relations are possible.

To achieve the stated goal, it is necessary to solve the following tasks:

- To formulate signs of class commonality;
- To improve the class model in order to provide comparison with other classes;
- To develop a method for restructuring the class model taking into account inheritance;
- To perform approbation of the research results.

1 PROBLEM STATEMENT

Let $mC = \{C_1, C_2, \dots, C_i, \dots, C_n\}$ be the set of class models of some software project. It is necessary to extract from mC such subsets of classes mC_1, mC_2, \dots, mC_k , for which common parent classes can be created. If some

subset $mC_j = \{C_{j1}, C_{j2}, \dots, C_{jq}\}$ is found, then it is transformed to the form $\langle Cp_j \{C'_{j1}, C'_{j2}, \dots, C'_{jq}\} \rangle$.

2 REVIEW OF THE LITERATURE

A good practical guide to inheritance is provided by [9], but it does not address the issue of inheritance of classes represented by models. In [10], it is proposed to put an abstract class as the basis of the hierarchy. It is shown that the effect of using an abstract class occurs when a number of subclasses are created on its basis in accordance with different specializations of the tasks being solved. However, the question of finding these specializations remains open. Disadvantages in the representation of classes in UML models are noted in [11]. The author suggests deepening your understanding of object-oriented concepts by determining relationships between actions and attributes, without considering the similarity of classes in terms of actions and attributes. In [12], the problem of the transition from the class model to the domain ontology is considered. An extension of the representation of classes, which, however, does not affect the identification of inheritance relations, is proposed.

In [13], the modularization of object-oriented software systems is proposed, considering the connectivity, concatenation, index of the number and sizes of packages. The said principles of restructuring at the package level can be partly transferred to the class level.

The work [14] is devoted to the analysis of software quality at three levels. At the class level, it is proposed to introduce additional quality assessment metrics. However, they do not provide an assessment of the existing or possible hierarchical relationships between classes.

In [15], a two-level clustering of class models is proposed: at the level of semantics and structure. Obviously, this approach makes it possible to select “similar” classes. However, the analysis of the possible “kinship” between such classes was not performed in the work. A similar problem of determining groups of “close” classes was solved in [16]. But here the aim was to reduce testing resources, not to restructure classes.

The question of the comparative efficiency of manual and automated search for features of functions was considered in [17]. The idea of organizing the search for features not only in the code, but also in models is very productive.

The analysis of hierarchical relations of classes was performed in [18]. However, it is not the process of forming a hierarchical structure that is being studied, but its analysis for the purpose of preserving secret information in inherited methods.

In [19], a method for automated description of UC was proposed, which made it possible to further automate the process of building a model of conceptual classes [20]. At the same time, additional information about the connection of the class with the UC, methods and attributes of the class was placed in the model. Such a model [20] contains more information for searching for

class “kinship”, but without significant development it cannot solve such a problem.

3 MATERIALS AND METHODS

Let start with an **improved model class**. In [20], a class model is proposed that can be taken as a basis. However, the specific task of finding a set of classes that can have a common “parent” requires a significant development of the said model. Let us formulate new requirements for the model:

- the class header is a comparison element. It must have the characteristic of responsibility.
- the class attribute is a comparison element. It must have the characteristic of responsibility and type;
- the class method is a comparison element. It must be represented by a responsibility and a signature;
- a class must have characteristics that define its role and relationships in the class hierarchy.

Basing on the foregoing, we will represent all the classes that are included into the project as a set:

$$mC = \{c\}, \quad (1)$$

and each class as a tuple:

$$c = \langle cHead, mMeth, mAttr \rangle. \quad (2)$$

Now let’s talk about a **class header**. To compare classes, it is proposed to introduce a set of responsibilities for which the class is used, formulated as separate sentences in the header of the class. In accordance with the technology of constructing a class model [21], a class is created when the UC “Create” item is implemented in the class model. At the same time, the first responsibility proposal is formed. For each subsequent point in the script, when the class must perform an action, a responsibility for the corresponding function, which is included into the set of class responsibilities is formed. For a possible tracing from the class model to the requirements (scenarios), the name of the corresponding UC and the number of the scenario item correspond to each new responsibility.

Further we will consider parent classes as abstract ones, since in our case they will not generally represent real objects of the subject area. Thus, the class header is represented as a tuple:

$$cHead = \langle cName, mResp, inheritance \rangle. \quad (3)$$

Each element of the set $mResp$ is represented by a tuple

$$\langle uName, nP, r \rangle. \quad (4)$$

An inheritance relationship is represented by a tuple:

$$\langle inheritTrait, mChildCl \rangle, \quad (5)$$

where $inheritTrait$ can take the following values: abstract, $cName1$, null (the class has no inheritance relationship with other classes), $mChildCl$.

In [20], a system of **data types** for a class model is proposed. In this work, this system has been developed at the expense of structured types.

Simple types: Numb, Bool, Text, Void.

Structured types. Struct – structure, in the general case, contains several fields of different types. The structure declaration has the following form:

$$Struct > NameS(n)(NameF1:Type1, NameF2:Type2, \dots NameK:TypeK).$$

A List can represent a linear list, an array, a set, and so on.

The list declaration looks as:

$$List > NameL(NameE:Type).$$

The declaration of a reference to an object of the $CPType$ class looks as:

$$CPType > CPName.$$

To provide the ability to compare **class attributes** it is proposed: to introduce the concept of the purpose (responsibility) of an attribute and data types.

As a result, each attribute from the set $mAttr$ will be presented as:

$$Attr = \langle attrName, attrResp, attrType \rangle. \quad (6)$$

To provide the possibility of comparing **class methods**, it is proposed: for each method to formulate its obligation in the form of a short phrase, for instance, “calculation of the cost of the order”; for method arguments to use the rules formulated earlier for attributes.

As a result, each method from set $mMeth(2)$ will take the form:

$$func = \langle fName, fResp, mArgs, returnVal, mRsArgs \rangle. \quad (7)$$

Figure 2 illustrates the resulting class model.

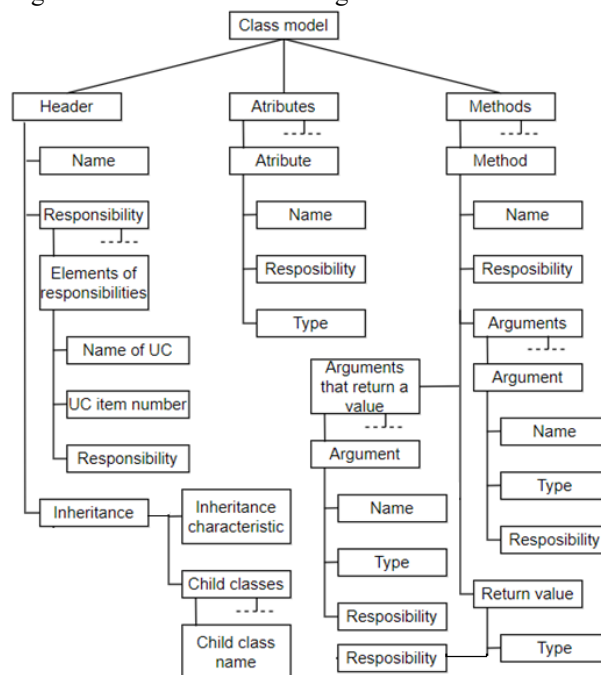


Figure 2 – The structure of the class model

The **class model restructuring method** involves four steps.

The first step is to determine the proximity of classes. Comparing two classes involves comparing class responsibilities, methods, and class attributes. To do this, it is necessary to compare various elements of the description of one class with other classes within the framework of the program class model, represented by a set mC (the total number of classes $nC=|mC|$).

For each comparison position, it is proposed to calculate the proximity coefficient

$$K = \frac{\text{Number_of_matching_elements}}{\text{Total_number_of_elements}}. \quad (8)$$

When comparing the elements represented by the text, fuzzy string comparison functions were used [22]. Therefore, the result of the comparison will be a number not exceeding 1. A threshold value of the coefficient of proximity of responsibilities of the class K_{cmin} has been introduced, below which it makes no sense to search for the “kinship” of classes.

To compare the responsibilities of classes, we transform the set of responsibilities $mResp_i$ (3) of a certain class C_i , excluding references to the UC and the scenario item.

$$mResp_i \Rightarrow mRC_i, \quad (9)$$

where $C_i mRC_i = \{r_{i,1}, \dots, r_{i,n}\}$, $C_i nRC_i = |mRC_i|$, $mRC_j = \{r_{j,1}, \dots, r_{j,m}\}$ and $C_j mRC_j = \{r_{j,1}, \dots, r_{j,n}\}$, $nRC_j = |mRC_j|$.

Let us define a set of overlapping responsibilities of classes C_i and C_j

$$mRC_{i,j} = \{ro_q \mid ro_q = r_{i,j} \wedge ro_q = r_{j,p}\}, \quad (10)$$

and their number

$$nRC_{i,j} = |mRC_{i,j}|. \quad (11)$$

If $nRC_{i,j} = 0$, then class comparison stops.

When comparing class methods, we proceed from the following considerations. Each time when a class is used to implement a script item, a responsibility is added to the class header. The same responsibility is attributed to the class function that implements it in the script item. To determine the identity of two functions with overlapping responsibilities from classes C_i and C_j , to match of all elements from (7) except the function names is required. Let us represent the set of coinciding functions of classes C_i and C_j in the form $mMethC_{i,j}$. If no match is found for a pair of functions, then $nRC_{i,j}$ is reduced by one.

Match of class attributes does not affect the assessment of class proximity degree, because there are methods that do not use the attributes of their class. However, matching attributes must be identified for further class transformation. To determine the identity of two attributes from classes C_i and C_j , their types and responsibilities must match. Let us represent the set of matching attributes of classes C_i and C_j in the form $mAttrC_{i,j}$.

The result of comparing two classes is called the proximity coefficient of the said classes ER . Its value must be different for classes C_i and C_j . For a class C_i :

$$ER_{i,j} = \frac{nRC_{i,j}}{nRC_i}. \quad (12)$$

For a class C_j :

$$ER_{j,i} = \frac{nRC_{i,j}}{nRC_j}. \quad (13)$$

The overall coefficient:

$$ERO_{i,j} = \frac{ER_{i,j} + ER_{j,i}}{2}. \quad (14)$$

The second stage is the construction of the class proximity matrix. To identify the possible “kinship” of classes from set $mC(1)$, it is proposed to use the matrix of class proximity. An example of such a matrix is presented in Table 1.

Table 1 – Matrix of class proximity

Classes	C1	C2	C3	C4	C5	C6	C7	C8	C9
C1	X	0	ER _{1,3}	0	0	ER _{1,6}	0	ER _{1,8}	0
C2	0	X	ER _{2,3}	0	ER _{2,5}	ER _{2,6}	0	0	0
C3	ER _{3,1}	ER _{3,2}	X	ER _{3,4}	0	0	0	0	ER _{3,9}
C4	0	0	ER _{4,3}	X	ER _{4,5}	ER _{4,6}	0	0	0
C5	0	ER _{5,2}	0	ER _{5,4}	X	0	0	0	0
C6	ER _{6,1}	ER _{6,2}	0	ER _{6,4}	0	X	0	ER _{6,8}	0
C7	0	0	0	0	0	0	X	0	0
C8	ER _{8,1}	0	0	0	0	ER _{8,6}	0	X	ER _{8,9}
C9	0	0	ER _{9,3}	0	0	0	0	ER _{9,8}	X

The cells of the matrix contain the values of the proximity coefficients for all pairs of classes from set mC . For instance, it follows from the matrix that there is no commonality between classes C_1 and C_2 , but there is a commonality between classes C_2 and C_5 .

The presence of commonality between class C_1 and classes C_3, C_6, C_8 does not mean that there will be one parent class for all these classes. The search for the optimal solution will consist in the fact that for any class from group C_3, C_6, C_8 , the condition for combining with C_1 is the greatest value of proximity with this particular class. For example, C_6 will enter a group with C_1 if $ER_{1,6} = \max(ER_{6,2}, ER_{6,4}, ER_{6,8})$.

The third stage is the formation of a set of abstract classes. At this stage, as a result of processing the matrix, it is necessary to form a set of abstract (parent) classes mCA (initially, the set is empty), each element of which has the form

$$mCA_i = \langle CA_i, mChildC_i \rangle. \quad (15)$$

Previously, we will place classes that can potentially become child classes in the set of child classes. Let us denote such a set $mChildC'$. The sequence of operations for the formation of the said sets is represented by the algorithm for identifying parent (abstract) classes:

1. To define the set of all classes and the set mC of abstract classes c .

2. To fill in the generality matrix of the size $K \times K$, where $K = |mC|$. To set the matrix row index $i=1$ and the abstract class index $r=1$.

3. For each proximity coefficient $ER_{j,n} \neq 0$, to calculate the total proximity coefficients $ERO_{j,n}$ for $j = i+1, K$.

4. If some $ERO_{j,n} > ERO_{i,n}$ is found, then $ERO_{i,n}$ is reset to zero. Otherwise, all $ERO_{j,n}$ are set to zero. If there is no more than one ERO in the current line, then go to step 6.

5. The set $mChildC'_r$ contains all classes of the i -th row for which $ER_{i,n} \neq 0$. Only the name of the abstract class is entered as CA_r . $cName = CAS$. To increase index r by 1.

6. To increase index i by 1. If $i < K$, go to step 3.

7. Completion of the algorithm.

The fourth stage is the formation of parent (abstract) and child classes. For each abstract class with the name CAS_r , it is necessary to form a header, methods and attributes using a set $mChildC'_r$ of classes. Each class in the

set $mChildC'_r$ must be converted into a derived class CAS_r by changing the header, excluding methods and attributes that passed into CAS_r .

The solution to this problem is formulated as a class restructuring algorithm:

1. We determine the possible number of abstract classes $Ka = |mCA|$ and set the index of the first abstract class $i=1$.

2. We determine the number of possible child classes for the i -th abstract from $Kc_i = |mCA_i.mChildC'_i|$ and define the responsibilities $mResp_i$ of an abstract class CAS_i by identifying, in accordance with (10), the general responsibilities of classes from $mChildC'_i$. We write in the inheritance relation $inheritTrait_i = abstract$, in the set $mChildC'_i$ we write the names of classes from $mChildC'_i$.

3. We determine methods $mMethCA_i$ of an abstract class CAS_i by identifying common methods of classes from $mChildC'_i$.

4. We determine the attributes $mAttrCA_i$ of an abstract class CAS_i by identifying common attributes of classes from $mChildC'_i$.

5. We set the index of the child class $j=1$.

6. In the class header $c_{i,j} \in mCA_i.mChildC'_j$, we set the inheritance flag $c_j.cHead.inheritTrait = CAS_j$.

7. We remove methods of class CAS_i $c_{i,j}$ from the class $c_{i,j}.mMeth := c_{i,j}.mMeth \cap CAS_i.mMeth$.

8. We remove attributes of class CAS_i $c_{i,j}$ from the class $c_{i,j}.mAttr := c_{i,j}.mAttr \cap CAS_i.mAttr$.

9. We set $j:=j+1$. If $j \leq Kc_i$, then go to step 6. Otherwise, go to step 7.

10. We demonstrate the analytics of the abstract class CAS_r and its child classes $mCA_i.mChildC'$. If inheritance is asserted, then each class from mC for which $mC.cName = mChildC_{i,j}.cName$ is replaced by the corresponding class $mChildC_{i,j}$ and an abstract class named CAS_i is added to the set mC .

11. We set $i:=i+1$. If $i \leq Ka$, then go to step 2. Otherwise, finish the algorithm.

4 EXPERIMENTS

In accordance with [21], a simplified scheme for constructing a class model is shown in Fig. 3.

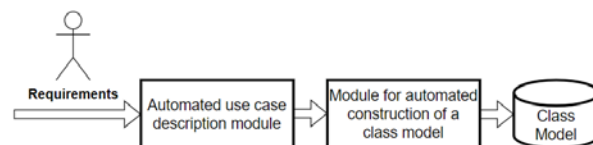


Figure 3 – Simplified scheme for building a class model

The system analyst, basing on the analysis of the subject area and consultations with an expert describes the UC using the UseCaseEditor program [20]. Basing on the obtained UCs, a programmer (perhaps a system analyst) creates a class model using the ModelEditor program [21].

To apply the proposed method of restructuring the class model, a software product HeirClass+ was developed.

Within the framework shown in Fig. 3 the technology (working mode), it is difficult to test the method of searching for inheritance relations, since it is impossible to select such UCs that would provide many classes in the model with the necessary characteristics in advance. Therefore, to test the decisions made, a software module was developed that allows you to create a class model bypassing the stage of automated UC description (experimental mode). Fig. 4 shows the class model restructuring scheme in an experimental and operational modes.

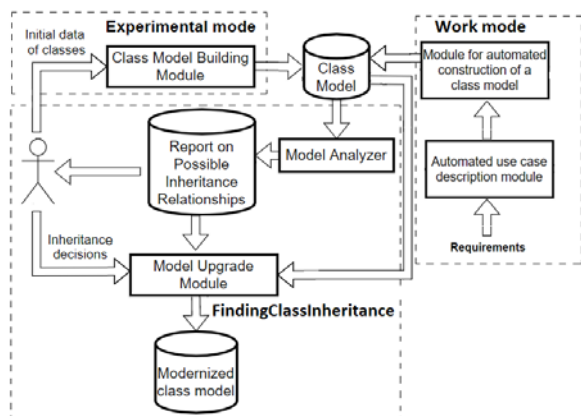


Figure 4 – Scheme for testing the decisions made and testing modules

For performing experiments 15 programmers (3rd year students) were involved. Of these, 5 teams were formed. For each team, requirements to 4 classes were formulated in the following form: “The class must perform The class contains a method that, basing on ..., returns The class contains an attribute that represents...”. The requirements were distributed in such a way that one team could not be given the task of describing potentially related classes. It was supposed that, in accordance with the requirements, there could be 6 groups of “related” classes.

5 RESULTS

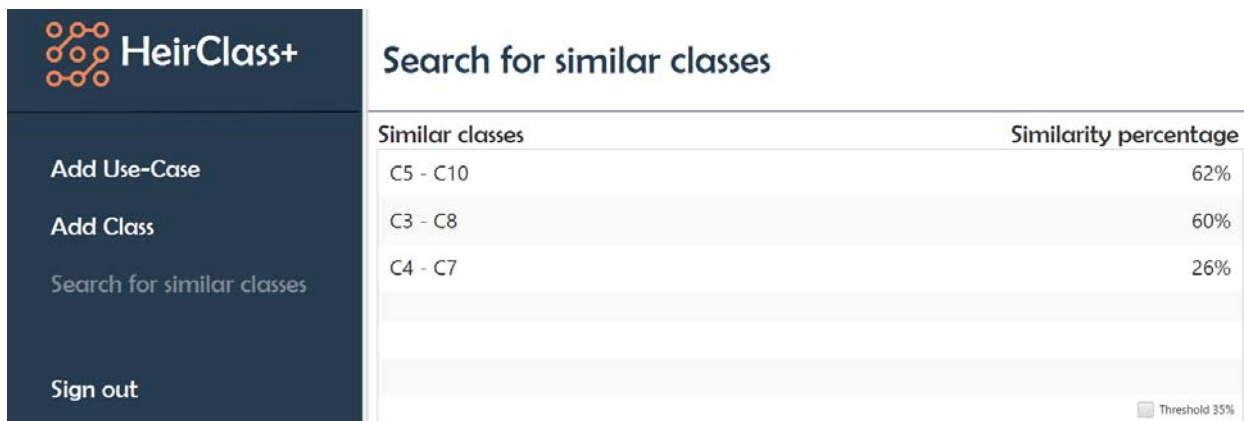
After completing the work on the models, the participants in the experiment were asked to identify potential inheritance relationships in a variety of classes. Simultaneously the program HeirClass+ with identical source data was started. After 2 hours, the teams identified 8 class groups with signs of inheritance relationships out of 9 supposed ones. Of these, 4 groups were accepted for restructuring. Program HeirClass+ identified 8 groups within 10 seconds. Of these, 5 groups were accepted for restructuring at a threshold commonality rate of 35%. In addition, HeirClass+ performed the restructuring flawlessly.

Table 2 shows the matrix of generality for the first 10 classes, obtained on the basis of the work of the program HeirClass+ (classes named C1-C10 for brevity).

Figure 5 presents a piece of information that is offered to the developer for deciding about inheritance.

Table 2 – Class commonality matrix (experiment)

Classes	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
C1	X	0%	29%	0%	43%	57%	29%	29%	43%	14%
C2	0%	X	38%	0%	38%	13%	0%	38%	13%	50%
C3	20%	30%	X	10%	10%	30%	0%	50%	0%	40%
C4	0%	0%	17%	X	0%	0%	33%	0%	17%	0%
C5	38%	38%	13%	0%	X	13%	13%	25%	50%	63%
C6	14%	57%	43%	0%	14%	X	0%	43%	14%	14%
C7	20%	0%	0%	20%	10%	0%	X	0%	20%	0%
C8	29%	43%	71%	0%	29%	43%	0%	X	14%	29%
C9	30%	10%	0%	10%	40%	10%	20%	10%	X	20%
C10	13%	50%	50%	0%	63%	13%	0%	25%	25%	X



Similar classes	Similarity percentage
C5 - C10	62%
C3 - C8	60%
C4 - C7	26%

Threshold 35%

Figure 5 – Class comparison result

6 DISCUSSION

Until now, class inheritance has been studied in terms of analyzing the effectiveness of its application [10], building class libraries, developing conditions and recommendations for specializing generated classes [18]. In this work, for the first time, the problem of automated search for possible inheritance relations and their implementation for a set of classes is solved. Class conversion automation is used in refactoring [8]. However, for refactoring, the object of modernization is the code, and the operations are initiated by a specialist.

From what has been said, it follows that the proposed method can only be compared with “manual” processing of a set of classes. The experiment showed that automated analysis was performed hundreds of times faster than manual analysis with a significant reduction in the number of errors, and class conversion turned out to be error-free.

CONCLUSIONS

It is shown that modern iterative software development technologies lead to the creation of a poorly structured code, which requires refactoring at relatively late stages of software design and is associated with high costs.

The paper solves the problem of automated determination of inheritance relations for a set of classes. For this purpose, signs of the generality of classes have been formulated; the class model has been improved by defining the concept of responsibility class, method, attribute; detailed description of the method signature has been given; a data type system for the class model has been proposed.

A method for restructuring the class model has been developed. The method uses an algorithm for forming subsets of classes that can have one parent and an algorithm for automatically creating and converting classes to build a two-level class hierarchy.

The results of the study are implemented in the HeirClass+ software product. An experiment using HeirClass+ showed a threefold reduction in errors in detecting inheritance and a multiple reduction in time in comparison with the existing technology.

REFERENCES

1. Booch G., Maksimchuk R. A., Engle M. W., Young B. J., Conallen J., Houston K. A., Wesley A. Object-Oriented Analysis and Design with Applications 3rd Edition. Boston, Addison-Wesley Professional, 2007, 694 p.
2. Lee G. Modern Programming: Object Oriented Programming and Best Practices. Birmingham, Packt, 2019, 266 p.
3. Baesens B., Backiel A., Broucke S. Beginning Java Programming: The Object-Oriented Approach. Birmingham, Wrox, 2015, 672 p.
4. Brand M., Tiberius V., Bican P. M., Brem A. Agility as an innovation driver: towards an agile front end of innovation framework. Potsdam, Springer, 2021, pp. 157–187.
5. Adeagbo M. A., Akinsola J., Awoseyi A. A., Kasali F. Project Implementation Decision Using Software Development Life Cycle Models: A Comparative Approach, *Journal of Computer Science and Its Application*, 2021, No. 28, pp. 122–133.
6. Jacobson I., Spence I., Bittner K. USE-CASE 2.0 The Guide to Succeeding with Use Cases [Electronic Recourse]. Access mode: https://www.ivarjacobson.com/sites/default/files/field_jji_file/article/use-case_2_0_jan11.pdf
7. Arcos-Medina G., Mauricio D. The Influence of the Application of Agile Practices in Software Quality Based on ISO/IEC 25010 Standard, *International Journal of Information Technologies and Systems Approach*, 2020, №13, pp. 1–27.
8. Mohan M., Greer D. A survey of search-based refactoring for software maintenance, *Journal of Software Engineering Research and Development*, 2018, №6, pp. 1–52.
9. Ryan M. Mastering OOP: A Practical Guide to Inheritance, Interfaces, and Abstract Classes [Electronic Recourse]. Access mode: <https://www.smashingmagazine.com/2019/11/guide-oop-inheritance-interfaces-abstract-classes/>
10. Taubler D. When to Use Abstract Classes [Electronic Recourse]. Access mode: <https://betterprogramming.pub/when-to-use-abstract-classes-70fe526165ac>
11. Al-Fedaghi S. Classes in Object-Oriented Modeling (UML): Further Understanding and Abstraction, *International Journal of Computer Science and Network Security*, 2021, №21, pp. 139–150.
12. Minh Hoang Lien Vo, Hoang Q. Transformation of UML class diagram into OWL Ontology, *Journal of Information and Telecommunication*, 2020, No. 4, Issue 1.

13. Gandhi P., Pradeep K. Optimization of Object-Oriented Design using Coupling Metrics, *International Journal of Computer Applications*, 2011, No. 27, pp. 41–44.
14. Saeed M. G., Alasaady M. T. Three Levels Quality Analysis Tool for Object Oriented Programming, *International Journal of Advanced Computer Science and Applications*, 2018, № 9, pp. 522–536.
15. Zongmin Ma, Zhongchen Yuan, Yan Li Two-level clustering of UML class diagrams based on semantics and structure, *Information and Software Technology*, 2021, No. 130, 106456.
16. Miao Zhang, Jacky Wai Keung, Yan Xiao, Md Alamgir Kabir Evaluating the effects of similar-class combination on class integration test order generation, *Information and Software Technology*, 2021, №129, 106438.
17. Pérez F., Echeverría J., Lapeña R., Cetina C. Comparing manual and automated feature location in conceptual models: A Controlled experiment, *Information and Software Technology*, 2020, No. 125, 106337.
18. Benlhachmi K., Benattou M. A Formal Model of Conformity and Security Testing of Inheritance for Object Oriented Constraint Programming, *Journal of Information Security*, 2013, №4, pp. 113–123.
19. Vozovikov Yu. N., Kungurtsev A. B., Novikova N. A. Information technology for automated compilation of use cases, *Science practices of Donetsk National Technical University*, 2017, No. 1 (30), pp. 46–59.
20. Kungurtsev O., Novikova N., Reshetnyak M., Cherepinina Ya., Gromaszek K., Jarykbassov D. Method for defining conceptual classes in the description of use cases. *Odessa: Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2019*, 2019, 1117624.
21. Kungurtsev O. B., Novikova N. O., Zinovatna S. L., Komleva N. O. Automated object-oriented for software module development, *Applied Aspects of Information Technology*, 2021, №4, pp. 338–353.
22. Winkler W. E. String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage, *Proceedings of the Section on Survey Research Methods*, 1990, pp. 354–359.

Received 10.09.2022.
Accepted 08.11.2022.

УДК 004.415.2

ВИЗНАЧЕННЯ ВІДНОСИН УСПАДКУВАННЯ ТА РЕСТРУКТУРИЗАЦІЯ МОДЕЛЕЙ ПРОГРАМНИХ КЛАСІВ У ПРОЦЕСІ РОЗРОБКИ ІНФОРМАЦІЙНИХ СИСТЕМ

Кунгурцев О. Б. – канд. техн. наук, професор кафедри Інженерії програмного забезпечення Національного університету «Одеська політехніка», Одеса, Україна.

Витнова А. І. – студентка кафедри Інженерії програмного забезпечення Національного університету «Одеська політехніка», Одеса, Україна.

АНОТАЦІЯ

Актуальність. Реалізація різних варіантів використання може виконуватись різними командами розробників у різний час. Це призводить до створення погано структурованого коду. Проблема ускладнюється при розробці середніх та великих проектів у стислий термін.

Мета. Оскільки успадкування є одним із ефективних способів структурування та покращення якості коду, метою дослідження є визначення можливих зв'язків успадкування для різноманітних моделей класів.

Метод. Запропоновано виділення з множини класів, що представляють модель класів на певному етапі проектування, підмножин, для яких можливий загальний батьківський клас (в окремому випадку абстрактний клас). Для вирішення завдання сформульовано ознаки спільності класів. Удосконалено математичну модель концептуального класу за рахунок включення інформації про обов'язки класу, його методи та атрибути. Встановлено зв'язок кожного класу з сценаріями, для яких він використовується. Запропоновано систему типів даних для елементів моделі класу. Розширено опис сигнатур методів класів. Розроблено метод реструктуризації моделі класів, що передбачає 3 етапи. У першому визначаються коефіцієнти близькості класів. На другому створюються підмножини можливих дочірніх класів. На третьому виконується автоматизоване перетворення структури класів з урахуванням виявлених відносин спадкування.

Результати. Розроблено програмний продукт для проведення експериментів щодо виявлення можливих відносин успадкування залежно від кількості класів та ступеня їхньої подібності. Результати проведених випробувань показали ефективність ухвалених рішень.

Висновки. Метод використовує алгоритм формування підмножин класів, які можуть мати одного предка та алгоритм автоматичного створення та перетворення класів для побудови дворівневої ієрархії класів. Результати дослідження реалізовані у програмному продукті. Експеримент показав триразове скорочення помилок при виявленні наслідування та багаторазове скорочення часу порівняно з існуючою технологією.

КЛЮЧОВІ СЛОВА: модель класу, атрибут класу, метод класу, типи даних, варіант використання, спадкування.

ЛІТЕРАТУРА / LITERATURE

1. Object-Oriented Analysis and Design with Applications 3rd Edition / [G. Booch, R. A. Maksimchuk, M. W. Engle et al.]. – Boston : Addison-Wesley Professional, 2007 – 694 p.
2. Lee G. Modern Programming: Object Oriented Programming and Best Practices / G. Lee. – Birmingham : Packt, 2019. – 266 p.
3. Baesens B. Beginning Java Programming: The Object-Oriented Approach / B. Baesens, A. Backiel, S. Broucke. – Birmingham : Wrox, 2015. – 672 p.
4. Agility as an innovation driver: towards an agile front end of innovation framework / [M. Brand, V. Tiberius, P. M. Bican, A. Brem]. – Potsdam : Springer, 2021. – P. 157–187.
5. Project Implementation Decision Using Software Development Life Cycle Models: A Comparative Approach / [M. A. Adeagbo, J. Akinsola, A. A. Awoseyi, F. Kasali] // Jour-

- nal of Computer Science and Its Application. – 2021. – № 28. – P. 122–133.
6. Jacobson I. USE-CASE 2.0 The Guide to Succeeding with Use Cases [Electronic Recourse] / I. Jacobson, I. Spence, K. Bittner. – Access mode: https://www.ivarjacobson.com/sites/default/files/field_iji_file/article/use-case_2_0_jan11.pdf
 7. Arcos-Medina G. The Influence of the Application of Agile Practices in Software Quality Based on ISO/IEC 25010 Standard / G. Arcos-Medina, D. Mauricio // *International Journal of Information Technologies and Systems Approach.* – 2020. – №13. – P. 1–27.
 8. Mohan M. A survey of search-based refactoring for software maintenance / M. Mohan, D. Greer // *Journal of Software Engineering Research and Development.* – 2018. – №6. – P. 1–52.
 9. Ryan M. Mastering OOP: A Practical Guide to Inheritance, Interfaces, and Abstract Classes [Electronic Recourse] / M. Ryan. – Access mode: <https://www.smashingmagazine.com/2019/11/guide-oop-inheritance-interfaces-abstract-classes/>
 10. Taubler D. When to Use Abstract Classes [Electronic Recourse] / D. Taubler. – Access mode: <https://betterprogramming.pub/when-to-use-abstract-classes-70fe526165ac>
 11. AI-Fedaghi S. Classes in Object-Oriented Modeling (UML): Further Understanding and Abstraction / S. AI-Fedaghi // *International Journal of Computer Science and Network Security.* – 2021. – №21. – P. 139–150.
 12. Minh Hoang Lien Vo Transformation of UML class diagram into OWL Ontology / Minh Hoang Lien Vo, Q. Hoang // *Journal of Information and Telecommunication.* – 2020. – №4. – Issue 1.
 13. Gandhi P. Optimization of Object-Oriented Design using Coupling Metrics / P. Grandhi, K. Pradeep // *International Journal of Computer Applications.* – 2011. – №27. – P. 41–44.
 14. Saeed M. G. Three Levels Quality Analysis Tool for Object Oriented Programming / M. G. Saeed, M. T. Alasaady // *International Journal of Advanced Computer Science and Applications.* – 2018. – №9. – P. 522–536.
 15. Zongmin Ma Two-level clustering of UML class diagrams based on semantics and structure / Zongmin Ma, Zhongchen Yuan, Li Yan // *Information and Software Technology.* – 2021. – №130. – 106456.
 16. Evaluating the effects of similar-class combination on class integration test order generation / [Miao Zhang, Jacky Wai Keung, Yan Xiao, Md Alamgir Kabir] // *Information and Software Technology.* – 2021. – №129. – 106438.
 17. Comparing manual and automated feature location in conceptual models: A Controlled experiment / [F. Pérez, J. Echeverría, R. Lapeña, C. Cetina] // *Information and Software Technology.* – 2020. – №125. – 106337.
 18. Benlhachmi K. A Formal Model of Conformity and Security Testing of Inheritance for Object Oriented Constraint Programming / K. Benlhachmi, M. Benattou // *Journal of Information Security.* – 2013. – №4. – P. 113–123.
 19. Vozovikov Yu. N. Information technology for automated compilation of use cases / Yu. N. Vozovikov, A. B. Kungurtsev, N. A. Novikova // *Science practices of Donetsk National Technical University.* – 2017. – No. 1 (30). – P. 46–59.
 20. Method for defining conceptual classes in the description of use cases / [O. Kungurtsev, N. Novikova, M. Reshetnyak et al.]. – Odessa: Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2019, 2019 – 1117624.
 21. Automated object-oriented for software module development / [O. B. Kungurtsev, N. O. Novikova, S. L. Zinovatna, N. O. Komleva] // *Applied Aspects of Information Technology.* – 2021. – №4. – P. 338–353.
 22. Winkler W. E. String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage / W. E. Winkler // *Proceedings of the Section on Survey Research Methods.* – 1990. – P. 354–359.

SYNTHESIS OF THE SYMBOLOGIES OF MULTICOLOR INTERFERENCE-RESISTANT BAR CODES ON THE BASE OF MULTI-VALUED BCH CODES

Sulema Ye. S. – Dr. Sc., Associate Professor, Head of the Department of Computer Systems Software of the National Technical University of Ukraine “Kyiv Polytechnic Institute named after Igor Sykorsky”, Kyiv, Ukraine.

Drozdhenko L. V. – Assistant Professor of the Department of Computer Systems Software of the National Technical University of Ukraine “Kyiv Polytechnic Institute named after Igor Sykorsky”, Kyiv, Ukraine.

Dychka A. I. – Post-graduate student of the Department of Computer Systems Software of the National Technical University of Ukraine “Kyiv Polytechnic Institute named after Igor Sykorsky”, Kyiv, Ukraine.

ABSTRACT

Context. The problem of constructing a set of barcode patterns for multicolor barcodes that are resistant to distortions of one or two elements within each pattern is considered.

Objective. The goal of the work is ensuring the reliability of the reading of multi-color barcode images.

Method. A multicolor barcode pattern has the property of interference immunity if its digital equivalent (vector) is a codeword of a multi-valued (non-binary) correcting code capable to correct errors (distortions of the pattern elements). It is shown that the construction of barcode patterns should be performed on the basis of a multi-valued correcting BCH code capable to correct two errors. A method is proposed for constructing a set of interference-resistant barcode patterns of a given capacity, which ensure reliable reproduction of data when they are read from a carrier. A procedure for encoding data with a multi-valued BCH code based on the generator matrix of the code using operations by the modulo of a prime number has been developed. A new method of constructing the check matrix of the multivalued BCH code based on the vector representation of the elements of the finite field is proposed. A generalized algorithm for generating symbologies of a multi-color barcode with the possibility of correcting double errors in barcode patterns has been developed. The method also makes it possible to build symbology of a given capacity based on shortened BCH codes. A method of reducing the generator and check matrices of a multi-valued full BCH code to obtain a shortened code of a given length is proposed. It is shown that, in addition to correction double errors, multi-valued BCH codes also make it possible to detect errors of higher multiplicity – this property is enhanced when using shortened BCH codes. The method provides for the construction of a family of multicolor noise-immune barcodes.

Results. On the basis of the developed software tools, statistical data were obtained that characterize the ability of multi-valued BCH codes to detect and correct errors, and on their basis to design multi-color interference-resistant bar codes.

Conclusions. The conducted experiments have confirmed the operability of the proposed algorithmic tools and allow to recommend it for use in practice for developing interference-resistant multi-color barcodes in automatic identification systems.

KEYWORDS: barcoding, multicolor barcodes, interference immunity of barcodes, BCH codes.

ABBREVIATIONS

BC – barcode;
BCH code – Bose-Choudhuri-Hocquenghem code;
BC-pattern – barcode pattern;
BC-symbol – barcode symbol;
HCC2D barcode – high capacity colored two dimensional barcode;
HCCB – high capacity color barcode;
JAB code – just another barcode;
LCM – least common multiple;
QR code – quick response barcode;
URL – uniform resource locator.

NOMENCLATURE

$B()$ is an information word;
barcode-pattern (B) is a program procedure, provides the printing of the barcode pattern for the given information word $B()$;
 $B(x)$ is a polynomial which corresponds to information word $B()$;
 $\det M$ is a determinant of the matrix M ;
 d_{\min} is a minimal Hamming distance of the correcting code;
do – execute;

for – start of cycle;
 $G_{(s,u)}$ is a generator matrix of (s, u) -BCH code;
 $GF()$ is a Galois field;
 $g(x)$ is a generator polynomial of the correcting code;
 $H_{(s,u)}$ is a check matrix of the (s, u) -BCH code;
 $M^{(i)}(x)$ is a minimal polynomial of the field's element;
 m is a degree of minimal polynomials;
 $p(x)$ is an irreducible polynomial;
 q is an amount of colors for barcode-patterns painting;
 S is an error syndrome;
 s is a the total length of the codewords;
 u is an amount of informational positions in code-words;
 V is a capacity of the barcode symbology;
 v is a degree of generator polynomial;
 x is a root of the error locator polynomial;
 X is an error locator;
 Y is an error value;
 $Z()$ is a codeword (vector of the barcode pattern);
 Z' is a the resived vector;
 $Z(x)$ is a polynomial which corresponds to codeword $Z()$;
 α is a primitive element of Galois field;
 σ is a coefficient of the error locator polynomial;

$\sigma(x)$ is an error locator polynomial;
 Ω is a barcode symbology.

INTRODUCTION

Data barcoding is one type of automatic identification. The advantages of barcoding are the speed of entering data into the computer system, the low cost of making barcode symbols and ease of use; BC is read from the accounting objects optically, including at a distance.

During its more than 60-year history of development, BCs have undergone a certain evolution: linear BCs, stacks, matrix, and eventually multi-color BCs. During the last decade, there has been a steady tendency to expand the scope of application of multi-color BCs, because multi-color allows to significantly (several times) increase the information density – to provide a greater of information quantity per unit area of the carrier without changing the geometric dimensions of the image elements, than is allowed by black-and-white BCs.

However, the processes of recognition and decoding of barcode images become more complicated when using multi-color barcodes. The researchers note that the reasons for this may be: distortion of the geometric dimensions of image elements during reading due to certain optical effects; mixing of colors on the border of neighboring elements with different colors; aging of dyes; folding and blurring of paint, etc. [1]. Therefore, for reliable reading of barcode data from the accounting object, it is necessary to ensure the interference immunity of barcode images [2].

Structurally, the BC-symbol consists of an array of BC-patterns arranged in the form of a rectangle or a square (Fig. 1). To ensure an adequate level of reliability of data reading, a multi-color BC should be constructed so that the property of immunity to interference is provided at two levels – at the level of the entire BC-symbol, as well as at the level of BC-patterns – the minimal structural units of the image.

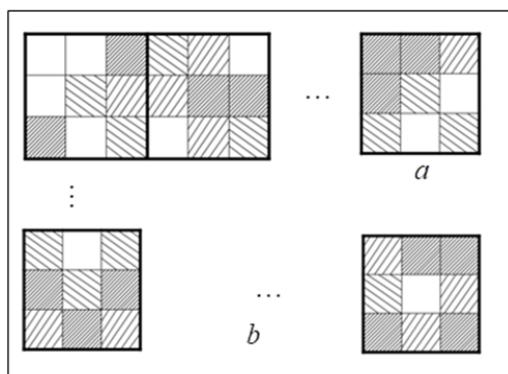


Figure 1 – The structure of a multi-color barcode symbol:
 a – BC-pattern; b – BC-symbol

The object of study is the process of developing multi-color barcodes with improved reliability indicators.

Single-level system for ensuring interference resistance is used in the existing BCs – the BC-symbol, which

is an array of BC-patterns, is coded with a correcting code capable to correct multiple errors, usually the Reed-Solomon code. To increase the interference resistance of a multi-color barcode image, it is suggested to additionally apply interference-resistant coding within each barcode pattern.

The subject of study is a method of ensuring interference-resistance of BC-patterns, which is based on the use of multi-valued BCH codes, which are capable to correct double errors in BC-patterns.

The purpose of the work is to develop the technology of designing multi-color interference-resistant BC-patterns, which would ensure reliable reproduction of data when read from the carrier.

1 PROBLEM STATEMENT

Every BC has its own set of BC-patterns, each of which denotes a symbol (or a combination of symbols) of computer alphabet. This set of barcode patterns is called barcode symbology.

Let it be necessary to create the symbology of the capacity V of an interference-resistant multi-color matrix BC with parameters q, s ; q is the number of colors used to color the elements of the BC-patterns when $q > 2$; s is the number of elements (cells) in the BC-pattern (see Fig. 1).

The colors with which each element of the BC-pattern can be painted will be denoted by numbers: $0, 1, 2, \dots, q-1$. Let's match the digital equivalent of the BC-pattern consisting of s elements – the vector of the BC-pattern $Z = (z_0 z_1 \dots z_{s-1})$, where $z_i \in \{0, 1, 2, \dots, q-1\}$.

In order for the BC-pattern to be interference-resistant, its vector Z must be a codeword of a multi-valued ($q > 2$) correcting code capable to correct distortions (errors). We will assume that during the reading from the carrier of BC-patterns, the most probable cases are damages (distortions) of one or two elements in BC-patterns. In this case, the vector Z of BC-pattern must be a codeword of a correcting code with a minimal code distance of $d_{\min} = 5$. Such a correcting code can be a multi-valued (q -valued) BCH code, $q > 2$.

Thus, it is necessary to solve the problem of construction the symbologies of multi-colored barcodes with the possibility of correction single or double errors in readable barcode patterns, as well as detecting errors (distortions) of greater multiplicity.

2 REVIEW OF THE LITERATURE

In 2007, Microsoft developed a 4-color High Capacity Color Barcode (HCCB), which uses black, red, green and yellow colors. Some researchers rightly believe that it is from this code (which is also called Microsoft Tag) multi-color barcoding has started. The first studies of HCCB were performed by Devi Parikh and Gavin Jancke, who proposed a method for decoding this code, the main procedures of which are the localization and segmentation of the color image of the BC-symbol [1]. The color image is divided into clusters, each center of the cluster is matched with one of the reference colors of the palette, which is used when printing the BC-symbol. In HCCB, informa-

tion is provided by colored triangles, which are placed on the carrier in the form of a matrix – the number of rows in the BC-symbol can be from 10 to 60, and in a row – from 20 to 120 triangles. The triangle gives a quadruple value of 0, 1, 2 or 3. Thus, the capacity of the BC-symbol can be from 200 to 7200 quadruple digits.

In subsequent scientific works on this topic, researchers began to use a square (instead of a triangle) as a minimal structural element of a multi-color barcode image, and also improved color recognition procedures during optical reading. For example, in [2] a series of algorithms were developed for recognizing color elements of an image using a small number of reference colors under illuminant of different intensities. The illuminant was considered as parametric color converter. This transformation was used to visualize (recognize) an unknown color element under a reference illuminant that can be identified using training data.

The authors of the study [3] proposed a method of recognizing a color barcode image without using reference colors, and also improved the decoding algorithm, which resulted in an increase in the speed of data reading and a decrease in the intensity of errors. In addition, it was possible to reduce the computational complexity of data reading and data processing.

The subject of the research in [4] was the spectral difference in the color channels of the devices for printing BC-symbols, where the color palette C, M, Y is used (C – cyan, M – magenta, Y – yellow), and color channels devices for reading BC-symbols, in which the R, G, B palette is used (R – red, G – green, B – blue). In order to mitigate the effect of inter-channel interference in color channels when printing BC-symbols and color channels of reading devices, the authors developed an algorithm for eliminating interference during the implementation of printing and reading (scanning) processes. As a result, it was possible to significantly reduce the probability of errors and increase the decoding speed.

A team of researchers in [5] introduced High Capacity Colored Two Dimensional (HCC2D) QR-based code with improved information density indicators due to the use of 4, 8 or 16 colors; at the same time, high reliability of data reproduction during scanning is ensured. The authors experimentally proved that HCC2D has the same information density as HCCB (Microsoft), and is not inferior in stability (reliability) QR code.

In [6] continued the study of HCC2D code. In particular, the authors investigated the frequency of decoding errors using different classifiers: the Minimum Distance, Decision Trees, K-means, Navie Bayes, and Support Vector Machines. It is shown that the K-means algorithm is the most effective classifier in experiments. An efficient algorithm for decoding multi-color BCs with reasonable computational costs is proposed.

Some researchers, in particular in [7, 8], proposed ways to increase the information density of QR-codes by using multicolor barcode elements in the QR-code structure. The palette C, M, Y when printing BC-symbols and the palette R, G, B in code scanning devices were studied,

and on their basis the possibility of obtaining 8-color components in QR-codes was analyzed.

The authors of the publication [9] performed a comparative analysis of two-dimensional barcodes and outlined some directions for improving their characteristics – information density and interference resistance. The issue of compression of alphanumeric data before their presentation in barcode form is considered in [10]. A method is proposed that allows you to compress data 1.4 – 1.7 times.

In addition to compression, some researchers, in particular the authors of [8], also suggest applying multiplexing and multilayered technique of color barcode images. For example, in [11] a three-layer 8-color BC is proposed for the presentation of three independent alphanumeric messages in a single barcode symbol, in particular, the presentation of URL as one of them.

Eight colors for data presentation were also used by the developers of the JAB Code (Just Another Barcode), which managed to triple the information density of the barcode labels without changing the geometric dimensions of the barcode image elements [12].

In multi-color barcoding the problem of ensuring interference resistance of barcode images is extremely important. For reliable reading of the BC-symbols, the data before being applied to the carrier are coded using a correcting code with a high corrective ability, usually the Reed-Solomon code [4, 7–9, 12]. Besides, structural methods of increasing reliability can also be additionally applied [13].

However, it is guaranteed to achieve reliable reproduction of data when scanning a large-capacity multi-color barcode image only under the conditions of two-level reliability assurance – at the level of the entire barcode symbol (upper level), as well as at the level of the smallest structural units (lower level). The study of this approach was started in [14], where it was proposed to use a multi-valued Hamming code at the level of the BC-patterns. In this investigation, a more powerful multi-valued BCH code is proposed for this purpose.

3 MATERIALS AND METHODS

The BCH code is a cyclic correcting code in which information encoding is reduced to the multiplication of a

polynomial of $B(x) = \sum_{i=0}^{u-1} b_i x^i$ of degree $u-1$, which corresponds to information word $B = (b_0 b_1 \dots b_{u-1})$, to the generator polynomial

$$g(x) = \sum_{i=0}^v g_i x^i = g_0 + g_1 x + \dots + g_v x^v \text{ of degree } v :$$

$$Z(x) = B(x)g(x) ,$$

where $Z(x) = \sum_{i=0}^{s-1} z_i x^i$ is a polynomial of degree $s-1$ corresponding to s -bit codeword $Z = (z_0 z_1 \dots z_{s-1})$, which is

the digital equivalent of the BC-pattern; $s = u + v$; $b_i, g_i, z_i \in GF(q)$ [15].

If the generator polynomial $g(x)$ is known, then the generator matrix $G_{(s,u)}$ of the (s, u) -BCH code can be written as follows:

$$G_{(s,u)} = \begin{pmatrix} g(x) \\ xg(x) \\ \vdots \\ x^{n-1}g(x) \end{pmatrix} = \begin{pmatrix} g_0g_1 \dots g_v & 0 & \dots & 0 \\ 0 & g_0 \dots g_{v-1} & g_v & 0 \dots 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & g_0 & g_1 & \dots & g_v \end{pmatrix}$$

it has dimension $u \times s$.

The generator polynomial $g(x)$ of the (s, u) -BCH code with the minimal code distance $d_{\min}=5$, capable to correct two errors, is defined as the least common multiple of the minimal polynomials $M^{(1)}(x), M^{(2)}(x), M^{(3)}(x), M^{(4)}(x)$ for elements $\alpha^1, \alpha^2, \alpha^3, \alpha^4$ of the field $GF(q^m)$, where α^i is the root of the irreducible polynomial $M^{(i)}(x)$, that is, $M^{(i)}(\alpha^i)=0$:

$$g(x) = \text{LCM}(M^{(1)}(x), M^{(2)}(x), M^{(3)}(x), M^{(4)}(x)),$$

$\alpha \in GF(q^m)$, m is the degree of minimal polynomials [15]. $GF(q)$ is called the field of characters, and $GF(q^m)$ is the field of locators.

Next, we will consider the case when q – prime number; the case when q is an exponent of a prime number is the subject of a separate study.

The BCH code is called full if $s = q^m - 1$.

Some non-binary ($q > 2$) full BCH codes with $d_{\min}=5$, which are suitable for the problem to be solved, are listed in Table 1.

Table 1 – Some non-binary full (s, u) -BCH code

Number of colors	$q=3$	$q=5$	$q=7$
Field of characters	$GF(3)$	$GF(5)$	$GF(7)$
(s, u) -BCH code	$(8, 3)$ - $(26, 17)$ -	$(24, 16)$ -	$(48, 40)$ -

Let $q=3$ (the case of a three-color barcode).

Consider, for example, the ternary $(8, 3)$ -BCH code (see Table 1).

To build such a code, two fields are used: $GF(3)$ – the field of characters (Fig. 2), and $GF(3^2)$ – the field of locators (Table 2), which is an extension over $GF(3)$. The construction of $GF(3^2)$ is based on the irreducible polynomial $p(x) = x^2 + x + 2$.

+	0	1	2	-	0	1	2	·	0	1	2	:	0	1	2	
0	0	1	2	0	0	2	1	0	0	0	0	0	0	-	0	0
1	1	2	0	1	1	0	2	1	0	1	2	1	-	1	2	
2	2	0	1	2	2	1	0	2	0	2	1	2	-	2	1	

Figure 2 – Performing operations in the $GF(3)$

Each element α^i of the field $GF(3^2)$, $i \geq 1$, corresponds to the minimal polynomial $M^{(i)}(x)$, the degree of which does not exceed two.

In the field $GF(3^2)$ $\alpha^i = x^{i-8}$, $\alpha^{-i} = \alpha^{8-i}$, $\alpha^8 = \alpha^{-8} = \alpha^0 = 1$.

Let's find the generator polynomial $g(x)$ of the ternary $(8, 3)$ -BCH code:

$$g(x) = \text{LCM}(M^{(1)}(x), M^{(2)}(x), M^{(3)}(x), M^{(4)}(x)) = \text{LCM}(x^2+x+2, x^2+1, x^2+x+2, x+1) = (x^2+x+2)(x^2+1)(x+1) = x^5 + 2x^4 + x^3 + x^2 + 2 \rightarrow g_0g_1 \dots g_5 = 2 \ 0 \ 1 \ 1 \ 2 \ 1.$$

The generator polynomial $g(x)$ corresponds to the generator matrix $G_{(8,3)}$:

$$G_{(8,3)} = \begin{pmatrix} z_0 & z_1 & z_2 & z_3 & z_4 & z_5 & z_6 & z_7 \\ 2 & 0 & 1 & 1 & 2 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 & 1 & 2 & 1 & 0 \\ 0 & 0 & 2 & 0 & 1 & 1 & 2 & 1 \end{pmatrix}$$

Table 2 – Elements of the field $GF(3^2)$ and their minimal polynomials

Exponential notation		Polynomial notation	Vector notation	Decimal notation	Minimal polynomial
With a non-negative degree of the primitive element α of the field	With a negative degree of the primitive element α of the field				
–	–	0	(0, 0)	0	–
α^0	α^{-8}	1	(0, 1)	1	–
α^1	α^{-7}	α	(1, 0)	3	$x^2 + x + 2$
α^2	α^{-6}	$2\alpha + 1$	(2, 1)	7	$x^2 + 1$
α^3	α^{-5}	$2\alpha + 2$	(2, 2)	8	$x^2 + x + 2$
α^4	α^{-4}	2	(0, 2)	2	$x + 1$
α^5	α^{-3}	2α	(2, 0)	6	$x^2 + 2x + 2$
α^6	α^{-2}	$\alpha + 2$	(1, 2)	5	$x^2 + 1$
α^7	α^{-1}	$\alpha + 1$	(1, 1)	4	$x^2 + 2x + 2$

On the basis of the generator matrix $G_{(s,u)}$ of the BCH code, it is possible to construct the symbology of an interference-resistant barcode with the possibility of correcting single or double distortions of elements (errors) in BC-pattern. For this, the u -bit informational word $B = (b_0 b_1 \dots b_{u-1})$ must be converted into the s -bit word $Z = (z_0 z_1 \dots z_{s-1})$, which is a vector (the digital equivalent) of the BC-pattern, i.e., encode the word B with the (s, u) -BCH code, and then match the BC-pattern to the vector Z . The coding operation is given by the equation $Z = B \cdot G_{(s,u)}$ and performed according to the rules of the field $GF(q)$.

For example, if $B = (b_0 b_1 b_2)$, where $b_i \in \{0, 1, 2\}$, then $Z = B \cdot G_{(8,3)}$, i.e.

$$\|b_0 b_1 b_2\| \cdot \begin{pmatrix} 2 & 0 & 1 & 1 & 2 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 & 1 & 2 & 1 & 0 \\ 0 & 0 & 2 & 0 & 1 & 1 & 2 & 1 \end{pmatrix} = \|z_0 z_1 \dots z_7\|, \quad (1)$$

whence it follows that

$$z_0 = 2b_0, \quad z_1 = 2b_1, \quad z_2 = b_0 + 2b_2, \quad z_3 = b_0 + b_1, \\ z_4 = 2b_0 + b_1 + b_2, \quad z_5 = b_0 + 2b_1 + b_2, \quad z_6 = b_1 + 2b_2, \quad z_7 = b_2$$

(operations should be performed according to modulo 3).

Let $B = (1\ 0\ 2)$, then $Z = (2\ 0\ 2\ 1\ 1\ 0\ 1\ 2)$, and the corresponding BC-pattern is shown in Fig. 3

Taking all possible values of the vector B – from $(0\ 0\ 0)$ to $(2\ 2\ 2)$ and applying the coding procedure (1) to each word, we get $3^3 = 27$ different BC-patterns that form the symbology Ω of an interference-resistant three-color barcode, in which correction of single or double errors is possible within each BC-pattern (Table 3).

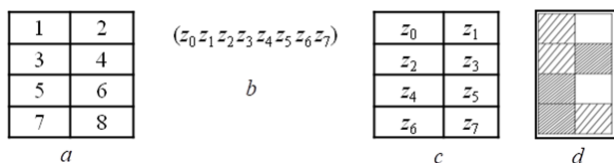


Figure 3 – Creation of a three-color BC-pattern with the possibility of correction double distortion of the elements of the BC-pattern; a – the structure of the BC-pattern; b – vector of the BC-pattern; c – filling out the BC-pattern; d – coloring the BC-pattern

The capacity of the Ω symbology is 27 BC-patterns ($N=27$); it can be matched with the numerical set $\Omega = \{0, 1, 2, \dots, 26\}$.

The process of synthesizing the symbology of a three-color interference-resistant barcode with the possibility of correcting double errors in BC-patterns can be described by the following generalized algorithm:

```
for  $B = 000$  to  $222$  do
 $|Z| = |B| \cdot |G_{(8,3)}|$ 
 $Z[1..8] := (z_0 z_1 \dots z_7)$ 
 $Z[1..8] \rightarrow \text{barcode\_pattern}(B)$ 
```

The alphanumeric sequence, which is to be presented in the form of a barcode, is converted into a numerical form, the elements of which are numbers from the range $0 - 26$ (from the set Ω), and then each number is matched with a BC-pattern (Table 3). Next, BC-patterns are applied to the carrier, forming a BC-symbol.

While reading BC-symbol successively allocate BC-patterns, each of which is matched with a digital equivalent – s -bit vector $Z' = (z'_0 z'_1 \dots z'_{s-1})$, which is decoded according to the rules of the (s, u) -BCH code with $d_{\min} = 5$.

Decoding is carried out on the basis of the check matrix of the of the BCH code, which is presented in the form:

$$H_{(s,u)} = \begin{pmatrix} \alpha_i \\ \alpha_i^2 \\ \alpha_i^3 \\ \alpha_i^4 \end{pmatrix}, i = 0, 1, 2, \dots, s-1,$$

where α_i are elements of $GF(q^m)$.

For the ternary $(8, 3)$ -BCH code, it looks like this:

$$H_{(8,3)} = \begin{pmatrix} \alpha^0 & \alpha^1 & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 & \alpha^7 \\ \alpha^0 & \alpha^2 & \alpha^4 & \alpha^6 & \alpha^8 & \alpha^{10} & \alpha^{12} & \alpha^{14} \\ \alpha^0 & \alpha^3 & \alpha^6 & \alpha^9 & \alpha^{12} & \alpha^{15} & \alpha^{18} & \alpha^{21} \\ \alpha^0 & \alpha^4 & \alpha^8 & \alpha^{12} & \alpha^{16} & \alpha^{20} & \alpha^{24} & \alpha^{28} \end{pmatrix}$$

Table 3 – Symbology of the three-color interference-resistant barcode with the possibility of correction single or double errors in the BC-patterns based on the ternary $(8, 3)$ -BCH code

The serial number of the BC-pattern	BC-pattern	Vector of BC-pattern
0		00000000
1		00201121
2		00102212
⋮		
26		11012202

Taking into account that $\alpha^0 = \alpha^8 = \alpha^{16} = \alpha^{24}$ and substituting the corresponding two-digit vector columns instead of α^i (see the vector representation of the field elements in Table 2), we obtain

$$H_{(8,3)} = \begin{pmatrix} z'_0 z'_1 z'_2 z'_3 z'_4 z'_5 z'_6 z'_7 \\ \begin{pmatrix} 0 & 1 & 2 & 2 & 0 & 2 & 1 & 1 \\ 1 & 0 & 1 & 2 & 2 & 0 & 2 & 1 \\ 0 & 2 & 0 & 1 & 0 & 2 & 0 & 1 \\ 1 & 1 & 2 & 2 & 1 & 1 & 2 & 2 \\ 0 & 2 & 1 & 1 & 0 & 1 & 2 & 2 \\ 1 & 2 & 2 & 0 & 2 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 2 & 1 & 2 & 1 & 2 \end{pmatrix} \end{pmatrix}$$

The product of the matrix $G_{(8,3)}$ to the transposed matrix $H_{(8,3)}$ is zero ($G_{(8,3)} H_{(8,3)}^T = 0$).

The received vector Z' is decoded according to the algorithm in Fig. 4.

The error syndrome $S = S_1 S_2 S_3 S_4$ is calculated as $S = Z' \cdot H_{(8,3)}^T$, operations are performed according to modulo 3:

$$\begin{aligned} S_1 &= z'_0 \binom{0}{1} + z'_1 \binom{1}{0} + z'_2 \binom{2}{1} + z'_3 \binom{2}{2} + z'_4 \binom{0}{2} + z'_5 \binom{2}{0} + z'_6 \binom{1}{2} + z'_7 \binom{1}{1}, \\ S_2 &= z'_0 \binom{0}{1} + z'_1 \binom{2}{1} + z'_2 \binom{0}{2} + z'_3 \binom{1}{2} + z'_4 \binom{0}{1} + z'_5 \binom{2}{1} + z'_6 \binom{0}{2} + z'_7 \binom{1}{2}, \\ S_3 &= z'_0 \binom{0}{1} + z'_1 \binom{2}{2} + z'_2 \binom{1}{2} + z'_3 \binom{1}{0} + z'_4 \binom{0}{2} + z'_5 \binom{1}{1} + z'_6 \binom{2}{1} + z'_7 \binom{2}{0}, \\ S_4 &= z'_0 \binom{0}{1} + z'_1 \binom{0}{2} + z'_2 \binom{0}{1} + z'_3 \binom{0}{2} + z'_4 \binom{0}{1} + z'_5 \binom{0}{2} + z'_6 \binom{0}{1} + z'_7 \binom{0}{2}. \end{aligned}$$

If $S = 0$, then there are no errors in the word Z' , otherwise ($S \neq 0$) – the word Z' contains one or two errors (Fig. 4).

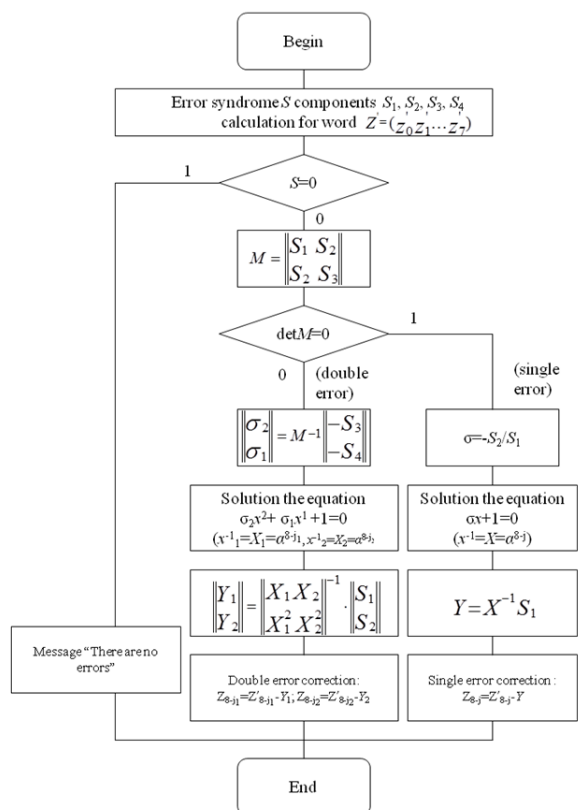


Figure 4 – Block-diagram of the single and double error correction algorithm for the word $Z' = (z'_0 z'_1 \dots z'_7)$

Further we form the matrix $M = \begin{vmatrix} S_1 & S_2 \\ S_2 & S_3 \end{vmatrix}$ and calculate

determinant $\det M$. If $\det M = 0$, then there is one error in the word Z , otherwise ($\det M \neq 0$) – two errors.

Consider the case when $\det M \neq 0$.

Find the roots x_1, x_2 of the error locator polynomial $\sigma(x) = \sigma_2 x^2 + \sigma_1 x + 1$, where the coefficients σ_2, σ_1 determine like

$$\begin{vmatrix} \sigma_2 \\ \sigma_1 \end{vmatrix} = M^{-1} \begin{vmatrix} -S_3 \\ -S_4 \end{vmatrix}.$$

The error locators X_1, X_2 are values: $X_1 = x_1^{-1}, X_2 = x_2^{-1}$.

The equation $\sigma_2 x^2 + \sigma_1 x + 1 = 0$ is solved in the field $GF(3^2)$ according to Chien search algorithm [15], which consists in the successive calculation of $\sigma(\alpha^j)$ for $j=0, 1, \dots, 7$, and checking the received value for zero. There is no other way of solving equations in finite fields.

Positions (digits) of the codeword Z correspond to the degrees of the primitive element α of the field:

$$\begin{matrix} \alpha^{-8} & \alpha^{-7} & \alpha^{-6} & \alpha^{-5} & \alpha^{-4} & \alpha^{-3} & \alpha^{-2} & \alpha^{-1}, \\ \alpha^0 & \alpha^1 & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 & \alpha^7, \\ Z = (z_0 z_1 z_2 z_3 z_4 z_5 z_6 z_7). \end{matrix}$$

Therefore, if $\sigma(\alpha^j) = 0$, then the error locator X is equal to $\alpha^{-j} = \alpha^{8-j}$, and the location of the error is the digit numbered $8-j$.

The equation $\sigma_2 x^2 + \sigma_1 x + 1 = 0$ has two roots: x_1, x_2 such that $x_1^{-1} = X_1 = \alpha^{-j_1} = \alpha^{8-j_1}, x_2^{-1} = X_2 = \alpha^{-j_2} = \alpha^{8-j_2}$.

If $X_1 = \alpha^{-j_1}, X_2 = \alpha^{-j_2}$, then the errors are in the digits of z'_{8-j_1}, z'_{8-j_2} the received word Z' , respectively.

Next, the values Y_1, Y_2 of the errors are calculated

$$\begin{vmatrix} Y_1 \\ Y_2 \end{vmatrix} = \begin{vmatrix} X_1 & X_2 \\ X_1^2 & X_2^2 \end{vmatrix}^{-1} \cdot \begin{vmatrix} S_1 \\ S_2 \end{vmatrix} \quad (2)$$

and perform error correction in the word Z'

$$z'_{8-j_1} = (z'_{8-j_1} - Y_1) \bmod 3, z'_{8-j_2} = (z'_{8-j_2} - Y_2) \bmod 3.$$

If $\det M = 0$ (one error in the accepted word Z'), then the root x of the polynomial $\sigma x + 1$ is found, where $\sigma = -S_2/S_1$. The equation $\sigma x + 1 = 0$ is also solved by Chien algorithm. The solution is $x = \alpha^j$ such that x^{-1} is the error locator ($X = x^{-1} = \alpha^{-j} = \alpha^{8-j}$) and the location of the error is the digit numbered $8-j$ (that is z'_{8-j}).

Let's consider an example of correcting two errors in a read BC-pattern.

Let's assume that BC-pattern was printed on carrier, the vector of which was equal to $Z = (2 \ 0 \ 2 \ 1 \ 1 \ 0 \ 1 \ 2)$.

Let a vector be obtained during the reading of this pattern is

$$\begin{matrix} z'_0 & z'_1 & z'_2 & z'_3 & z'_4 & z'_5 & z'_6 & z'_7 \\ Z' = (\underline{1} & 0 & \underline{2} & \underline{2} & 1 & 0 & 1 & 2), \end{matrix}$$

which contains two errors (underlined units).

Let's calculate syndrome components S_1, S_2, S_3, S_4 :

$$\begin{aligned} S_1 &= 1 \binom{0}{1} + 0 \binom{1}{0} + 2 \binom{2}{1} + 2 \binom{2}{2} + 1 \binom{0}{2} + 0 \binom{2}{0} + 1 \binom{1}{2} + 2 \binom{1}{1} = \binom{2}{1} = \alpha^2, \\ S_2 &= 1 \binom{0}{1} + 0 \binom{2}{1} + 2 \binom{0}{2} + 2 \binom{1}{2} + 1 \binom{0}{1} + 0 \binom{2}{1} + 1 \binom{0}{2} + 2 \binom{1}{2} = \binom{1}{1} = \alpha^7, \\ S_3 &= 1 \binom{0}{1} + 0 \binom{2}{2} + 2 \binom{1}{2} + 2 \binom{1}{0} + 1 \binom{0}{2} + 0 \binom{1}{1} + 1 \binom{2}{2} + 2 \binom{2}{0} = \binom{1}{2} = \alpha^6, \\ S_4 &= 1 \binom{0}{1} + 0 \binom{0}{2} + 2 \binom{0}{1} + 2 \binom{0}{2} + 1 \binom{0}{1} + 0 \binom{0}{2} + 1 \binom{0}{1} + 2 \binom{0}{2} = \binom{0}{1} = \alpha^0. \end{aligned}$$

Let's put it together the matrix

$$M = \begin{vmatrix} S_1 & S_2 \\ S_2 & S_3 \end{vmatrix} = \begin{vmatrix} \alpha^2 & \alpha^7 \\ \alpha^7 & \alpha^6 \end{vmatrix}.$$

Let's calculate the determinant of the matrix M : $\det M = \alpha^2 \alpha^6 - \alpha^7 \alpha^7 = \alpha^8 - \alpha^{14} = \alpha^0 - \alpha^6 = (0, 1) - (1, 2) = (0, 1) + (2, 1) = (2, 2) = \alpha^3 \neq 0$ (see Table 2).

Since $\det M \neq 0$, there are two errors in the word Z' . Let's find the coefficients σ_2, σ_1 of the error locator polynomial $\sigma(x) = \sigma_2 x^2 + \sigma_1 x + 1$:

$$\begin{vmatrix} \sigma_2 \\ \sigma_1 \end{vmatrix} = M^{-1} \begin{vmatrix} -S_3 \\ -S_4 \end{vmatrix}.$$

To do this, first calculate M^{-1} :

$$M^{-1} = (1/\det M) \cdot \begin{vmatrix} S_3 & -S_2 \\ -S_2 & S_1 \end{vmatrix} = (1/\alpha^3) \cdot \begin{vmatrix} \alpha^6 & -\alpha^7 \\ -\alpha^7 & \alpha^2 \end{vmatrix} = \begin{vmatrix} \alpha^3 & -\alpha^4 \\ -\alpha^4 & \alpha^{-1} \end{vmatrix}$$

Since $-\alpha^4 = -(0, 2) = (0, 1) = \alpha^0$, and $\alpha^{-1} = \alpha^7$, then

$$M^{-1} = \begin{vmatrix} \alpha^3 & \alpha^0 \\ \alpha^0 & \alpha^7 \end{vmatrix}$$

$$\text{Then, } \begin{vmatrix} \sigma_2 \\ \sigma_1 \end{vmatrix} = \begin{vmatrix} \alpha^3 & \alpha^0 \\ \alpha^0 & \alpha^7 \end{vmatrix} \cdot \begin{vmatrix} -\alpha^6 \\ -\alpha^0 \end{vmatrix} = \begin{vmatrix} \alpha^3 & \alpha^0 \\ \alpha^0 & \alpha^7 \end{vmatrix} \cdot \begin{vmatrix} \alpha^2 \\ \alpha^4 \end{vmatrix} = \begin{vmatrix} \alpha^3 \\ \alpha^1 \end{vmatrix}$$

(since $-\alpha^6 = -(1, 2) = (2, 1) = \alpha^2$, and $-\alpha^0 = -(0, 1) = (0, 2) = \alpha^4$).
Next, we will solve the equation $\alpha^3 x^2 + \alpha^1 x + 1 = 0$ in the field $GF(3^2)$.

For this, we will apply Chien algorithm:

$$\begin{aligned} x = \alpha^0 &\rightarrow \alpha^3(\alpha^0)^2 + \alpha^1(\alpha^0) + 1 = \alpha^3 + \alpha^1 + 1 = (2, 2) + (1, 0) + 1 = (0, 0) = 0, \\ x = \alpha^1 &\rightarrow \alpha^3(\alpha^1)^2 + \alpha^1(\alpha^1) + 1 = \alpha^5 + \alpha^2 + 1 = (2, 0) + (2, 1) + 1 = (1, 2) \neq 0, \\ x = \alpha^2 &\rightarrow \alpha^3(\alpha^2)^2 + \alpha^1(\alpha^2) + 1 = \alpha^7 + \alpha^3 + 1 = (1, 1) + (2, 2) + 1 = (0, 1) \neq 0, \\ x = \alpha^3 &\rightarrow \alpha^3(\alpha^3)^2 + \alpha^1(\alpha^3) + 1 = \alpha^1 + \alpha^4 + 1 = (1, 0) + (0, 2) + 1 = (1, 0) \neq 0, \\ x = \alpha^4 &\rightarrow \alpha^3(\alpha^4)^2 + \alpha^1(\alpha^4) + 1 = \alpha^5 + \alpha^5 + 1 = (2, 2) + (2, 0) + 1 = (1, 0) \neq 0, \\ x = \alpha^5 &\rightarrow \alpha^3(\alpha^5)^2 + \alpha^1(\alpha^5) + 1 = \alpha^5 + \alpha^6 + 1 = (2, 0) + (1, 2) + 1 = (0, 0) = 0, \\ x = \alpha^6 &\rightarrow \alpha^3(\alpha^6)^2 + \alpha^1(\alpha^6) + 1 = \alpha^7 + \alpha^7 + 1 = (1, 1) + (1, 1) + 1 = (2, 0) \neq 0, \\ x = \alpha^7 &\rightarrow \alpha^3(\alpha^7)^2 + \alpha^1(\alpha^7) + 1 = \alpha^1 + \alpha^0 + 1 = (1, 0) + (0, 1) + 1 = (1, 2) \neq 0. \end{aligned}$$

As we can see, the roots of the equation are $x_1 = \alpha^0$ and $x_2 = \alpha^5$.

So, $X_1 = x_1^{-1} = \alpha^0 = \alpha^{8-0} = \alpha^8 = \alpha^0$, and

$X_2 = x_2^{-1} = \alpha^{-5} = \alpha^{8-5} = \alpha^3$. This means that errors locate in digits z'_0 and z'_3 of accepted word.

Let's calculate the error values based on (2):

$$\begin{vmatrix} Y_1 \\ Y_2 \end{vmatrix} = \begin{vmatrix} \alpha^0 & \alpha^3 \\ \alpha^0 & \alpha^6 \end{vmatrix}^{-1} \cdot \begin{vmatrix} \alpha^2 \\ \alpha^7 \end{vmatrix}$$

First, we find the inverse matrix:

$$\begin{aligned} \begin{vmatrix} \alpha^0 & \alpha^3 \\ \alpha^0 & \alpha^6 \end{vmatrix}^{-1} &= (1/\alpha^5) \cdot \begin{vmatrix} \alpha^6 & -\alpha^3 \\ -\alpha^0 & \alpha^0 \end{vmatrix} = (1/\alpha^5) \cdot \begin{vmatrix} \alpha^6 & \alpha^7 \\ \alpha^4 & \alpha^0 \end{vmatrix} \\ &= \begin{vmatrix} \alpha^1 & \alpha^2 \\ \alpha^{-1} & \alpha^{-5} \end{vmatrix} = \begin{vmatrix} \alpha^1 & \alpha^2 \\ \alpha^7 & \alpha^3 \end{vmatrix} \end{aligned}$$

$$\text{Then } \begin{vmatrix} Y_1 \\ Y_2 \end{vmatrix} = \begin{vmatrix} \alpha^1 & \alpha^2 \\ \alpha^7 & \alpha^3 \end{vmatrix} \cdot \begin{vmatrix} \alpha^2 \\ \alpha^7 \end{vmatrix} = \begin{vmatrix} \alpha^4 \\ \alpha^0 \end{vmatrix}$$

But $\alpha^4 = (0, 2) = 2$, $\alpha^0 = (0, 1) = 1$.

Therefore, $Y_1 = 2$, $Y_2 = 1$.

Let's correct the errors:

$$z_0 = (z'_0 - Y_1) \bmod 3 = (1 - 2) \bmod 3 = 2$$

$$z_3 = (z'_3 - Y_2) \bmod 3 = (2 - 1) \bmod 3 = 1.$$

Thus, the right vector of the read BC-pattern is $Z = (2 \ 0 \ 2 \ 1 \ 1 \ 0 \ 1 \ 2)$; double error is corrected.

To construct the symbologies of multi-color BCs of different capacities, shortened BCH code should be used. To obtain shortened BCH code it is needed to remove the required number of columns and rows from the original generator matrix and appropriate number of columns in check matrix.

We will consider the construction of shortened codes on the example of a full triple (26, 17)-BCH code (see Table 1), which uses two fields: $GF(3)$ – the field of characters (Fig. 2) and $GF(3^3)$ – the field of locators (Table 4). The field of locators is built on the basis of an irreducible polynomial of third degree $p(x) = x^3 + 2x + 1$.

In $GF(3^3)$ $\alpha^i = \alpha^{i-26}$, $\alpha^{-i} = \alpha^{26-i}$, $\alpha^{26} = \alpha^{-26} = \alpha^0 = 1$.

Table 4 – Elements of the field $GF(3^3)$ and their minimal polynomials (fragment)

Exponential notation		Poly-nomial notation	Vector notation	Deci-mal notation	The minimal polyno-mial
With a non-negative degree of the primitive element α of the field	With a negative degree of primitive element α of the field				
–	–	0	(0,0,0)	0	–
α^0	α^{-26}	1	(0,0,1)	1	–
α^1	α^{-25}	α	(0,1,0)	3	x^3+2x+1
α^2	α^{-24}	α^2	(1,0,0)	9	x^3+x^2+x+2
α^3	α^{-23}	$\alpha+2$	(0,1,2)	5	x^3+2x+1
α^4	α^{-22}	$\alpha^2+2\alpha$	(1,2,0)	15	x^3+x^2+2
α^5	α^{-21}	$2\alpha^2+\alpha+2$	(2,1,2)	23	x^3+x^2+x+1
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
α^{25}	α^{-1}	$2\alpha^2+1$	(2,0,1)	19	x^3+2x^2+1

The generator polynomial $g(x)$ of the ternary (26, 17)-BCH code is defined as follows: $g(x) = \text{LCM}(x^3+2x+1, x^3+x^2+x+2, x^3+2x+1, x^3+x^2+2) = (x^3+2x+1)(x^3+x^2+x+2)(x^3+x^2+2) = x^9+2x^8+x^7+x^6+x^5+2x^4+2x^3+2x^2+x+1 \rightarrow g_0 g_1 \dots g_9 = 1 \ 1 \ 2 \ 2 \ 2 \ 1 \ 1 \ 1 \ 2 \ 1$.

It corresponds to the generator matrix $G_{(26, 17)}$ of the full code.

If, for example, we remove 10 columns to the right ($z_{16} - z_{25}$) and 10 bottom rows from $G_{(26, 17)}$; and 10 columns to the right ($z'_{16} - z'_{25}$) from $H_{(26, 17)}$ then we get generator matrix $G_{(16, 7)}$ and, accordingly, check matrix $H_{(16, 7)}$, of shortened (16, 7)-BCH code (Fig. 5).

Moving the right columns and bottom rows from $G_{(26, 17)}$, and the corresponding right columns from $H_{(26, 17)}$, we will obtain different shortened triple ($q=3$) BCH codes: (17, 8)-; (16, 7)-; (15, 6)-; (14, 5)-; (13, 4)-, on the basis of which it is possible to synthesize the symbologies of interference-resistant three-colored BCs of different capacities (Table 5).

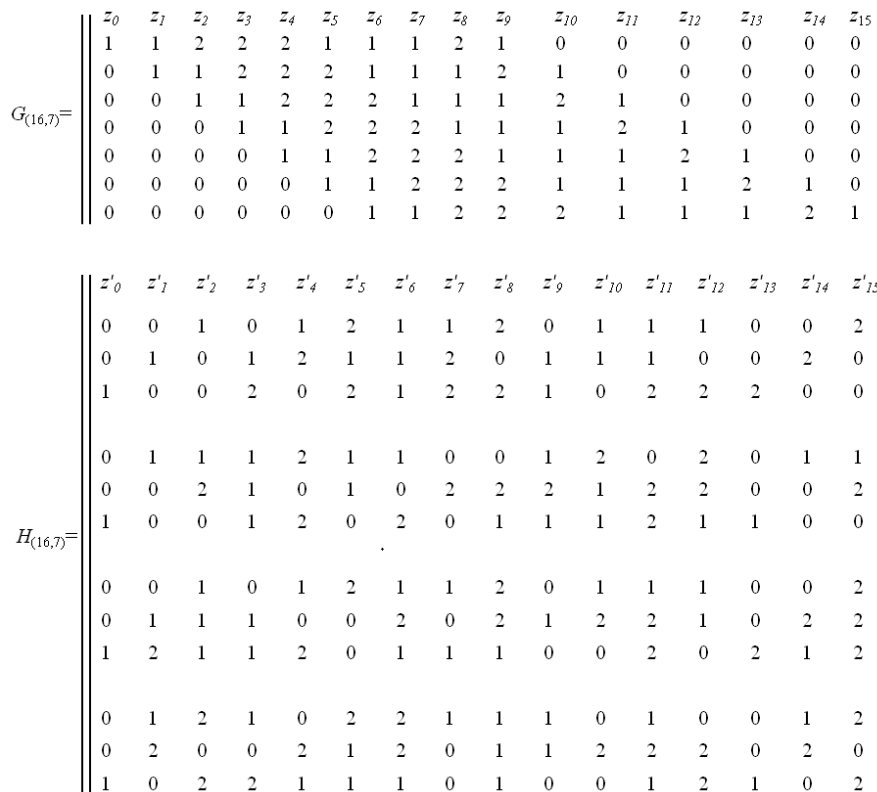


Figure 5 – Generator (G) and check (H) matrices of the shortened ternary (16, 7)-BCH code based on the full (26, 17)-BCH code

Table 5 – Capacity (V) of the symbolologies of multi-color (q) interference-resistant BCs based on shortened (s, u)-BCH codes

$q = 3$		$q = 5$		$q = 7$	
(s, u) –	V	(s, u) –	V	(s, u) –	V
(13, 4)–	81	(10, 2)–	25	(10, 2)–	49
(14, 5)–	243	(11, 3)–	125	(11, 3)–	343
(15, 6)–	729	(12, 4)–	625	(12, 4)–	2401
(16, 7)–	2187	(13, 5)–	3125	(13, 5)–	16807
(17, 8)–	6561	(14, 6)–	15625		

Similarly, on the basis of shortened quinary ($q=5$) and septenary ($q=7$) BCH codes, it is possible to synthesize the symbolologies of interference-resistant five-color and seven-color BCs with the possibility of correction double errors in BC-patterns. So, assigned in the Table 5 shortened quinary codes, formed on the basis of the full quinary (24, 16)-BCH code (see Table 1), which uses the field of characters $GF(5)$ and the field of locators $GF(5^2)$, and the shortened septenary codes, formed on the basis of full septenary (48, 40)-BCH code, which uses the character field $GF(7)$ and the locator field $GF(7^2)$.

Such a series of BCH codes makes it possible to build the family of multi-color interference-resistant BCs with symbolologies of different capacity.

4 EXPERIMENTS

The considered BCH codes with a minimal code distance of $d_{\min}=5$ ensure the correction of a single or double error inside each BC-pattern when read from the carrier. In order to explore the correction capabilities of shortened BCH codes, in particular, the ability to detect multiple

errors in BC-patterns, a software product was developed in Java in the environment IntelliJ idea.

Experimental studies were carried out on a computer with the macOS operating system BigSur, 32 GB RAM, 2.4 GHz 8-Core processor Intel Core i9.

This software product makes it possible to carry out statistical studies of the corrective ability of BCH codes in conditions of multiple damages to the elements of BC-patterns. All possible cases of occurrence of one to seven errors in BC-patterns were studied. For each case, one of three possible events were recorded: an error is detected (for single and double error detection is equivalent to correction – according to the algorithm in Fig. 4); error syndrome is equal to zero; the combination of errors is undetected.

Not only all possible locations in words of probable errors, but also all possible values of errors were generated.

5 RESULTS

Statistical data characterizing the ability of multi-valued BCH codes to detect and correct multiple errors was obtained. It has been proven that all single and double errors in data words (vectors of BC-patterns) are corrected. The ability to detect (3–7)-tuple errors for 18 BCH codes – 8 ternary codes, 6 quinary codes, and 4 septenary codes – was studied. For each code, corresponding indicators were obtained – for example, for the full ternary (8, 3)-BCH code, they are presented in the Table 6.

Generalized indicators for ternary BCH-codes are reflected in Fig. 6 (the abscissa axis indicates the investigated ternary (s, u)-BCH codes, the ordinate axis indicates

the percentage of detected errors), where the upper curve is the percentage of 3-tuple errors that are detected, and the lower curve is the percentage of detection (4–7)-tuple errors.

Table 6 – Corrective ability of the full ternary (8, 3)-BCH code in the case of multiple errors

Multiplicity of error	The percentage of errors which		
	are detected	are undetectable	give zero syndrome
3	46.4%	53.6%	0
4	39.3%	60.7%	0
5	29.5%	69.7%	0.8%
6	32.6%	66.9%	0.5%
7	34.4%	64.9%	0.7%

Quinary BCH codes allow to detect of 70.7 – 96.0% of 3-tuple errors and 67.9–93.8% (4–7)-tuple errors; septenary BCH codes, respectively, 97.9–99.0% 3-tuple errors and 97.1–98.3% (4–6)-tuple in data words.

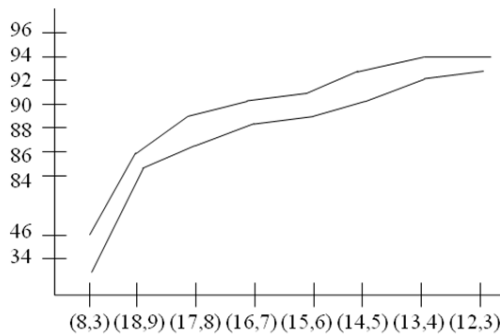


Figure 6 – Ability to detect multiple errors with ternary codes

6 DISCUSSION

Research shows that full BCH codes, such as a ternary (8, 3)-code or a quinary (24, 16)-code, provide fewer multiple-error detections compared to shortened codes. This is because shortened codes have more redundancy. It was also found out that combinations of multiple errors, which give a zero syndrome during decoding, are extremely rare (0.004 – 0.009%) precisely in shortened BCH codes, and they are not detected by the decoder.

The obtained results are highly reliable, since the verification of the proposed algorithm for decoding BCH codes was carried out on data words of different bit sizes, with an overview of all possible combinations of errors that may occur during data processing, as well as for a sufficiently large number of codes.

The obtained results give reason to conclude about the expediency of the two-level security of multi-color barcode images, when the lower level should be based on the use of a multi-valued BCH code (the level of the BC-pattern; the digital vector of the BC-pattern is the codeword of the BCH code), and the upper level (the BC-symbol in general) – on the use of the Reed-Solomon code, which is capable to correct two types of distortion – errors and erasures, and for which the minimal structural units (word bits) are the BC-patterns of the barcode im-

age. For the Reed-Solomon code, an “error” is considered to be a situation when neither the location of the distortion in the read word (BC-symbol) nor the value of distortion is unknown; and “erasure” is a situation when the location of the distortion is known, and only the value of the distortion is not known.

When reading a multi-color barcode, the program sequentially separates BC-patterns from the barcode image, thus forming a Reed-Solomon codeword, in which each BC-pattern is a separate digit of the word. A software decoder of the BCH-code operates inside every BC-pattern, the result of which can be three conclusions: “BC-pattern is not damaged”, “BC-pattern is corrected”, “BC-pattern is erased”.

The solution “BC-pattern is not damaged” is formed if the error syndrome is equal to zero. It should be noted that those rare cases (the number of which is less than one-hundredth of a percent) when combinations of element distortions in the BC-patterns give zero syndrome, and may occur with an error multiplicity of more than four, will be detected by the software decoder of the Reed-Solomon code.

The solution “BC-pattern is corrected” is formed when the BCH code decoder corrects a one- or two errors in the BC-pattern vector. If the BC-pattern contains three or more damages, and the decoder of the BCH code perceives them as a single or double error, and, accordingly, will correct it incorrectly (and in such cases, for example, for a ternary (15, 6)-code or a quinary (13, 5)-code, – about 8–10%), then such a BC-pattern will be perceived by the decoder of the Reed-Solomon code as an “error”.

The solution “BC-pattern is erased” is formed by the BCH code decoder, if it detects distortion of three or more elements in the BC-pattern. For the (15, 6)- and (13, 5)-BCH codes mentioned above, this will happen in 90 – 92% of cases. Such a situation would be qualified by a Reed-Solomon code decoder as “erasure”.

It is known that to correct each error in the structure of the codeword of the Reed-Solomon code, two check digits must be provided, and to detect each erasure – one check digit [15]. Therefore, the use of a multi-valued, for example, three-color (15, 6)-BCH code at the lower level of ensuring interference resistance of a multi-color BC, which, in addition to correcting single and double damage, also allows to detect about 90% of multiple (three or more) damages of elements in BC-pattern, strengthens the corrective capabilities of the Reed-Solomon code by an average of 45% by transferring “erasure” situations instead of “error” situations to the upper level of immunity protection.

CONCLUSIONS

The work solves the actual scientific problem of improving the interference resistance of multi-color barcodes.

The scientific novelty of the work lies in the fact that the method of constructing the symbology of a given capacity of a multi-color barcode is firstly proposed, the barcode patterns of which have the properties of interfer-

ence resistance, which is consist in the fact that when reading a multi-color barcode image the data will be reliably reproduced from the barcode patterns even in case of damage one or two graphic elements of the pattern. This is achieved due to the fact that the vector (digital twin) of each barcode pattern is a codeword of the correcting multi-valued BCH code with a minimal code distance of five.

The construction (synthesis) of BC symbologies is considered in detail for the case of three-, five- and seven-color two-dimensional barcodes.

It is shown that in the BC-patterns synthesized on the basis of the BCH code, during their reading from the carrier, in addition to the correction of single and double errors, a significant part (from 33.9% to 97.1%) of (3 – 7)-tuple errors is also detected. It has been proven that the use of shortened multi-valued BCH codes for the synthesis of symbologies of multi-color barcodes significantly increases the ability to detect multiple errors in BC-patterns compared to full codes, in particular by 1.91 – 2.75 times – for three-color ones and by 1.30–1.38 times – for five-color BC-patterns, and also allows to receive symbologies of different capacity, which makes it possible to create family of multi-color interference-resistant barcodes.

The practical significance of the obtained results lies in the fact that the developed method of constructing interference-resistant BC-patterns based on BCH codes can be used at the lower level in the system of two-level interference resistance of multi-color barcode images, when the Reed-Solomon code is used at the upper level. At the same time, the corrective capabilities of the Reed-Solomon code are significantly strengthened (up to 45%) with an unchanged number of control digits, as a result of which the immunity of barcodes patterns of multi-color barcodes is significantly improved.

Prospects for further research should be focused on improving the mechanism of complementary application of correcting codes for two-level interference immunity of multi-color barcode images.

ACKNOWLEDGEMENTS

The research was carried out within the framework of the state-budget scientific research work “Mathematical and software methods for processing multimodal data of monitoring of medical and biological objects for the diagnosis of the health status of patients” of the National Technical University of Ukraine “Ihor Sikorsky Kyiv Polytechnic Institute” (state registration number 0120U 102134).

REFERENCES

1. Parikh D., Jancke G. Localization and Segmentation of a 2D High Capacity Color Barcode, *Workshop on Application of Computer Vision (WACV'08) : Copper Mountain, CO, USA, January 7–9, 2008, IEEE proceedings*, 2008, pp. 1–6. DOI : <https://doi.org/10.1109/WACV.2008.4544033>
2. Wang F., Manduchi R. Color-constant information embedding, *Trends and Topics in Computer Vision*. Springer, 2012, Vol. 6554, pp. 13–26. DOI : https://doi.org/10.1007/978-3-642-35740-4_2

© Sulema Ye. S., Drozdenko L. V., Dychka A. I., 2022
DOI 10.15588/1607-3274-2022-4-9

3. Bagherinia H., Manduchi R. High information rate and efficient color barcode decoding, *Lecture Notes in Computer Science*. Springer, 2012, Vol. 7584, pp. 482–491. DOI : https://doi.org/10.1007/978-3-642-33868-7_48
4. Blasinski H., Bulan O., Sharma G. Per-colorant-channel color barcodes for mobile applications: an interference cancellation framework, *Transactions on Image Processing, IEEE*, 2013, Vol. 22(4), pp. 1498–1511. DOI : <https://doi.org/10.1109/TIP.2012.2233483>
5. Grillo A., Lentini A., Querini M., Italiano G. F. High capacity colored two dimensional codes, *International Multiconference on Computer Science and Information Technology*. Wisla, Poland, October 18–20, 2010, IEEE proceedings, 2011, pp. 709–716. DOI : <https://doi.org/10.1109/IMCSIT.2010.5679869>
6. Querini M., Italiano G. F. Color Classifiers for 2D Color Barcodes, *Federal Conference on Computer Science and Information Systems*. Krakow, Poland, September 8–11, 2013, IEEE proceedings, 2013, pp. 611–618.
7. Ramya M., Jayasheela M. Color QR Codes for Real Time Applications with High Embedding Capacity, *International Journal of Computer Application*, 2014, Vol. 91(8), pp. 8–12. DOI : <https://doi.org/10.5120/15899-4889>
8. Abas A., Yusof Y., Din R., Azali F., Osman B. Increasing data storage of coloured QR code using compress, multiplexing and multilayered technique, *Bulletin of Electrical Engineering and Informatics*, 2020, Vol. 9(6), pp. 2555–2561. DOI : <https://doi.org/10.11591/eei.v9i6.2481>
9. Zhurakovskiy B. Yu., Druzhynin V. A. Bahatovymirni shtryhovi kody, *Mizhvidomchyi naukovо-tekhnichnyi zbirnyk “Adaptyvni systemy avtomatychnoho upravlinnya”*, 2018, Vol. 2(33), pp. 15–31. DOI : <https://doi.org/10.20535/1560-8956.33.2018.164669>
10. Dychka I., Onai M., Sulema O. Data Compression in Black-Gray-White Barcoding, *Radio Rlectronics, Computer Science, Control*, 2020, Vol.1, pp. 125–134. DOI : <https://doi.org/10.15588/1607-3274-2020-1-13>
11. Pang P., Wu J., Long C. CodeCube: A Multi-Layer Color Barcode for Mobile Social Applications, *The 29th Chinese Control and Decision Conference (CCDC). Chongqing, China, May 28–30, 2017, IEEE proceedings*, 2017, pp. 7713–7718. DOI : <https://doi.org/10.1109/CCDC.2017.7978590>
12. Berchtold W., Liu H., Steinebach M., Klein D., Senger T., Thence N. JAB Code – A Versatile Polychrome 2D Barcode *Electronic Imaging*, 2020, Vol. 2020(3), pp. 207–212. DOI : <https://doi.org/10.2352/ISSN.2470-1173.2020.3.MOBMU-207>
13. Wang G., Yang Z., Chen J. Security Mechanism Improvement for 2D Barcodes using Error Correction with Random Segmentation, *The 6th International Conference on Information Technology: IoT and Smart City*. Bhubaneswar, India, December 19–21, 2018, proceedings, IEEE, 2018, pp. 104–109. DOI : <https://doi.org/10.1145/3301551.3301593>
14. Sulema Ye. S., Onai M. V., Dychka A. I. Algoritmichne zabezpechennya zavodostiykosti bahatokolirnyh shtryhkodovyh znakov na osnovi polya GF(p), *Naukovi visti KPI*, 2021, Vol. 1(132), pp. 50–62. DOI : <https://doi.org/10.20535/kpissn.2021.1.231210>
15. Blahut R. E. Theory and Practice of Error Control Codes. Addison – Wesley, 1983, P. 576.

Received 26.09.2022.
Accepted 12.11.2022.

УДК 004.627

СИНТЕЗ СИМВОЛІК БАГАТОКОЛІРНИХ ЗАВАДОСТІЙКИХ ШТРИХОВИХ КОДІВ НА ОСНОВІ МНОГОЗНАЧНИХ КОДІВ БЧХ

Сулема Є. С. – д-р техн. наук, доцент, завідувач кафедри програмного забезпечення комп'ютерних систем Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна.

Дрозденко Л. В. – асистент кафедри програмного забезпечення комп'ютерних систем Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна.

Дичка А. І. – аспірант кафедри програмного забезпечення комп'ютерних систем Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна.

АНОТАЦІЯ

Актуальність. Розглянуто задачу побудови набору (символіки) штрихкодів знаків для багатоколірних штрихових кодів, стійких до ушкодження одного або двох елементів у межах кожного знака.

Мета. Забезпечення надійності зчитування багатоколірних штрихкодів зображень.

Метод. Багатоколірний штрихкодів знак має властивість завадостійкості, якщо його цифровий еквівалент (вектор) є кодовим словом многозначного (недвійкового) коректувального коду, здатного виправляти помилки (спотворення елементів знака). Показано, що побудову штрихкодів знаків слід виконувати на основі многозначного коректувального коду БЧХ, здатного виправляти дві помилки. Запропоновано метод побудови множини завадостійких штрихкодів знаків заданої потужності, які забезпечують достовірне відтворення даних при їх зчитуванні з носія. Розроблено процедуру кодування даних многозначним кодом БЧХ на основі твірної матриці коду з використанням операцій за модулем простого числа. Запропоновано новий спосіб побудови перевірної матриці многозначного коду БЧХ на основі векторного подання елементів скінченного поля. Розроблено узагальнений алгоритм генерування символіки багатоколірного штрихового коду з можливістю корекції двократних помилок у штрихкодів знаках. Метод також дозволяє будувати символіки заданої потужності на основі скорочених кодів БЧХ. Запропоновано спосіб скорочення твірної та перевірної матриць многозначного повного коду БЧХ для отримання скороченого коду заданої довжини. Показано, що крім виправлення двократних помилок, многозначні коди БЧХ дозволяють також виявляти помилки більшої кратності; ця властивість посилюється при використанні скорочених кодів БЧХ. Метод забезпечує побудову сімейства багатоколірних завадостійких штрихових кодів.

Результати. На основі розробленого програмного забезпечення отримані статистичні дані, що характеризують здатність многозначних кодів БЧХ виявляти та виправляти помилки, і на їх основі проєктувати багатоколірні завадостійкі штрихові коди.

Висновки. Проведені експерименти підтвердили працездатність розробленого алгоритмічного забезпечення і дозволяють рекомендувати його для використання на практиці при проєктуванні завадостійких багатоколірних штрихових кодів у системах автоматичної ідентифікації.

КЛЮЧОВІ СЛОВА: штрихове кодування, багатоколірні штрихові коди, завадостійкість штрихових кодів, коди БЧХ.

УДК 004.627

СИНТЕЗ СИМВОЛІК МНОГОЦВЕТНЫХ ПОМЕХОУСТОЙЧИВЫХ ШТРИХОВЫХ КОДОВ НА ОСНОВЕ МНОГОЗНАЧНЫХ КОДОВ БЧХ

Сулема Е. С. – д-р техн. наук, доцент, заведующий кафедрой программного обеспечения компьютерных систем Национального технического университета Украины «Киевский политехнический институт имени Игоря Сикорского», Киев, Украина.

Дрозденко Л. В. – ассистент кафедры программного обеспечения компьютерных систем Национального технического университета Украины «Киевский политехнический институт имени Игоря Сикорского», Киев, Украина.

Дичка А. И. – аспирант кафедры программного обеспечения компьютерных систем Национального технического университета Украины «Киевский политехнический институт имени Игоря Сикорского», Киев, Украина.

АННОТАЦИЯ

Актуальность. Рассмотрена задача построения набора штрихкодированных знаков для многоцветных штриховых кодов, устойчивых к искажениям одного или двух элементов в пределах каждого знака.

Цель. Обеспечение надежности считывания многоцветных штрихкодированных изображений.

Метод. Многоцветный штрихкодированный знак имеет свойство помехоустойчивости, если его цифровой эквивалент (вектор) является кодовым словом многозначного (недвоичного) корректирующего кода, способного исправлять ошибки (искажения элементов знака). Показано, что построение штрихкодированных знаков следует выполнять на основе многозначного корректирующего кода БЧХ, способного исправлять две ошибки. Предложен метод построения множества помехоустойчивых штрихкодированных знаков заданной мощности, обеспечивающих достоверное воспроизведение данных при их считывании с носителя. Разработана процедура кодирования данных многозначным кодом БЧХ на основе образующей матрицы кода с использованием операций по модулю простого числа. Предложен новый способ построения проверочной матрицы многозначного кода БЧХ на основе векторного представления элементов конечного поля. Разработан обобщенный алгоритм генерирования символіки многоцветного штрихового кода с возможностью коррекции двукратных ошибок в штрихкодированных знаках. Метод позволяет строить символіки заданной мощности на основе сокращенных кодов БЧХ. Предложен способ сокращения образующей и проверочной матрицы многозначного полного кода БЧХ для получения сокращенного кода за-

данной длины. Показано, что кроме исправления двукратных ошибок, многозначные коды БЧХ позволяют также обнаруживать ошибки большей кратности – это свойство усиливается при использовании укороченных кодов БЧХ. Метод обеспечивает построение семейства многоцветных помехоустойчивых штриховых кодов.

Результаты. На основе разработанного программного обеспечения получены статистические данные, характеризующие способность многозначных кодов БЧХ обнаруживать и исправлять ошибки, и на их основе проектировать многоцветные помехоустойчивые штриховые коды.

Выводы. Проведенные эксперименты подтвердили работоспособность разработанного алгоритмического обеспечения и позволяют рекомендовать его для использования на практике при проектировании помехоустойчивых многоцветных штриховых кодов в системах автоматической идентификации.

КЛЮЧЕВЫЕ СЛОВА: штриховое кодирование, многоцветные штриховые коды, помехоустойчивость штриховых кодов, коды БЧХ.

ЛИТЕРАТУРА / LITERATURA

1. Parikh D. Localization and Segmentation of a 2D High Capacity Color Barcode / D. Parikh, G. Jancke // Workshop on Application of Computer Vision (WACV'08) : Copper Mountain, CO, USA, January 7–9, 2008 : IEEE proceedings. – 2008. – P. 1–6. DOI : <https://doi.org/10.1109/WACV.2008.4544033>
2. Wang F. Color-constant information embedding / F. Wang, R. Manduchi // Trends and Topics in Computer Vision. – Springer, 2012. – Vol. 6554. – P. 13–26. DOI : https://doi.org/10.1007/978-3-642-35740-4_2
3. Bagherinia H. High information rate and efficient color barcode decoding / H. Bagherinia, R. Manduchi // Lecture Notes in Computer Science. – Springer, 2012. – Vol. 7584. – P. 482–491. DOI : https://doi.org/10.1007/978-3-642-33868-7_48
4. Blasinski H. Per-colorant-channel color barcodes for mobile applications: an interference cancellation framework / H. Blasinski, O. Bulan, G. Sharma // Transactions on Image Processing. – IEEE, 2013. – Vol. 22(4). – P. 1498–1511. DOI : <https://doi.org/10.1109/TIP.2012.2233483>
5. High capacity colored two dimensional codes / [A. Grillo, A. Lentini, M. Querini, G. F. Italiano] // International Multi-conference on Computer Science and Information Technology : Wisla, Poland, October 18–20, 2010 : IEEE proceedings. – 2011. – P. 709–716. DOI : <https://doi.org/10.1109/IMCSIT.2010.5679869>
6. Querini M. Color Classifiers for 2D Color Barcodes / M. Querini, G. F. Italiano // Federal Conference on Computer Science and Information Systems : Krakow, Poland, September 8–11, 2013 : IEEE proceedings. – 2013. – P. 611–618.
7. Ramya M. Color QR Codes for Real Time Applications with High Embedding Capacity / M. Ramya, M. Jayasheela // International Journal of Computer Application. – 2014. – Vol. 91(8). – P. 8–12. DOI : <https://doi.org/10.5120/15899-4889>
8. Increasing data storage of coloured QR code using compress, multiplexing and multilayered technique / [A. Abas, Y. Yusof, R. Din et al.] // Bulletin of Electrical Engineering and Informatics. – 2020. – Vol. 9(6). – P. 2555–2561. DOI : <https://doi.org/10.11591/eei.v9i6.2481>
9. Жураковський Б. Ю. Багатомірні штрихові коди / Б. Ю. Жураковський, В. А. Дружинін // Міжвідомчий науково-технічний збірник «Адаптивні системи автоматичного управління». – 2018. – № 2(33). – С. 15–31. DOI:<https://doi.org/10.20535/1560-8956.33.2018.164669>
10. Dychka I. Data Compression in Black-Gray-White Barcoding / I. Dychka, M. Onai, O. Sulema // Radio Electronics, Computer Science, Control. – 2020. – Vol. 1. – P. 125–134. DOI : <https://doi.org/10.15588/1607-3274-2020-1-13>
11. Pang P. CodeCube: A Multi-Layer Color Barcode for Mobile Social Applications / Pang P., Wu J., Long C. // The 29th Chinese Control and Decision Conference (CCDC) : Chongqing, China, May 28–30, 2017 : IEEE proceedings. – 2017. – P. 7713–7718. DOI: <https://doi.org/10.1109/CCDC.2017.7978590>
12. JAB Code – A Versatile Polychrome 2D Barcode / [W. Berchtold, H. Liu, M. Steinebach et al.] // Electronic Imaging. – 2020. – Vol. 2020(3). – P. 207–212. DOI : <https://doi.org/10.2352/ISSN.2470-1173.2020.3.MOBMU-207>
13. Wang G. Security Mechanism Improvement for 2D Barcodes using Error Correction with Random Segmentation / G. Wang, Z. Yang, J. Chen // The 6th International Conference on Information Technology: IoT and Smart City : Bhubaneswar, India, December 19–21, 2018 : proceedings. – IEEE, 2018. – P. 104–109. DOI: <https://doi.org/10.1145/3301551.3301593>
14. Сулема С. С. Алгоритмічне забезпечення завадостійкості багатокольорних штрихкодів на основі поля GF(p) / С. С. Сулема, М. В. Онай, А. І. Дичка // Наукові вісті КПП. – 2021. – № 1(132). – С. 50–62. DOI : <https://doi.org/10.20535/kpissn.2021.1.231210>
15. Blahut R. E. Theory and Practice of Error Control Codes / R. E. Blahut. – Addison – Wesley, 1983. – P. 576.

PERMANENT DECOMPOSITION ALGORITHM FOR THE COMBINATORIAL OBJECTS GENERATION

Turbal Y. V. – Dr. Sc., Professor of Computer Science and Applied Mathematics Department, National University of Water and Environmental Engineering, Rivne, Ukraine.

Babych S. V. – Programming Department, College of National University of Life and Environmental Sciences of Ukraine, Rivne, Ukraine.

Kunanets N. E. – Dr. Sc., Professor of Department of Information Systems and Networks, National University “Lviv Polytechnic”, Lviv, Ukraine.

ABSTRACT

Context. The problem of generating vectors consisting of different representatives of a given set of sets is considered. Such problems arise, in particular, in scheduling theory, when scheduling appointments. A special case of this problem is the problem of generating permutations.

Objective. Problem is considered from the point of view of a permanent approach and a well-known one, based on the concept of lexicographic order.

Method. In many tasks, it becomes necessary to generate various combinatorial objects: permutations, combinations with and without repetitions, various subsets. In this paper we consider a new approach to the combinatorial objects generation, which is based on the procedure of the permanent decomposition. Permanent is built for the special matrix of incidence. The main idea of this approach is including to the process of the algebraic permanent decomposition by row additional function for the column identifiers writing into corresponding data structures. In this case, the algebraic permanent is not calculated, but we get a specific recursive algorithm for generating a combinatorial object. The computational complexity of this algorithm is analyzed.

Results. It is investigated a new approach to the generation of complex combinatorial objects, based on the procedure of decomposition of the modified permanent of the incidence matrix by line with memorization of index elements.

Conclusions. The permanent algorithms of the combinatorial objects generation is investigated. The complexity of our approach in the case of permutation is compared with the lexicographic algorithm and the Johnson-Trotter algorithm.

The obtained results showed that our algorithm belongs to the same complexity class as the lexicographic algorithm and the Johnson-Trotter method. Numerical results confirmed the effectiveness of our approach.

KEYWORDS: algorithm, permutation, permanent, decomposition, complexity.

ABBREVIATIONS

JSP is a job-shop problem;

NP-Complete is a nondeterministic polynomial-time complete;

PD-algorithm is a permanent decomposition algorithm;

PD-approach is a permanent decomposition approach;

SDR is a system of different representatives.

NOMENCLATURE

i, j are indices of vectors and matrix elements;

a_j is element of the sets;

S_i is set of the elements;

n_{ij} is the number of occurrences of the element a_j in

the set S_i ;

R_i is a row of the schedule matrix;

$permod$ is modified permanent;

n is size of the array;

$Q(n)$ is computational complexity;

$O()$ is complexity class;

(v_1, v_2, \dots, v_m) is SDR;

v_{ij} is element of the schedule;

$n!$ is factorial number $1*2*...*n$;

e is natural number;

class SDR is special class in C++ notation for storing information about SDR;

SDR() is constructor of the class SDR;

$s, p, next, sizes, sizep, n$ are fields of the class SDR;

${}_p, {}_n, psize$ are parameters of the constructor SDR;

$head$ is first element of the list;

$generic()$ is recursive function for the permutation generation;

$s1, p1$ are additional arrays in the function $generic()$;

\rightarrow is class field access operator via pointer.

INTRODUCTION

Task planning can be defined as a procedure of allocation of resources for a specific job at a specific time.

The most important goal of planning is use of resources. The goal is to minimize waiting time planning. A good time algorithm provides a good system productivity. Problems of combinatorial object generation often arise in computer modeling, cryptography, theory of schedules.

In this paper we consider a new approach to the generation of generalized combinatorial objects of special structure that are well suited for some scheduling tasks (schedule of meetings). Scheduling problems is the most widely studied problems in computer science. There are well known Job-shop scheduling or the job-shop problem (JSP), the nurse operations research problem of finding

an optimal way to assign nurses to shifts, typically with a set of hard and soft constraints. The complexity of the corresponding algorithms in such problems is a critical parameter that allows us to assess the possibility of using a particular algorithm in practice. [1]

At the heart of our approach are procedures for the permanent decomposition of incidence matrices with memorization of identifier elements. We called our approach PD-methods.

The object of study is combinatorial objects generation in the task of Job-shop scheduling.

Despite the large number of publications on the generation of combinatorial objects, the development of new algorithms and approaches is relevant due to their computational complexity.

The subject of study is permanent decomposition algorithms for the combinatorial objects generation.

The purpose of the work is to develop methods for generating combinatorial objects that can be extended to solving complex scheduling problems.

1 PROBLEM STATEMENT

Suppose we have n elements (a_1, a_2, \dots, a_m) , that can be part of m sets (S_1, S_2, \dots, S_m) and the occurrence of the same element several times is allowed. Information about which elements are included in the corresponding sets will be given in the form of an incidence matrix:

$$\begin{pmatrix} & a_1 & a_2 & \dots & a_n \\ S_1 & n_{11} & n_{12} & \dots & n_{1n} \\ S_2 & n_{21} & n_{22} & \dots & n_{2n} \\ \dots & \dots & \dots & \dots & \dots \\ S_m & n_{m1} & n_{m2} & \dots & n_{mn} \end{pmatrix} \quad (1)$$

The elements (a_1, a_2, \dots, a_m) will be called the identifiers of the columns of the incidence matrix. The system of different representatives (SDR) will be called a vector of the form:

$$(v_1, v_2, \dots, v_m), v_i \in S_i, i = \overline{1, m}, v_i \neq v_j, i \neq j.$$

We divide identifier elements on regular and “stream”. If the element a_i is “stream”, then it must be simultaneously written in all positions of the sample vector, where the correspondent incidence matrix column contains non-zero elements. An arbitrary vector of samples (or a matrix, the rows of which are samples) will be called a schedule.

The schedule

$$((v_{11}, v_{12}, \dots, v_{1m}), (v_{21}, v_{22}, \dots, v_{2m}), \dots, (v_{k1}, v_{k2}, \dots, v_{km}))$$

will be considered correct under the conditions:

1. $\forall j \in \{1, 2, \dots, m\}$:
 $\{v_{1j} \cup v_{2j} \cup \dots \cup v_{kj}\} = \{n_{j1} * a_1 \cup n_{j2} * a_2 \cup \dots \cup n_{jn} * a_n\}$,
 $l * a = \{a_1, a_2, \dots, a_l\}, a_i = a, i = \overline{1, l}$.
2. $\forall i \in \{1, 2, \dots, k\} : v_{ij} \neq v_{ir}, j \neq r$, elements v_{ij}, v_{ir} , are non-stream.

Obviously, in the case when each element is included in each set only once and all elements are non-stream, matrix of incidence is

$$\begin{pmatrix} & a_1 & a_2 & \dots & a_n \\ S_1 & 1 & 1 & \dots & 1 \\ S_2 & 1 & 1 & \dots & 1 \\ \dots & \dots & \dots & \dots & \dots \\ S_n & 1 & 1 & \dots & 1 \end{pmatrix} \quad (2)$$

Then one of the variants of the correct schedule can be written in the form:

$$\begin{pmatrix} a_1 & a_2 & a_3 & \dots & a_n \\ a_2 & a_3 & a_4 & \dots & a_1 \\ a_3 & a_4 & a_5 & \dots & a_2 \\ \dots & \dots & \dots & \dots & \dots \\ a_n & a_1 & a_2 & \dots & a_{n-1} \end{pmatrix}. \quad (3)$$

The rows of schedule matrix consist of n permutations of the corresponding elements. The algorithm of cyclic shift of column or row elements is implemented here. Obviously, the number of correct schedules constructed by the cyclic shift algorithm is $n!$. The task of scheduling is very complex, NP-complete. In the general case, to construct all possible variants of correct schedules, it is necessary to analyze all possible SDR variants. Therefore, using any algorithm for solving the problem of generating permutations in the “same place”, it is necessary to additionally store each variant of permutations in memory. Therefore, we must use the most optimal algorithm for generating all possible permutations. In this paper, from the point of view of complexity, an approach is investigated that is based on the use of the procedure for decomposing the permanent of the incidence matrix. It requires the development of new approaches, in particular, to the problems of generating combinatorial objects.

2 REVIEW OF THE LITERATURE

Despite the fact that a significant number of algorithms have been developed to generate various combinatorial objects, such as permutations, permutations with repetitions of different types, systems of subsets of some sets of elements [1–4, 6–13], new approaches and algorithms are still emerging.

Given the novelty of the combination of permanent and decomposition solutions within the calendar calculation – it is difficult to rely on similar literature.

Consider the most relevant areas of application of such solutions.

A significant amount of most recent research has focused on the tasks of scheduling within cloud computing. There are numerous and excellent resources available in the cloud. The cost of performing tasks in the cloud depends on what resources are used. Cloud planning is different from traditional planning. In the environment of cloud computing, the task of scheduling is the biggest and most difficult issue. Task scheduling problem is the NP-complete problem. Many heuristics have introduced scheduling algorithms, but more improvements are needed to make the system faster and more responsive [5].

A detailed overview of the combinatorial algorithms can be given by Knuth [6], Ruskey [7] which considers the concept of combinatorial generation and distinguishes the following tasks: listing-generating elements of a given combinatorial set sequentially, ranking – numbering elements of a given combinatorial set, unranking – generating elements of a given combinatorial set in accordance with their ranks and random selection-generating elements of a given combinatorial set in random order.

General methods for developing combinatorial generation algorithms were studied by such researchers as S. Bacchelli [1, 2], E. Barucci [2], A. Del Lungo [3, 4], V.V. Kruchinin [8, 9], P. Flajolet [10] and others. It is wellknown algorithms for the permutation generation [11–14], such as Bottom-Up, Lexicography, Johnson-Trotter [8], PIndex [15], Inversion [15].

3 MATERIALS AND METHODS

The main idea that we use in our approach to the problem of generating combinatorial objects is based on the using of the modified permanent properties.

Definition: Modified permanent of the incidence matrix will be the sum of all possible products of the numerical elements of the matrix, each of which contains one element from each row and column, and the element of the flow column (the column corresponding to the flow element) cannot be in the product together with the elements. Other rows corresponding to the same stream element (the corresponding rows will be crossed out in the schedule or with elements of other columns corresponding to the same element).

In the case of flow elements absence, the modified permanent is a normal permanent. The procedure for finding a permanent can easily be implemented recursively in the same way as finding the determinant of a matrix by decomposition on any line. Based on the definition, the decomposition procedure will be as follows: a nonzero row element is multiplied on a modified permanent matrix formed by the following rules – if a row element belongs to a stream column, the matrix

is formed by deleting the column where this element is located and all rows corresponding to all elements of this stream. If a row element does not belong to a stream column, then the matrix is formed by deleting the row and column where this element is situated, as well as all stream columns at the intersection of which are non-zero elements.

Obviously, the permanent of the square incidence matrix consisting of 1 is equal to n! (we decompose on the first line, then we have n components that already contain matrices of dimension n–1, etc.).

The main idea of our algorithm construction: in the process of permanent decomposition can be memorized the identifiers of the current elements columns.

Consider an example. Let's incidence matrix present in the form:

$$\begin{pmatrix} & 1 & 2 & 3 \\ R_1 & 1 & 1 & 1 \\ R_2 & 1 & 1 & 1 \\ R_3 & 1 & 1 & 1 \end{pmatrix} \quad (4)$$

Thus, we have:

$$\begin{aligned} \text{per mod} \begin{vmatrix} 1 & 2 & 3 \\ 1 & 1 & 1 \\ 2 & 1 & 1 \\ 3 & 1 & 1 \end{vmatrix} &= 1^1 \text{per mod} \begin{vmatrix} 2 & 3 \\ 1 & 1 \end{vmatrix} + \\ &+ 1^2 \text{per mod} \begin{vmatrix} 1 & 3 \\ 1 & 1 \end{vmatrix} + 1^3 \text{per mod} \begin{vmatrix} 1 & 2 \\ 1 & 1 \end{vmatrix} = \\ &= 1^1 1^2 \text{per mod} \begin{vmatrix} 3 \\ 1 \end{vmatrix} + 1^1 1^3 \text{per mod} \begin{vmatrix} 2 \\ 1 \end{vmatrix} + \\ &+ 1^2 1^1 \text{per mod} \begin{vmatrix} 3 \\ 1 \end{vmatrix} + 1^2 1^3 \text{per mod} \begin{vmatrix} 1 \\ 1 \end{vmatrix} + \\ &+ 1^3 1^1 \text{per mod} \begin{vmatrix} 2 \\ 1 \end{vmatrix} + 1^3 1^2 \text{per mod} \begin{vmatrix} 1 \\ 1 \end{vmatrix} = \\ &= 1^1 2^1 3^1 + 1^1 3^1 2^1 + 1^2 1^1 3^1 + 1^2 3^1 1^1 + 1^3 1^1 2^1 + 1^3 1^2 1^1. \end{aligned}$$

As we see, in the case of a square matrix all elements of which equals to 1, we can get all permutations by decomposing the permanent with “memorization”. Based on the decomposition procedure, the following general recursive PD-algorithm for forming systems of different representatives of sets is obvious:

1. The initial matrix of incidence is formed.
2. The first row of the matrix is viewed and all non-zero elements are found.
3. For each non-zero element of the first line:
 - a) the corresponding identifier element is added to the corresponding permutation;
 - b) a new incidence matrix is formed from the initial one by deleting the column and row where the found non-

zero element stands (memory is allocated and data is copied);

c) is called recursively the generation function for the new matrix.

4 EXPERIMENTS

Let's consider in more detail the problem of generating permutations. The specificity of our approach to permutation generation is that we need to keep in mind all permutations for their further use, in particular, in scheduling tasks for scheduling generation. Note that the incidence matrix has all the elements 1 and it is square. In this case, during the decomposition of the permanent, there is no need to store the incidence matrix in memory, it is enough to know only the identifier elements. So, we consider a permanent vector. We will use a singly linked list to store all elements. Each item in the list will contain information about the SDR. We can use two arrays – one to represent the already written part of the SDR, another – to place the elements that will still be used for decomposition. We can use special class in C++ notation for storing information about SDR:

```
class SDR {
public:
    char *s;
    char* p;
    SDR* next;
    int sizes;
    int sizep;
    int n;
    SDR(char* _p, int _n, int psize)
    {
        n=_n; sizep=psize; sizes=_n-psize;
        p=new char[psize];
        for(int i=0;i<psize;i++) p[i]=_p[i];
        next=NULL; }
    SDR(SDR* head, int k) {
        sizes=1+head->sizes;
        n=head->n;
        sizep=n-sizes;
        next=NULL;
        s=new char[sizes];
        p= new char[sizep];
        for(int j=0;j<sizep;j++) s[j]=head->s[j];
            s[sizes-1]=head->p[k];
        int l=0;
        for(int i=0;i<sizep+1;i++)
            if(i!=k) p[l++]=head->p[i]; } };
```

In this class we create two constructors: SDR(char* _p, int _n, int psize) for the first initialization and SDR(SDR* head, int k) for the creation new class member on the base of head in which k-th element of array p writes to the arrays. Obviously, using a singly linked list, we must correctly insert the newly created element after the head element in the list:

```
SDR *tmp=new SDR(head,i);
tmp->next=head->next;
head->next=tmp;
```

Thus we can construct recursive function for the permutation generation according to our approach:

```
void generic(SDR* head) {
    if (head->sizes<head->n) {
        for(int i=head->sizep-1;i>0;i--) {
            SDR *tmp=new SDR(head,i);
            tmp->next=head->next;
            head->next=tmp;
            generic(tmp); }
        char* s1=new char[head->sizes+1];
        char* p1= new char[head->sizep-1];
        for(int j=0;j<head->sizes;j++)
            s1[j]=head->s[j];
        s1[head->sizes]=head->p[0];
        for(int i=0;i<head->sizep-1;i++)
            p1[i]=head->p[i+1];
        delete head->s;
        delete head->p;
        head->s=0;
        head->p=0;
        head->s=s1;
        head->p=p1;
        head->sizes++;
        head->sizep--;
        generic(head); } }
```

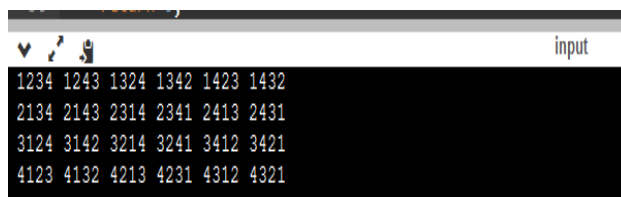


Figure 1 – Example of the program running, online GDB C++ compiler

We can consider small examples of the our program running, results is on the Fig.1, where initial array is char p[]={'1','2','3','4'} and on the Fig. 2, where initial array is char p[]={'r','i','v','n','e'}.

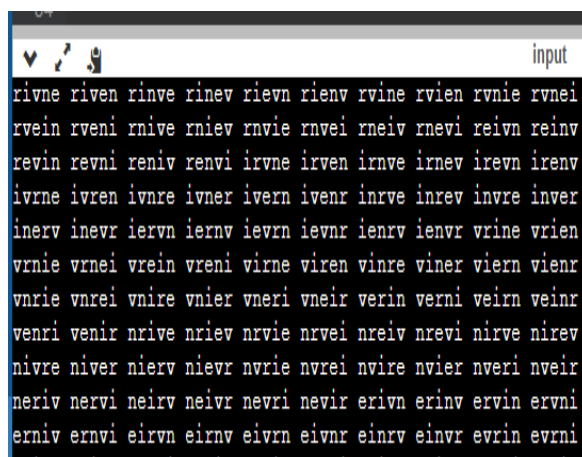


Figure 2 – Example of the program running, online GDB C++ compiler

5 RESULTS

Let us consider complexity of the PD-approach for permutation generation (see function generic()). In order

to compare the computational complexity of our approach with the known methods, we will take into account that in most cases, are used algorithms that do not require recording all variants of permutations, the so-called generation algorithms “in the same place”. When calculating the number of copies, we assume that copying to the array *s* is constructive in the case when all permutations are stored in memory, and copying to the array *p* creates an additional computational load and must be taken into account:

```
p= new char[sizep]; // ~n-1 assignments
int l=0; // 1 assignment
for(int i=0; i<sizep+1; i++) // n increments
if(i!=k) p[l++] = head->p[i]; // n-1 assignments, 1 //increment, n-1
class field access, n comparisons
The size of the array p sizep will decrease from n (initial
iteration) to 0.
```

Let $Q(n)$ be a number of assignments and increments, n -size of array *p*. Obviously, that $Q(1) = 1$. Then we have:

$$\begin{aligned} Q(n) &= n(Q(n-1) + 3n) = nQ(n-1) + 3n^2 = \\ &= n(n-1)Q(n-2) + 3n(n-1)^2 + 3n^2 = \\ &= n(n-1)(n-2)Q(n-3) + 3n(n-1)(n-2)^2 + \\ &+ 3n(n-1)^2 + 3n^2 = \dots = n(n-1)\dots(n-k)Q(n-k-1) + \\ &+ 3n(n-1)\dots(n-k+1)(n-k)^2 + \\ &+ 3n(n-1)\dots(n-k+2)(n-k+1)^2 + \dots + \\ &+ 3n(n-1)(n-2)^2 + 3n(n-1)^2 + 3n^2 = \\ &= n! + 3n(n-1)\dots 32^2 + \dots + 3n(n-1)(n-2)^2 + \\ &+ 3n(n-1)^2 + 3n^2 = n! + 3n!(2 + \frac{3}{2!} + \frac{4}{3!} + \dots + \\ &+ \frac{n-k-2+1}{(n-k-2)!} + \dots + \frac{n-1+1}{(n-1)!}) = n! + 3n!(1 + 1 + \frac{1}{2!} + \dots + \\ &+ \frac{1}{(n-k-1)!} + \dots + \frac{1}{(n-2)!}) + 3n!(1 + \frac{1}{2!} + \dots + \frac{1}{(n-1)!}) \leq \\ &\leq n! + 3n!(1 + 1 + \frac{1}{2!} + \dots + \frac{1}{n!} + \dots) + \\ &+ 3n!(1 + \frac{1}{2!} + \dots + \frac{1}{(n-1)!} + \dots) = n!(1 + 3e + 3(e-1)) = \\ &= n!(6e - 2) = O(n!). \end{aligned}$$

Consider the case of an arbitrary incidence matrix. In this case, it is necessary to prepare a new incidence matrix in the process of recursively calling the generation function, allocate memory and copy data. Thus we'll have minimum $2(n-1)^2$ additional arithmetic operations. Let all elements in the matrix be nonzero. Then we have:

$$\begin{aligned} Q(n) &= n(Q(n-1) + 2(n-1)^2) = nQ(n-1) + 2n(n-1)^2 = \\ &= n(n-1)Q(n-2) + 2n(n-1)(n-2)^2 + 2n(n-1)^2 = \\ &= n(n-1)(n-2)Q(n-3) + 2n(n-1)(n-2)(n-3)^2 + \\ &+ 2n(n-1)(n-2)^2 + 2n(n-1)^2 = \dots = \\ &= n! + 2n(n-1)(n-2)\dots 21^2 + \dots + \\ &2n(n-1)(n-2)\dots 32^2 + \dots + 2n(n-1)(n-2)^2 + \\ &+ 2n(n-1)^2 = n! + 2n!(1^2 + \frac{2^2}{2!} + \dots + \frac{(n-1)^2}{(n-1)!}) = \\ &= n! + 2n!(1^2 + \frac{2}{1!} + \dots + \frac{(n-1)}{(n-2)!}) = n! + 2n! \sum_{k=0}^{n-2} \frac{k+1}{k!} = \\ &= n! + 2n!(\sum_{k=0}^{n-3} \frac{1}{k!} + \sum_{k=0}^{n-2} \frac{1}{k!}) \leq n!(1 + 4e) = O(n!). \end{aligned}$$

6 DISCUSSION

Thus, the use of a permanent approach for generating permutations has made it possible to obtain an algorithm whose computational complexity is comparable to the fastest known algorithms. It is known [1] that, for example, Johnson Trotter's algorithm PMin(*n*) or lexicographic algorithm Plex(*n*) also have computational complexity $O(n!)$.

Obvious, that PD-algorithm generates the list of permutation in lexicographic order. This order can be defined by the initial array. If this array have standart lexicographically ordered elements, we'll get lexicographically ordered perturbations. We can consider small example of the our program running (See Fig.1), where `char p[] = {'1','2','3','4'}`. If we use initial array `char p[] = {'r','i','v','n','e'}` than we define special order: `'r' < 'i' < 'v' < 'n' < 'e'`. Our perturbations are ordered according to this order (see Fig. 2).

An essential feature of our approach is the possibility of modifying it to generate combinatorial objects of a more complex structure. To do this, it is enough to specify the appropriate incidence matrix. If it is necessary to consider some additional conditions, then it may be necessary to modify the definition of the permanent and, accordingly, the procedure for decomposition by string (for example, the impossibility of the presence in one SDK the same elements, unless they are streamed). The following modification can be considered. If the element is not “stream” then we delete the column where it stands, the row and all “stream” columns if non-zero elements intersect with this row. If the running element is streaming, then all rows where the streaming element and all columns with the same identifiers to the current one is crossed out. If there are non-zero elements of other streams at the intersection with the stream rows, the corresponding columns are also crossed out. Thus, we obtain a solution to the problem of correctness of the SDR: on the one hand in the SDR is not possible the presence of two identical elements, on the other hand such a presence is possible if the element is streaming.

If the number of non-zero elements in each line of the matrix of incidence is less than n or equal to $k < n$, the number of arithmetic operations can be significantly less. For example, for $k=1$ the number of recursive calls will not exceed n . However, when viewing a row of such a matrix, will be necessary comparison with 0 of each elements. And that's why we still have $n!$ comparisons. However, this problem is easily solved by considering the rows of the incidence matrix as dynamic arrays with numbers of nonzero elements. Then the complexity of the algorithm at $k = 1$ will be $O(n)$.

CONCLUSIONS

Thus, the paper considers a new approach to the generation of complex combinatorial objects, based on the procedure of decomposition of the modified permanent of the incidence matrix by line with memorization of index elements. The specificity of this approach is that certain additional conditions imposed on the relevant SDRs are taken into account at the stage of permanent decomposition procedures. Thus, in the case when the matrix consists of only 1, we obtain the decomposition procedure of the ordinary permanent. If we set the condition of the presence of "stream" elements in the SDR and the arbitrary configuration of the structure of the incidence matrix, the decomposition procedure must be modified.

The paper evaluates the complexity of the PD-algorithm for generating permutation and shows that it is equal to $O(n!)$. Such complexity is in the fastest algorithms, such as lexicographic, Johnson-Trotter. However, in practice PD-algorithm will obviously work slower, in particular due to the large number of memory operations and data coping. However, the recursive PD algorithm can easily be modified to generate much more complex objects, while the mentioned known approaches exclusively use the specifics of permutations.

The scientific novelty of this paper lies in the fact that in the work it was possible to use the algebraic properties of special modifications of matrices permanents to construct efficient algorithms for generating combinatorial objects.

The practical significance of obtained results is that the software realizing the proposed methods is developed, as well as experiments to study their properties are conducted. The PD-algorithm can be used in software development where the generation of combinatorial objects is used, in particular, in information security systems.

Prospects for further research are to study the proposed methods for a broad class of scheduler problems, job-shop problem (JSP), the nurse operations research problem .

REFERENCES

1. Bacchelli S., Barcucci E., Grazzini E., Pergola E. Exhaustive generation of combinatorial objects by ECO, *Acta Informatica*, 2004, Vol. 40, pp. 585–602.
2. Bacchelli S., Ferrari L., Pinzani R., Sprugnoli R. Mixed succession rules: The commutative case, *Journal of Combinatorial Theory*, 2010, Vol. 117, Series A, pp. 568–582. DOI: 10.1016/j.jcta.2009.11.005
3. Barcucci E., Del Lungo A., Pergola E., Pinzani R. ECO: A methodology for the enumeration of combinatorial objects, *Journal of Difference Equations and Applications*, 1999, Vol. 5, pp. 435–490.
4. Del Lungo A., Duchi E., Frosini A., Rinaldi S. On the generation and enumeration of some classes of convex polyominoes, *The Electronic Journal of Combinatorics*, 2011, Vol. 11, № 1, pp. 1–46. DOI: 10.37236/1813
5. Arnaw Wadhonkar, Theng Deepti A Task Scheduling Algorithm Based on Task Length and Deadline in Cloud Computing, *International Journal of Scientific & Engineering Research*, 2016, Vol. 7, № 4, pp. 1905–1909.
6. Knuth D. E. The Art of Computer Programming, Vol. 4A : Combinatorial Algorithms Part 1. Boston, Addison-Wesley Professional, 2011.
7. Ruskey F. Combinatorial Generation. Working Version (1j-CSC 425/520) [Electronic resource], *Department of Computer Science University of Victoria*, 2003. Access mode: <http://page.math.tu-berlin.de/~felsner/SemWS17-18/Ruskey-Comb-Gen.pdf>. Accessed 1 May 2020.
8. Kruchinin V. V. Methods for Developing Algorithms for Ranking and Unranking Combinatorial Objects Based on AND/OR Trees. Tomsk, V-Spektr, 2007.
9. Shablya Y., Kruchinin D., Kruchinin V. Method for Developing Combinatorial Generation Algorithms Based on AND/OR Trees and Its Application, *Mathematics*, 2020, Vol. 8, № 962. DOI: 10.3390/math8060962
10. Flajolet P., Zimmerman P., Cutsem B. A calculus for the random generation of combinatorial structures, *Theoretical Computer Science*, 1994, Vol. 132, pp. 1–35.
11. Xin Chen, Yan Lan, Attila Benkő, György Dósa, Xin Han Optimal algorithms for online scheduling with bounded rearrangement at the end, *Theoretical Computer Science*. - 2011, Vol. 412, No. 45, pp. 6269–6278. DOI: 10.1016/j.tcs.2011.07.014
12. Do P. T., Tran T. T. H, Vajnovszki V. Exhaustive generation for permutations avoiding (colored) regular sets of patterns, *Discrete Applied Mathematics*, 2019, Vol. 268, pp. 44–53.
13. Mirshekarian Sadegh, Šormaz Dušan N. Correlation of job-shop scheduling problem features with scheduling efficiency, *Expert Systems with Applications*, 2016, Vol. 62, pp. 131–147. DOI: 10.1016/j.eswa.2016.06.014
14. Humble Travis S. Yuma Nakamura, Kazuki Ikeda Application of Quantum Annealing to Nurse Scheduling Problem, *Scientific Reports*, 2019, Vol. 9, No. 1, P. 12837. DOI: 10.1038/s41598-019-49172-3
15. Fedoriaeva T. I. Combinatorial algorithms. Novosibirsk, Novosibirsk State University, 2011.

Received 08.09.2022.

Accepted 12.11.2022.

АЛГОРИТМ ДЕКОМПОЗИЦІЇ ПЕРМАНЕНТУ ДЛЯ ГЕНЕРАЦІЇ КОМБІНАТОРНИХ ОБ'ЄКТІВ

Турбал Ю. В. – д-р техн. наук, професор кафедри комп'ютерних наук та прикладної математики Національного університету водного господарства та природокористування, Рівне, Україна.

Бабич С. В. – викладач відділення Програмування, Рівненського Фахового Коледжу Національного університету біоресурсів і природокористування України, Рівне, Україна.

Кунанець Н. Е. – д-р наук із соціальних комунікацій, професор кафедри Інформаційних систем та мереж, Інституту комп'ютерних наук та інформаційних технологій, Національного університету «Львівська політехніка», Львів, Україна.

АНОТАЦІЯ

Актуальність. Розглядається задача генерування векторів, що складаються з різних представників заданої множини. Такі проблеми виникають, зокрема, в теорії складання розкладів, при плануванні зустрічей. Окремим випадком цієї задачі є задача генерування перестановок. Мета роботи – розглянути проблему з точки зору постійного та загальновідомого підходу, виходячи з концепції лексикографічного порядку.

Метод. У багатьох завданнях виникає необхідність генерувати різноманітні комбінаторні об'єкти: перестановки, комбінації з повтореннями і без них, різноманітні підмножини. У цій роботі розглядається новий підхід до генерації комбінаторних об'єктів, який базується на процедурі постійної декомпозиції. Перманент будується для спеціальної матриці інцидентності. Основна ідея цього підходу полягає в включенні до процесу алгебраїчної перманентної декомпозиції за допомогою додаткової функції рядка для запису ідентифікаторів стовпців у відповідні структури даних. У цьому випадку алгебраїчний перманент не обчислюється, а отримуємо конкретний рекурсивний алгоритм генерації комбінаторного об'єкта. Проаналізовано обчислювальну складність цього алгоритму.

Результати. В межах PD-підходу розглянуто задачі генерації комбінаторних об'єктів, зокрема, перестановок. Досліджено обчислювальну складність запропонованих алгоритмів у порівнянні з відомими підходами. Розглянуто варіант програмної реалізації розроблених алгоритмів.

Висновки. У роботі розглянуто новий підхід до генерації складних комбінаторних об'єктів, що ґрунтується на процедурі декомпозиції модифікованого перманенту матриці інцидентності за рядком із запам'ятовуванням елементів індексу. Специфіка цього підходу полягає в тому, що певні додаткові умови, що накладаються на відповідні системи різних представників, враховуються на етапі процедур декомпозиції. Досліджено складність розглянутих алгоритмів. У разі більш складних варіантів матриці інцидентності пропонується відповідна модифікація поняття перманенту і, відповідно, процедура його декомпозиції.

КЛЮЧОВІ СЛОВА: алгоритм, перманент, декомпозиція, складність обчислення.

ЛІТЕРАТУРА / LITERATURA

1. Exhaustive generation of combinatorial objects by ECO / [S. Bacchelli, E. Barucci, E. Grazzini, E. Pergola] // *Acta Informatica*. – 2004. – Vol. 40. – P. 585–602.
2. Mixed succession rules: The commutative case / [S. Bacchelli, L. Ferrari, R. Pinzani, R. Sprugnoli] // *Journal of Combinatorial Theory*. – 2010. – Vol. 117, Series A. – P. 568–582. DOI: 10.1016/j.jcta.2009.11.005
3. ECO: A methodology for the enumeration of combinatorial objects / [E. Barucci, A. Del Lungo, E. Pergola, R. Pinzani] // *Journal of Difference Equations and Applications*. – 1999. – Vol. 5. – P. 435–490.
4. On the generation and enumeration of some classes of convex polyominoes / [A. Del Lungo, E. Duchi, A. Frosini, S. Rinaldi] // *The Electronic Journal of Combinatorics*. – 2011. – Vol. 11, № 1. – P. 1–46. DOI: 10.37236/1813
5. Arnaw Wadhonkar A Task Scheduling Algorithm Based on Task Length and Deadline in Cloud Computing / Arnaw Wadhonkar, Deepti Theng // *International Journal of Scientific & Engineering Research*. – 2016. – Vol. 7, № 4. – P. 1905–1909.
6. Knuth D. E. *The Art of Computer Programming / D. E. Knuth*. – Vol. 4A : Combinatorial Algorithms Part 1. Boston : Addison-Wesley Professional, 2011.
7. Ruskey F. *Combinatorial Generation. Working Version (1j-CSC 425/520)* [Electronic resource] / F. Ruskey. – Department of Computer Science University of Victoria, 2003. Access mode: <http://page.math.tu-berlin.de/~felsner/SemWS17-18/Ruskey-Comb-Gen.pdf>. Accessed 1 May 2020.
8. Kruchinin V. V. *Methods for Developing Algorithms for Ranking and Unranking Combinatorial Objects Based on AND/OR Trees* / V. V. Kruchinin. – Tomsk: V-Spektr, 2007.
9. Shablya Y. *Method for Developing Combinatorial Generation Algorithms Based on AND/OR Trees and Its Application* / Y. Shablya, D. Kruchinin, V. Kruchinin // *Mathematics*. – 2020. – Vol. 8, № 962. DOI: 10.3390/math8060962
10. Flajolet P. *A calculus for the random generation of combinatorial structures* / P. Flajolet, P. Zimmerman, B. Cutsem // *Theoretical Computer Science*. – 1994. – Vol. 132. – P. 1–35.
11. Optimal algorithms for online scheduling with bounded rearrangement at the end / Chen Xin, Lan Yan, Benkő Attila, Dósa György, Han Xin // *Theoretical Computer Science*. – 2011. – Vol. 412, № 45. – P. 6269–6278. DOI: 10.1016/j.tcs.2011.07.014
12. Do P. T. *Exhaustive generation for permutations avoiding (colored) regular sets of patterns* / P. T. Do, T.T.H Tran, V. Vajnovszki // *Discrete Applied Mathematics*. – 2019. – Vol. 268. – P. 44–53.
13. Mirshekarian Sadegh *Correlation of job-shop scheduling problem features with scheduling efficiency* // Mirshekarian Sadegh, N. Šormaz Dušan // *Expert Systems with Applications*. – 2016. – Vol. 62. – P. 131–147. DOI: 10.1016/j.eswa.2016.06.014
14. Humble Travis S. *Application of Quantum Annealing to Nurse Scheduling Problem* / S. Humble Travis, Nakamura Yuma, Ikeda Kazuki. // *Scientific Reports*. – 2019. – Vol. 9, № 1. – P. 12837. DOI: 10.1038/s41598-019-49172-3
15. Fedoriaeva T. I. *Combinatorial algorithms* / T. I. Fedoriaeva – Novosibirsk : Novosibirsk State University, 2011.

ТЕХНОЛОГІЯ ІДЕНТИФІКАЦІЇ РЕРАЙТУ В ТЕКСТОВОМУ КОНТЕНТІ НА ОСНОВІ МЕТОДІВ МАШИННОГО НАВЧАННЯ

Холодна Н. М. – студент кафедри «Інформаційні системи та мережі», Національний університет «Львівська політехніка», Львів, Україна.

Висоцька В. А. – канд. техн. наук, доцент, доцент кафедри «Інформаційні системи та мережі», Національний університет «Львівська політехніка», Львів, Україна.

АНОТАЦІЯ

Актуальність. Перефразований текстовий контент або рерайт є однією із складних проблем виявлення академічного плагіату. Більшість систем ідентифікації плагіату призначені для виявлення спільних слів, послідовності лінгвістичних одиниць та незначних змін, але не здатні виявити суттєві семантичні та структурні зміни. Тому більшість випадків плагіату із застосуванням перефразування залишаються непоміченими.

Мета – розроблення технології виявлення перефразувань у тексті на основі моделі класифікації та методів машинного навчання через використання сіамської нейронної мережі на основі рекурентних та типу Transformer – RoBERTa для аналізу рівня подібності речень текстового контенту.

Метод. Для даного дослідження у якості ознак обрані такі метрики семантичної подібності або показники: коефіцієнт Жаккара для спільних N-грам, косинусна відстань між векторними поданнями речень, Word Mover’s Distance, відстані за словниками WordNet, передбачення двох ML-моделей: сіамської нейронної мережі на основі рекурентних та типу Transformer – RoBERTa.

Результати. Розроблено інтелектуальну систему виявлення перефразувань у тексті на основі моделі класифікації та методів машинного навчання. Розроблена система використовує принцип стекінгу моделей і інжиніринг ознак (feature engineering). Додаткові ознаки вказують на семантичну приналежність речень або нормовану кількість спільних N-грам. Додатково налаштована (fine-tuned) нейронної мережі RoBERTa (із додатковими повнозв’язними шарами) має меншу чутливість до пар речень, що не є перефразуваннями один одного. Така специфічність моделі може сприяти неправильному звинуваченню у плагіаті або некоректному об’єднанню згенерованого користувачами контенту. Додаткові ознаки збільшують як загальну точність класифікації, так і чутливість моделі до пар тих речень, що не є перефразуваннями один одного.

Висновки. Створена модель показує відмінні результати класифікації на тестових даних PAWS: зважена влучність (precision) – 93%, зважена повнота (recall) – 92%, F-міра (F1-score) – 92%, точність (accuracy) – 92%. Результати дослідження показали, що NN типу Transformer можуть бути успішно застосовані для виявлення перефразувань у парі текстів із досить високою точністю без потреби додаткового генерування ознак.

КЛЮЧОВІ СЛОВА: опрацювання природної мови, NLP, ідентифікація рерайту, виявлення перефразувань у тексті, машинне навчання з вчителем, глибинне навчання, класифікація тексту, аналіз тексту, векторне вкладення слів, WordNet, семантична подібність.

АБРЕВІАТУРА

БД – база даних;
ІС – інтелектуальна система;
ІТ – інформаційна технологія;
ПЗ – програмне забезпечення;
ML – machine learning;
NLP – natural language processing;
NN – neural network;
TRP – text pre-processing;
QA – question answering.

НОМЕНКЛАТУРА

S – система ідентифікації рерайту;
I – множина вхідних даних;
O – множина вихідних даних;
R – основні правила опрацювання потоку вхідних даних в ІС ідентифікації рерайту;
U – параметри опрацювання вхідних даних;
N – нейронна мережа;
 α – оператор скачування вхідних даних;
 β – оператор опрацювання вхідних даних;
 γ – оператор пошуку рівня подібності речень;
 μ – оператор попереднього опрацювання тексту;
 χ – NLP-оператор;

ω – оператор машинного навчання ІС на достовірних текстових даних;
 λ – оператор визначення перефразувань текстів;
 i_1 – множина даних ідентифікації;
 i_2 – сховище даних тексту/посилань на джерела;
 i_3 – множина аналогічних робіт автора/користувача;
 i_4 – конкретний запит/текст автора/користувача;
 o_1 – запити з ІС до конкретних джерел тексту;
 o_2 – колекція джерел, звідки запозичений текст;
 o_3 – множина ідентифікованих перефразувань;
 r_1 – правила алгоритму взаємодії;
 r_2 – NLP-правила;
 r_3 – правила алгоритму нейронної мережі;
 r_4 – правила алгоритму ідентифікації рерайту;
 u_1 – множина рівнів доступу;
 u_2 – множина вимог доступу;
 u_3 – множина NLP-вимог;
 u_4 – множина метрик машинного навчання;
 u_5 – множина вимог ідентифікації рерайту.

ВСТУП

Процес перефразування (рерайту) полягає у переписуванні тексту для зміни слів та послідовності

зі збереженням початкового сенсу. Ідентифікація рерайту відіграє важливу роль у різних NLP-задачах, включаючи виявлення плагіату, визначення авторства, використання у QA-системах, узагальнення тексту, машинний переклад, аналіз тексту в цілому тощо. Більш загальна задача вимірювання семантичної подібності текстів є важливим в NLP-галузі. Перефразований плагіат є однією із складних проблем, з якими стикаються системи виявлення плагіату. Більшість подібних систем ідентифікації плагіату призначені для виявлення спільних слів і незначних змін, але не здатні виявити серйозні семантичні та структурні зміни. Тому багато випадків рерайту залишаються непоміченими.

Генерація парафраз означає перетворення речення природної мови на нове речення, яке має те саме семантичне значення, але іншу синтаксичну або лексичну форму. Детекцію та генерування перефразувань застосовують у таких напрямках:

- виявлення порушення авторського права – перевірка на плагіат, визначення авторів тексту;
- поєднання дублікатів контенту, згенерованого користувачами, на інформаційних ресурсах;
- поєднання дублікатів записів однієї теми або питання, що дозволяє отримати більшу повноту (recall) для релевантного контенту при пошуку;
- машинний переклад (спрощення речень);
- QA – отримання додаткової інформації шляхом генерування варіантів запиту для отримання відповідей з БД та перефразування відповідей;
- text summarization (підвид перефразування);
- генерація природної мови (рерайт речень);
- зміна стилю письма.

Виявлення парафраз тісно пов'язане із NLP-задачею оцінки семантичної подібності тексту. У той час як ML-модель повертає ймовірність перефразування або результат бінарної класифікації пар речень, ML-модель для оцінки семантичної подібності повертає ступінь подібності за певною метрикою, напр. оцінку від 1 до 5 (завдання SentEval). При перевірці на перефразування або визначення семантичної подібності до основними проблемами є:

- відсутність спільних слів, наприклад: Is there a Quora user who have seen an UFO? Have you seen an alien?;
- усі слова спільні, наприклад: How did Portugal's team performed in match against Germany? How did Germany's team performed in match against Portugal?;
- ручна розмітка пар документів – результат анотації є суб'єктивним в залежності від ситуації;
- орфографічні і синтаксичні помилки, використання сленгу – неправильно написані слова ідентифікуються системою як нові або абсолютно відмінні від правильного значення;
- омоніми – слова мають однакове написання, однак семантичне значення є залежним від контексту.

Вищезазначені проблеми перешкоджають стрімкому розвитку детекції перефразувань як однієї з

задач з NLP-галузі. До основних методів виявлення перефразувань належать:

- модель векторного простору (vector space model), основою якої є векторизація або вкладення тексту та розрахунок відстані/подібності між двома текстами (коефіцієнт Жаккара, Евклідова відстань, косинусна відстань, відстань Word Mover's тощо);
- штучні нейронні моделі глибинного навчання (згорткові, рекурентні, сіамські, encoder-decoder, трансформери з алгоритмом уваги тощо);
- розраховані відстані та результати NN-класифікації як окремих ознак навчання фінального класифікатора.

Одним із перших підходів до вимірювання семантичної подібності між текстовим контентом є модель векторного простору (VSM) для задач області пошуку інформації [1]. Метою VSM є подання кожної сутності колекції (літер у словах, слів у реченнях, речень у контенті, контенту у корпусі) як точки в n -вимірному просторі, тобто як вектор у VSM [2]. Чим ближче розташовані точки в цьому просторі, тим більше вони є семантично подібними, і навпаки. Для заданого набору з k текстів $D=\{D_1, D_2, \dots, D_k\}$ текст D_i подають у вигляді вектора $D_i=(w_{i1}, w_{i2}, \dots, w_{in})$. У класичному VSM на основі слів кожен вимір відповідає одному терміну/слову з набору текстів. Вагу визначають на основі різних схем зважування; Bag-of-Words та TF-IDF зазвичай використовують в VSM на основі слів. Подібність між двома текстами D_i і D_j обчислюють за коефіцієнтом Жаккара, Евклідовою відстанню, косинусною відстанню тощо. Основними недоліками цієї моделі є висока розмірність, розрідженість і проблеми зі словником. Тому існують різні модифікації та узагальнення VSM.

Як і попередні методи, у глибинному навчанні документи або тексти подають у вигляді векторів за допомогою методу Doc2Vec. Окрім того, слова також подані як вектори на основі методу Word2Vec [3]. Існують варіанти навчання векторного подання слів на основі методу матричної декомпозиції, наприклад, LSA. Інший алгоритм використовує методи на основі контексту, наприклад, skip-grams, Continuous Bag of Words. Ці вектори порівнюють за допомогою косинусної відстані або іншої міри подібності.

Поява моделі Word2Vec спонукала дослідників створити інші векторні моделі, такі як Doc2Vec, FastText, GloVe, USE та ELMO. Усі ці моделі є моделями *2Vec, оскільки вони перетворюють текст (у вигляді слів, фраз, речень, розділів і цілих документів) у векторну форму, створюючи n -вимірні векторні простори. NN-навчання з текстами з великих немаркованих корпусів призводить до створення VSM з використанням довільних параметрів, найважливішими з яких є: розмірність векторного простору, мінімальна частота слів, швидкість навчання та розмір вікна/контексту спостереження кожного слова. Word2Vec складається з двох підмоделей: CBOW (безперервний мішок слів), що прогнозує пропущене слово, якщо надаємо моделі

контекст пропущеного слова, тоді як skip-gram прогнозує контекст даного слова.

Метою дослідження є розроблення ІТ для детекції перефразувань, яка дозволить спростити і водночас покращити перевірку текстів на плагіат або об'єднання даних на інформаційних ресурсах за однаковими темами або запитаннями. Проектована система має виявляти дублікати за перефразування за допомогою пошуку аналогічних текстів. Для досягнення мети були поставлені такі завдання:

- розробити та описати функціональні вимоги проєктованої системи згідно з методологією Rational Unified Process та Unified Model Language;
- проаналізувати state-of-the-art методи, що використовуються для детекції перефразувань;
- розробити та описати NN з різною архітектурою (згорткові, рекурентні, сіамські NN тощо);
- обрати найбільш оптимальну модель у контексті TPP, векторного вкладення або векторизації, вибору та генерування ознак, ML-алгоритму та відповідних параметрів;
- реалізація відповідної ІС ідентифікації рерайту та відповідної апробації отриманих результатів.

1 ПОСТАНОВКА ПРОБЛЕМИ

Систему ідентифікації рерайту S подано короткем:

$$S = \langle I, O, R, U, N, \alpha, \beta, \gamma \rangle,$$

де $I = \{i_1, i_2, i_3, i_4\}$, $O = \{o_1, o_2, o_3\}$, $R = \{r_1, r_2, r_3, r_4\}$, $U = \{u_1, u_2, u_3, u_4, u_5\}$.

Основними процесами ІС ідентифікації рерайту є «Попереднє опрацювання тексту», «NLP», «Машинне навчання» та «Визначення перефразувань».

Процес попереднього опрацювання вхідного тексту ІС ідентифікації рерайту опишемо суперпозицією:

$$C_{AU} = \mu \circ \beta \circ \alpha,$$
$$C_{AU} = \mu(\beta(\alpha(i_1, i_2, i_4), r_1, u_1), u_2).$$

NLP-процес ІС NLP опишемо суперпозицією: $C_{CU} = \chi \circ \beta \circ \alpha$, тобто

$$C_{CU} = \chi(\beta(\alpha(C_{AU}, i_2, i_3, i_4), r_1, u_3), r_2).$$

Процес машинного навчання на достовірних даних ІС ідентифікації рерайту опишемо суперпозицією:

$$C_{UL} = \omega \circ \gamma \circ \beta \circ \alpha,$$
$$C_{UL} = \omega(\gamma(\beta(\alpha(C_{CU}, i_2), i_3), u_4), r_3).$$

Процес визначення перефразувань ІС ідентифікації рерайту опишемо суперпозицією:

$$C_{US} = \lambda \circ \gamma \circ \beta \circ \alpha,$$
$$C_{US} = \lambda(\gamma(\beta(\alpha(C_{US}, i_2), i_4), u_5), r_4).$$

Задача перефразування можна розділити на дві підзадачі як виявлення і генерування перефразувань.

У задачі виявлення парафразувань результатом є ймовірність від 0 до 1, де значення, близьке до 1,

означає пару речень як перефразування один одного, 0 – різні за семантичним навантаженням значення.

Використання глибоких NN для NLP значно зросло за останні роки. Для ідентифікації перефразувань у дослідженнях використовують сіамські, згорткові, рекурентні, складні архітектури моделей на основі поєднання згорткових та рекурентних NN, трансформерів тощо. Точність моделі залежить від таких факторів, як кількість класів, на які розподіляються дані, вибору методів традиційного або глибокого навчання та їх комбінації з методами векторизації (вкладення) слів і TPP. При побудові ІС визначення перефразування у тексті для нового набору даних необхідно оцінити не лише різні методи і архітектури ML-моделей, а й TPP-алгоритми і подання тексту у числовому форматі та можливі комбінації їх поєднання для отримання найкращої можливої якості (точності) функціонування типової ІС ідентифікації рерайту.

Об'єктом дослідження є процеси детекції рерайту у тексті із застосуванням оптимального конвеєру (pipeline), що включає TPP, векторизацію або вкладення слів, вибір та генерування ознак, бінарну класифікацію за допомогою певного ML-алгоритму і подальше опрацювання отриманих результатів для виявлення перефразувань на інформаційних джерелах. Наукова новизна полягає у застосуванні NN з новою архітектурою, огляд state-of-the-art методів та порівняння різних етапів конвеєру (pipeline) дасть змогу визначити таку їх комбінацію, яка дозволить отримати якісну модель визначення перефразувань в тексті для обраних контрольних наборів даних. Проектована ІС дозволить покращити процес визначення рерайту у тексті. Розроблювана ІС може використовуватися модераторами інформаційних ресурсів та соціальних мереж для оцінки підтримки якості публікацій та об'єднання даних за темами або запитаннями. Окрім того, детекція перефразувань може бути модулем до існуючих систем перевірки текстів на плагіат.

2 АНАЛІЗ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

Більшість ІС перевірки на плагіат порівнюють частини речення або спільні слова, однак детекція перефразувань досі залишається актуальним завданням у галузі опрацювання природньої мови і не є реалізованою у більшості онлайн-платформ і додатків. Окрім того, наявні системи перевірки на плагіат не можуть із достатньою точністю виявити перефразування та вказати оригінальне джерело [4].

Отже, детекція перефразувань є актуальною проблемою і наразі у більшості академічних робіт дослідники використовують методи глибокого навчання і архітектури мовленнєвих моделей, що мають мільйони параметрів. Однак історія вирішення цього завдання бере початок із більш простих методів розрахунку семантичної відмінності як відстані між двома термінами у базі даних WordNet [5]. Це БД семантичних зв'язків між словами (синсетами).

Зв'язки містять синоніми, гіпоніми (підтип), мероніми (частина) тощо. Ієрархічна структура БД дає змогу розрахувати семантичну спорідненість окремих слів за різними формулами. В [6] запропоновано підхід, заснований на значеннях семантичної спорідненості слів із словника WordNet. У основі методу – ідея порівняння пари слів відповідно до частини мови.

Дослідники використали шість різних відстаней семантичної подібності основи WordNet [7–12]. Максимальну подібність шукають лише всередині класів слів з однаковою частиною мови. Для впровадження двонапрямленості використовують середнє арифметичне двох значень запропонованої функції, значення якої залежить від значення максимальної подібності пари слів та їх специфічності. Аналогічно можна поєднати усі шість відстаней, усереднивши показники фінальної оцінки семантичної подібності. Семантична подібність слів згідно з [7] залежить від довжини найкоротшого шляху між синсетами як вузлами графа. За [8] подібність слів визначається як перетин множин слів визначень, що відповідають заданим термінам. Функція відстані в [9] залежить від глибини двох концептів у таксономії та глибини найближчого спільного предка, за [10] – від ймовірності зустріти спільного предка у великому корпусі, за [11–12] – від ймовірності зустріти як спільного предка, так і два синсети, що порівнюються.

Автори в [13] запропонували новий алгоритм вимірювання семантичної спорідненості коротких речень із використанням методів, заснованих на корпусі (pointwise mutual information та latent semantic analysis) або на вищезазначених відстанях згідно даних з WordNet. Формула семантичної спорідненості речень поєднує метрики подібності пари слів та їх специфічності. Дана формула є потенційно гарним індикатором подібності двох введених текстів, оскільки отримані значення точності, влучності, повноти і F-міри є кращими у порівнянні із базовим підходом вимірювання косинусної відстані між двома вхідними реченнями, переведеними до векторного формату на основі TF-IDF.

У роботі [14] запропонований метод визначення семантичної спорідненості, який генерує семантичний профіль для слів на основі основних концептуальних ознак, зібраних з енциклопедичних знань. Основна ідея моделі – значення слова визначають поняттями, які знаходяться в прямому контексті. Даний метод складається з двох основних кроків. Спочатку на основі Wikipedia створюють анотований корпус основних концептуальних слів з відповідними посиланнями. Далі корпус використовують для вимірювання семантичної подібності слів та текстів.

В [15] адаптували формулу косинусної відстані між векторами для розрахунку семантичної подібності текстів. Запропонований алгоритм використовує інформацію про подібність слів, отриману із словників WordNet. Для розрахунку подібності між парами речень використовується

косинусна відстань між векторами, що подають речення, та матриці семантичної подібності, що містять інформацію про подібність пар слів, отриману з шести семантичних відстаней WordNet, заснованих на ієрархії. Кожне речення подано у вигляді двійкового вектора (з елементами, рівними 1, якщо слово присутнє у реченні, і 0 в іншому випадку).

У [16] проведено порівняльне дослідження між векторними вкладеннями слів, отриманими за допомогою NN, і традиційними векторними поданнями слів, заснованими на підрахунку спільних появ. Автори виконали два завдання невеликого масштабу (розбір значень слів і подібність речень), і два великомасштабних (виявлення парафразувальних і тегування актів діалогу). Для задачі розпізнавання перефразувальних дослідники використали косинусну відстань між векторами. Використання векторних вкладень слів, отриманих із використанням NN, значно покращує якість моделі для виявлення перефразувальних та тегування актів діалогу, однак не дає значного приросту точності у розв'язанні завдань невеликого масштабу.

В [17] автори запропонували метод вимірювання семантичної подібності текстів з використанням вимірювання семантичної подібності слів на основі корпусу (pointwise mutual information) та нормалізованої і модифікованої версії алгоритму відповідності рядків найдовшої спільної послідовності (LCS). Дослідження зосереджено на вимірюванні подібності між двома реченнями або між двома короткими абзацами. Метод оцінений на основі даних Microsoft Paraphrase Corpus (MSRP).

В [18] удосконалили процес перевірки тексту на плагіат за допомогою використання різних NLP-алгоритмів: кількість однакових 3-грам, Language Model Probability, Longest Common Subsequence, Dependency Relations Matching. Отримані ознаки використані для навчання наївного Байєсівського класифікатора за двома категоріями в залежності від наявності плагіату та за чотирма категоріями в залежності від рівня та типу перефразування. Результати експериментів є задовільними для класифікації документів у корпусі на дві категорії (з рерайтом та унікальні): 37 із 38 документів без плагіату правильно класифіковані, і лише декілька документів із плагіатом класифіковані як без плагіату (5 з 57). Розмежування між трьома різними рівнями плагіату виявилось набагато складнішим завданням. Незадовільна точність моделі класифікації на типи плагіату залежить як від складності завдання, так і від можливої помилки анотаторів.

В [19–20] розробили дві системи для визначення семантичної подібності пар коротких речень. Усі міри подібності слів засновані на WordNet. Щоб обчислити оцінку подібності для пари слів, автори взяли максимальний бал подібності для всіх можливих пар понять – синсетів WordNet та використали бібліотеку NLTK для обчислення мір подібності Leacock and Chodorow і Lin. Для подібності слів на основі корпусу

автори використали дистрибутивну лексичну семантичну модель. Вони використали латентний семантичний аналіз (LSA) на великому корпусі для оцінки розподілів. Система навчена прогнозувати оцінки подібності речень, використовуючи регресійну модель опорного вектора з вимірюванням кількох характеристик ступеня накладання спільних слів, подібності синтаксису і семантичної близькості слів.

У [21] подано метод для ідентифікації наявності плагиату із застосуванням перефразувань. Цей метод використовує класичний ML-алгоритм – логістичну регресію. В даній системі ознаками виступають певні особливості лексики, синтаксису, семантики та структури, які отримуються з «підозрілого» документу та оригіналу. До ознак особливостей лексики належать: Dice Coefficient (кількість спільних символів), Jaro Distance, Jaccard Coefficient (відношення спільних термінів до загальної кількості термінів), Levenshtein Distance, адаптована Manhattan Distance, Ngram Distance – аналог Manhattan Distance для N-грамів, Soundex Distance. До синтаксичних ознак належать POS N-gram Distance та Noun Ratio, семантичних – Semantic Similarity Distance, структурних – Stopword N-gram Distance, Word Pair Order, String Length Ratio. Результати експерименту на корпусі даних PAN@CLEF2013 показують, що використання різних типів ознак дає змогу отримати більш точні результати.

В [22] запропонували нову архітектуру глибокого навчання Vi-CNN-MI для виявлення/ідентифікації рерайту шляхом порівняння речень на кількох рівнях деталізації (уніграми, N-грами та речення) на основі згорткової NN та моделюючи особливості взаємодії на кожному рівні. Отримані характеристики є вхідними ознаками для бінарного класифікатора на основі логістичної регресії.

У [23] подана система для вирішення задачі детекції перефразувань шляхом виявлення відмінностей між реченнями та оцінки того, наскільки речення є різними. Такий метод також дозволяє виявити такі перефразування, що містять незначну кількість додаткової інформації. У якості алгоритму класифікації використано метод опорних векторів.

В [24] порівняли класичні ML-алгоритми (метод опорних векторів, наївний Байєсів класифікатор, maximum entropy classifier). Ознаками є дані про лексичну і семантичну спорідненість речень. Група ознак «word overlap» містить відношення кількості однакових N-грамів до загальної кількості слів у реченнях. До лексичної групи ознак також належать skip-grams та longest common subsequence. До семантичних ознак належить noun/verb semantic similarity measure – кількість спільних іменників або дієслів, proper name – кількість спільних власних назв. Семантична ознака cardinal number дає змогу фіксувати числа з порівнянням «більше ніж» та «менше ніж», а також числа, записані словами. За результатами дослідження, найкращі показники досягнуті за допомогою методу опорних векторів.

© Холодна Н. М., Висоцька В. А., 2022
DOI 10.15588/1607-3274-2022-4-11

У [25] показали, що метрики якості машинного перекладу (BLEU, NIST, WER, PER) можуть бути застосовані як ознаки у системі визначення і детекції перефразувань для навчання класифікатора і отримання якісних результатів. Окрім того, дослідники розробили класифікатор, заснований на Position independent word error rate (PER) та інформації про розподіл частин мови у реченнях.

В [26] повторно дослідили гіпотезу про те, що метрики, розроблені для автоматизованої оцінки якості машинного перекладу, можуть бути використані як ознаки класифікатора для детекції перефразувань. Результати показали, що мета-класифікатор, ознаками якого є лише метрики якості машинного перекладу, значно перевершує показники точності, отримані у попередніх дослідженнях. Усього використано три алгоритми класичного машинного навчання для класифікації: логістична регресія, SMO імплементацію методу опорних векторів і варіацію алгоритму найближчих сусідів. До метрик якості машинного перекладу належать: BLEU, NIST, TER, TERp, METEOR, SEPIA. Окреме використання метрики TERp забезпечує непогані результати і перевершує багато інших методів класифікації на основі численних складних ознак.

В [27] розробили систему для детекції перефразувань у коротких текстах на основі методів глибинного навчання. Архітектура класифікатора заснована на шарах згортки та рекурентних (long-short term memory) NN, що опрацьовують отриманий результат. Результат опрацювання рекурентними NN є семантичним поданням речення, тому ознаками фінального класифікатора є різниця між двома векторами виходу з рекурентних NN. Згорткова NN перетворює парну матрицю подібності усіх слів на вектор семантичної подібності речень, що використовується як ознаки для класифікатора. Додатковими ознаками є косинусна відстань між векторами двох речень, середнє значення відстані між іменниками, дієсловами, прикметниками, обчисленої на основі WordNet, відношення кількості спільних уні-, 2-, 3-грам до загальної кількості N-грамів тощо.

У роботі [28] поданий підхід до ідентифікації перефразувань на основі рекурсивних автокодерів, що досліджують вектори ознак для фраз за допомогою синтаксичних дерев. Ці ознаки використані для вимірювання подібності слів і фраз між двома реченнями. Оскільки речення мають довільну довжину, результуюча матриця подібності має змінний розмір. Додано новий рівень динамічного пулінгу, який обчислює подання фіксованого розміру з матриць змінного розміру. Дане подання використовувалося як вхідні дані для класифікатора.

У дослідженні [29] на протипагу створенню та відбору великої кількості ознак використано рекурентну NN із шарами довгої-короткочасної пам'яті. NN приймає на вхід речення змінної довжини і навчається завданню регресії – передбаченню ступеня семантичної спорідненості пари речень. NN

має сіамську архітектуру і навчена з використанням функції втрат Mean Squared Error. Для подання слів у вигляді векторів використані попередньо навчені векторні вкладення word2vec, отримані на великих нерозмічених корпусах даних.

Автори в [30] поєднали сіамську рекурентну NN із двома напрямленими рекурентними мережами із шарами довгої-короткочасної пам'яті на рівні символів. Така модель є нечутливою до помилок орфографії, заміни синонімів і зайвих слів.

В [31] протестували декілька різних типів сіамських NN для задачі детекції перефразувань: LSTM, Bi-directional LSTM, GRU, Bi-directional GRU, LSTM + Attention, GRU + Attention, GRU + Capsule + Flatten. Найкращі результати отримані для NN з клітинами типу gated recurrent unit.

В [32] використали сіамську NN із довгою-короткочасною пам'яттю для класифікації пари текстів арабською мовою, відповідно до того, чи є вони перефразуванням один одного. Для векторного подання слів дослідники застосували метод векторного вкладення сімейства word2vec, що має назву Glove. Для розрахунку семантичної подібності текстів використана косинусна відстань між двома векторними поданнями речень.

В [33] використали сіамську NN для детекції парафраз у текстах мови тѣлугу. Більшість даних були зібрані вручну з різних газет. Окрім того, дослідники порівняли три методи векторного вкладення (Word2Vec, Glove, Fasttext) та їх комбінацію. Найкраща точність на тестовому наборі даних досягнута для комбінації цих трьох методів.

У [34] поданий підхід глибокого навчання з підкріпленням до генерації парафразів. Описана нова структура розв'язку даної задачі, яка складається з генератора та оцінювача. Генератор, побудований як ML-модель на основі рекурентних NN архітектури sequence-to-sequence, перефразовує вхідне речення. Оцінювач класифікує, чи є два речення перефразуваннями одне одного. Генератор спочатку тренується за допомогою глибокого навчання, а потім додатково налаштовується за допомогою навчання з підкріпленням, під час якого оцінювач дає винагороду. Для навчання оцінювача дослідники пропонують два методи, засновані на навчанні із вчителем та навчанні з оберненим підкріпленням відповідно, залежно від типу навчальних даних.

В [35] розроблено систему SimAll, що поєднує методи, засновані на порівнянні стрічок, корпуси та знання, і підтримує англійську та арабську мови. Усього система підтримує 61 алгоритм оцінки семантичної подібності речень, результати оцінки кожного алгоритму після нормування належать інтервалу [0, 1]. Користувач має обрати метод агрегування результатів: середнє, сума або максимальне значення. До метрик, заснованих на знаннях, належить 6 відстаней семантичної схожості: Path (path), Leacock & Chodorow (lch), Wu & Palmer,

Resnik (res), Lin (lin), Jiang & Conrath (jcn). Метрики доступні лише для англійської мови.

В [36] запропоновано два варіанти деревоподібної моделі LSTM для опрацювання дерев залежностей або синтаксичних дерев. Створені моделі протестовані на задачі визначення ступеня семантичної подібності двох речень.

У [37] розроблено системи, що поєднують згорткові та рекурентні NN для вимірювання семантичної подібності речень. Дослідники використали мережу згортки для врахування локального контексту слів і LSTM для врахування глобального контексту речень. Така поєднання мереж допомагає зберегти відповідну інформацію про речення та покращує обчислення ступеня подібності речень.

В [38] презентували великомасштабний корпус паралельних даних, утворений за допомогою моделей глибокого навчання BERT, RoBERTa, Longformer архітектури Transformer. Набір даних включає параграфи з наукових праць у arXiv, тези, а також статті Вікіпедії та їх перефразовані аналоги (всього 1,5 млн абзаців). Архітектура глибокого навчання Transformer дає змогу досить точно класифікувати оригінальний та перефразований тексти із використанням статичних векторних вкладень (fastText) [39]. Модель RoBERTa досягла найкращих результатів у задачі виявлення перефразувань.

В [40] запропонували і застосували новий підхід до створення паралельних корпусів перефразованих текстів за допомогою генеративних NN архітектури Transformer. Відповідно до результатів досліджень, хоча згенеровані NN набори даних дають змогу покращити точність систем виявлення перефразувань, менший, але створений людиною набір даних ще більше покращує точність класифікації.

В [41] застосували додаткове налаштування (fine-tuning) моделі глибокого навчання BERT архітектури Transformer. Модель BERT має 110 мільйонів параметрів у базовій версії, 340 мільйонів – у «великій» [42]. Попереднє налаштування BERT складається з двох завдань. Першим завданням є моделювання мови, за якого модель має передбачити випадково обраний і замаскований токен. Іншим завданням є передбачення наступного речення, де модель має визначити, чи є два речення послідовними. Додаткове налаштування моделі для основного завдання відбувається за допомогою відповідного набору даних.

В [43] презентували метрику відстані між документами Word Mover's Distance на основі векторних вкладень. Word Mover's Distance вимірює відмінність між як мінімальну відстань, яку «мають пройти» векторні вкладення слів першого речення до векторних вкладень слів другого речення.

3 МАТЕРІАЛИ ТА МЕТОДИ

ІС виявлення перефразувань призначена для модераторів інформаційних ресурсів та соціальних мереж для оцінки та підтримки якості публікацій і

об'єднання даних за темами або запитаннями. Окрім того, алгоритм детекції перефразувань може бути доданий до існуючих систем перевірки текстів на плагіат. Наведемо приклад опису вимог у контексті ІС перевірки унікальності академічних робіт.

Функціонування ІС забезпечується науковим керівником, автором тексту, особою, що відповідальна за перевірку тексту та програмним забезпеченням. Науковий керівник (або викладач) здійснює контроль на всіх етапах написання автором своєї роботи і проводить ТРР до автоматизованої перевірки на оригінальність. Автор активно взаємодіє з керівником і реагує на його критику, виправляє роботу відповідно до його зауважень і потім передає свою роботу на перевірку. Відповідальна особа завантажує текст у програму і отримує результат, повідомляє автора та наукового керівника, формує звіт. Зацікавлені особи прецеденту та їх вимоги:

– ПЗ для перевірки текстів має відповідати визначеним системним вимогам, повинне забезпечити максимальну точність при перевірці результатів та формуванні звітів; має автентифікувати користувача, перевірити роботу і повернути результат, зберегти результат у обліковому записі користувача;

– автор хоче отримати підтвердження оригінальності роботи для її публікації або захисту; повинен написати текст, оформити роботу згідно вимог і надати її для подальшого опрацювання;

– відповідальна за перевірку особа повинна отримати результат автоматизованої перевірки, впевнитись в тому, що програма правильно визначила скопійовані фрагменти та першоджерела перефразованих речень.

Користувач ІС: відповідальна за перевірку особа, що буде використовувати ІС, для перевірки текстів на унікальність. Передумови прецеденту:

– наукова робота має бути належним чином оформлена і подана в одному з форматів, які підтримує система;

– комп'ютер, за допомогою якого здійснюватиметься перевірка, підключений до Інтернету та має встановлене необхідне ПЗ;

– відповідальна за перевірку особа має бути успішно авторизована.

Основний успішний сценарій:

– відповідальна за перевірку особа завантажує текст у програму і отримує результат;

– результат експертизи підлягає додатковому аналізу щодо кількості виявлених співпадінь та адекватності посилань на першоджерела.

Альтернативні потоки:

– помилка у роботі програми:

1. Відповідальний за перевірку звертається до служби технічної підтримки (розробника) і повідомляє про помилку у програмі.

2. Розробник усуває помилку і надає нову версію або надає інформацію про способи її самостійного усунення (точка повернення).

Пост-умови:

© Холодна Н. М., Висоцька В. А., 2022
DOI 10.15588/1607-3274-2022-4-11

– дані про результат перевірки занесені до БД.

Спеціальні вимоги: ІС має точно встановлювати відсоток унікальності тексту, підсвічувати частини тексту відповідно до їх типів (скопійований текст, оригінальний текст, перефразований текст, цитата або неправильне вказування першоджерела) та надати список сайтів (робіт) з відсотком збігу, повноцінно підтримувати українську та англійську мови, легко інсталюватися, бути загальнодоступною, підтримувати різні формати (.txt, .doc, .docx тощо).

На діаграмі використання (рис. 1) основними акторами є користувачі та додатки, що по-різному реалізують функції системи. На цій діаграмі не зображені деталізовані варіанти використання, як-от реєстрація користувача, завантаження файлу або взаємодія адміністратора із системою, що у свою чергу містить налаштування системи, навчання NN, обрання методів виявлення перефразувань (рис. 2–3).

Індивідуальний користувач платформи може використовувати систему як додаток для перевірки на плагіат та унікальність вмісту. Така система може використовуватись для перевірки академічної чесності студентів та наукових співробітників. Окрім того, система перевірки на плагіат може використовуватись для встановлення автора тексту. Завдяки додатковій перевірці на наявність перефразувань така система матиме більшу повноту (recall). У такому випадку збільшиться частка правильно вказаних джерел від усіх істинних.

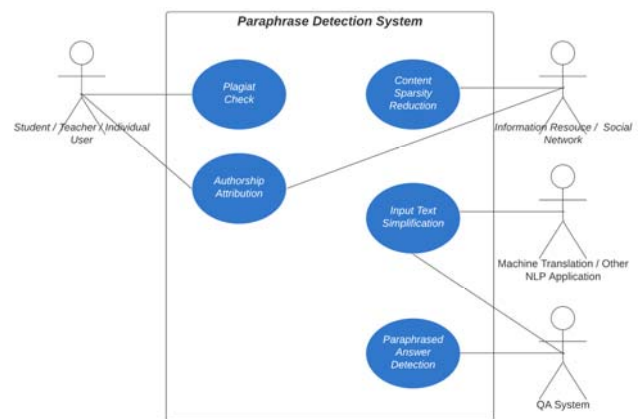


Рисунок 1 – Діаграма use case ІС виявлення рерайту

Інформаційний ресурс (у тому числі і соціальні мережі) можуть використовувати систему для зменшення розрідженості контенту шляхом об'єднання публікацій або запитань, що відносяться до однієї спільної теми. Також ці ресурси можуть використовувати систему виявлення перефразувань для встановлення автора тексту. Як додаток машинного перекладу, так і QA-системи можуть застосовувати функціонал запропонованої ІС для спрощення вхідного повідомлення. Окрім того, QA-системи також можуть застосовувати детекцію перефразувань для більш розширеного пошуку відповідей на запитання.

Для побудови діаграми класів визначено сім основних класів: Individual User (користувач системи), Document (документ, унікальність якого має бути перевірена), Paraphrase Detection Software (додаток для перевірки на плагіат), User Database (БД користувачів), Documents Database (БД документів), Data Processing Server (сервер опрацювання даних), ML Model (ML-модель). Діаграма класів на рис. 5 позначає застосування системи лише індивідуальним користувачем для перевірки унікальності власних текстів. На діаграмі класів застосування системи виявлення перефразувань, що використовується сторонніми додатками, мають бути визначені

додаткові методи обміну повідомленнями із головною програмою за допомогою запитів. Користувач може завантажити та змінити документ, здійснити його перевірку та отримати результати.

Клас додатку перевірки на плагіат позначає програму, що реалізовує користувацький інтерфейс та базові методи перевірки на плагіат шляхом порівняння стрічок. Така програма може бути встановлена локально на комп'ютері або реалізована як онлайн-платформа. Додаток відправляє запити до обчислювального серверу та бази даних користувачів.

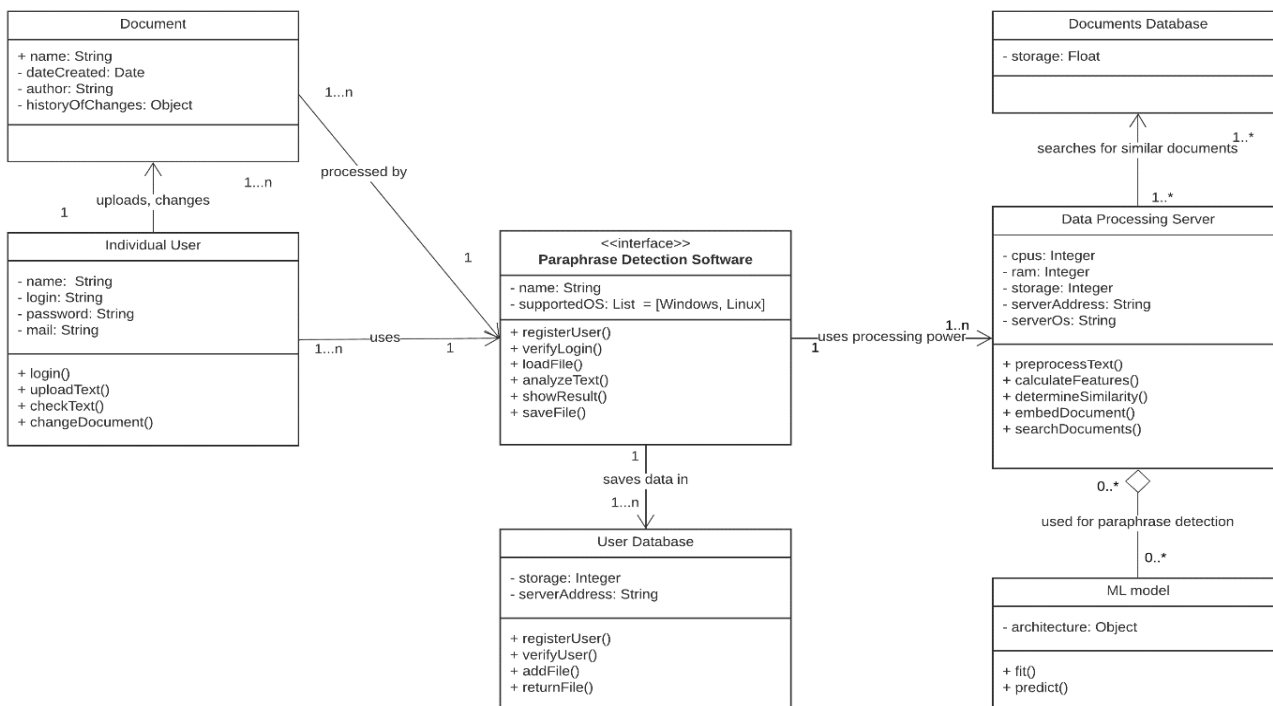


Рисунок 2 – Діаграма класів ІС виявлення перефразувань у тексті

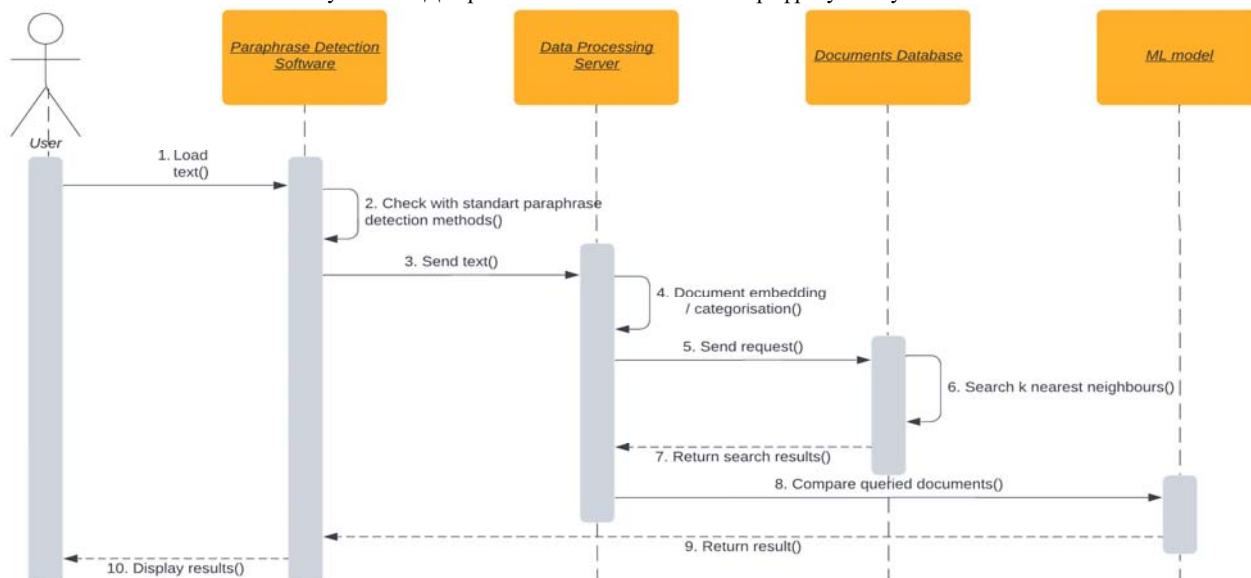


Рисунок 3 – Діаграма послідовності системи виявлення перефразувань

Сервер опрацювання даних містить методи для порівняння документів за допомогою ML-алгоритмів та застосовує векторне вкладення або рубрикацію документів для пошуку найближчих сусідів, оскільки попарне порівняння тексту із усіма документами, наявними в БД, є надто ресурсозатратним процесом. Сервер опрацювання даних також має розраховувати додаткові ознаки, що будуть використовуватись як вхідні дані класифікатора. Такими додатковими ознаками можуть бути спільна кількість іменованих сутностей або частин мови (рис. 3). Розрахунок таких ознак вимагатиме додаткової реалізації спеціальних парсерів. Діаграма послідовності показує часові аспекти взаємодії компонентів системи. Оминаючи налаштування системи, реєстрацію і вхід користувача, після завантаження файлу наступним кроком є перевірка текстів на унікальність за допомогою базових алгоритмів, що засновані на порівнянні стрічок та їх фрагментів. Для більш детальної перевірки на наявність перефразувань документ спочатку має бути рубрикований або переведений у векторне подання. Такий підхід застосовується з метою збереження обчислювальних ресурсів та зменшення часу опрацювання запиту.

До важливих внутрішніх налаштувань системи належить кількість документів, які будуть порівняні із користувацьким. Завелика кількість подібних документів сповільнить перевірку, замала – зменшить її точність і повноту. Результати визначення ступеня семантичної подібності або бінарної класифікації повертаються і мають бути візуалізовані разом з результатами попередньої простої перевірки.

Важливою частиною розробки системи автоматичного виявлення перефразувань є створення та навчання ML-моделі для задачі класифікації текстів. Для отримання максимальної якості перевірки необхідно дослідити різні методи розрахунку відстаней, ML-алгоритми, TRP-методи. Діаграма діяльності розподілена на дві частини (рис. 4–5) і позначає усі можливі розгалуження під час TRP та побудови системи відповідно. Отже, перед розробниками системи постає система розгалужень з усіх можливих методів та їх комбінацій. У випадку бінарної класифікації для початку потрібно впевнитись, що обсяг корпусу є достатньо великим і збалансованим, оскільки чисельна перевага записів певного класу може призвести до упередженого рішення штучної NN.

Якщо даних не є достатньо, застосовують методи генерації штучних даних або зміни наявних записів. Після цього потрібно видалити "шум", що може бути зайвими символами, що не містять смислове навантаження. Короткі тексти, особливо публікації у соціальних мережах із великою ймовірністю будуть містити емої. Емої можуть вказувати на тональне або емоційне забарвлення тексту, тому розробники

системи можуть замінити їх відповідними словами або видалити. Наступним кроком є розбиття речень на токени. Стоп-слова не містять семантичного забарвлення, тому їх видалення є опціональним. До наступних кроків TRP належать лематизація (початкова форма слова), стемінг (основа слова), виправлення орфографічних помилок та видалення цифр. Як і у випадку із видаленням стоп-слів, застосування вищезгаданих кроків не є обов'язковим, однак може по-різному впливати на точність класифікації. Після TRP набір даних розділяється на навчальну та тренувальну вибірки.

На рис. 5 зображені усі основні методи та їх поєднання для бінарної класифікації пари речень в залежності від того, чи є вони перефразуванням один одного. Розробники системи можуть застосувати одну метрику для фінального рішення або агрегувати декілька значень за допомогою ML-методів, наприклад, логістичної регресії або випадкового лісу. Для спрощення вигляду діаграми, на ній частково відсутні сторожові умови переходу між діяльностями. «Нотатками» позначені додаткові можливі розгалуження вибору методу розрахунку відстані за WordNet, методу агрегування відстаней, векторного вкладення слів, класифікації. На діаграмі розгортання системи автоматичного виявлення перефразувань (рис. 6) зображені процесори Сервер опрацювання даних і дві бази даних. Користувач має доступ до функціоналу системи за допомогою графічного інтерфейсу завантаженої програми або онлайн-платформи. У БД зберігається інформація про користувачів та документи. Сервер опрацювання даних містить модулі ML-моделі, базової перевірки на плагіат і векторного вкладення або рубрикації документу. Альтернативним варіантом схеми розгортання є використання власних обчислювальних ресурсів персонального комп'ютера для аналізу даних, однак, з врахуванням того, що для отримання достовірного результату необхідно застосувати кілька етапів перевірки, особливо з використанням ML-методів і порівняння документів, що зберігаються у базі даних, така архітектура не є доцільною.

Для ML найпопулярнішими мовами є Python, R, Java, C++. Перевагою Python з-поміж інших мов саме для створення системи розпізнавання перефразувань є його підтримка великої кількості бібліотек:

- для роботи з ML-методами: Scikit-Learn;
- для створення штучних NN, глибинного навчання: TensorFlow, Keras, PyTorch;
- для опрацювання природньої мови: NLTK, spaCy, WordNet;
- для роботи із масивами, матрицями: NumPy;
- роботи з таблицями: pandas;
- візуалізації даних (в тому числі, інтерактивної): Matplotlib, seaborn, Plotly.

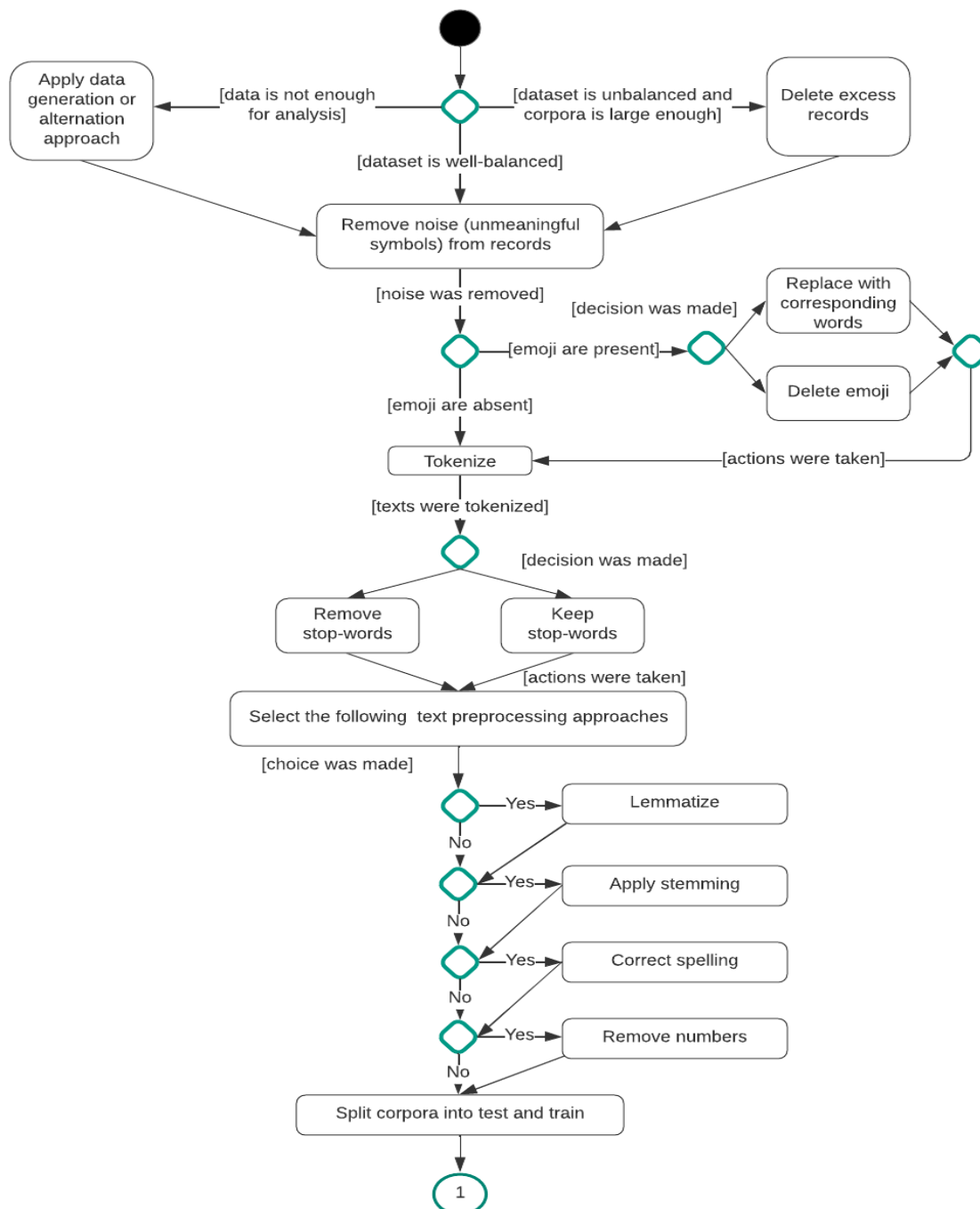


Рисунок 4 – Перша частина діаграми активності створення системи виявлення перефразувань

Бібліотека Scikit-Learn підтримує ТРР, зменшення розмірності даних, вибір ML-моделей для регресії, класифікації або кластерного аналізу. Однак Scikit-Learn не має комплексної підтримки для створення моделей глибокого навчання. Для створення штучних NN обрана бібліотека Keras, що слугує високорівневим API для TensorFlow 2. Keras дозволяє будувати послідовні моделі у вигляді графу, вершинами якого є шари (layers) певного типу із заданою кількістю вузлів. Також, за допомогою Keras можна поєднувати результати роботи кількох окремих частин NN для їх подальшого опрацювання, така структура не є лінійною. TensorFlow дає можливість

імпортувати навчену ML-модель для її подальшого використання у інших програмах. TensorFlow також підтримує виконання низькорівневих операцій із тензорами за допомогою центральних процесорів, графічних процесорів та тензорних блоків опрацювання. Бібліотека NLTK у цьому дослідженні застосовується для ТРР: токенизації, видалення стоп-слів, стемінгу, лематизації. Також за допомогою функцій з цієї бібліотеки можна виявляти найпопулярніші N-грами і частини мови окремих токенів, розпізнавати іменовані сутності тощо. До додаткових бібліотек, що спрощують роботу із природньою мовою, належать Regex та емої – для

використання регулярних виразів і заміни емоджі словами відповідно. Для роботи над задачами з дослідження даних і застосування ML зручним інструментом є Jupyter Notebook, який дозволяє запускати написаний код невеликими фрагментами – комірками. Одним із онлайн-сервісів, що надає змогу використовувати Jupyter Notebook без локального встановлення, є Google Colab. Цей сервіс дає можливість використовувати графічні процесори GPU і TPU, що значно пришвидшують навчання NN.

Для навчання і тестування ML-моделі, а також конструювання ознак обраний набір даних Paraphrase Adversaries from Word Scrambling [44], частина

PAWS-Wiki Labeled (Final). PAWS-Wiki містить 65 401 пару речень, 44,2% з яких є перефразуваннями один одного. Оскільки у різних дослідженнях для визначення семантичної подібності та виявлення перефразувань використовуються відмінні одна від одної методології (від розрахунку кількості спільних N-грам до застосування глибинного ML-методів) [45], для отримання максимально великої точності класифікації необхідно порівняти та (або) поєднати різні алгоритми виміру семантичної подібності речень. Для даного дослідження у якості ознак обрані такі метрики семантичної подібності або показники: коефіцієнт Жаккара для спільних N-грам, косинусна

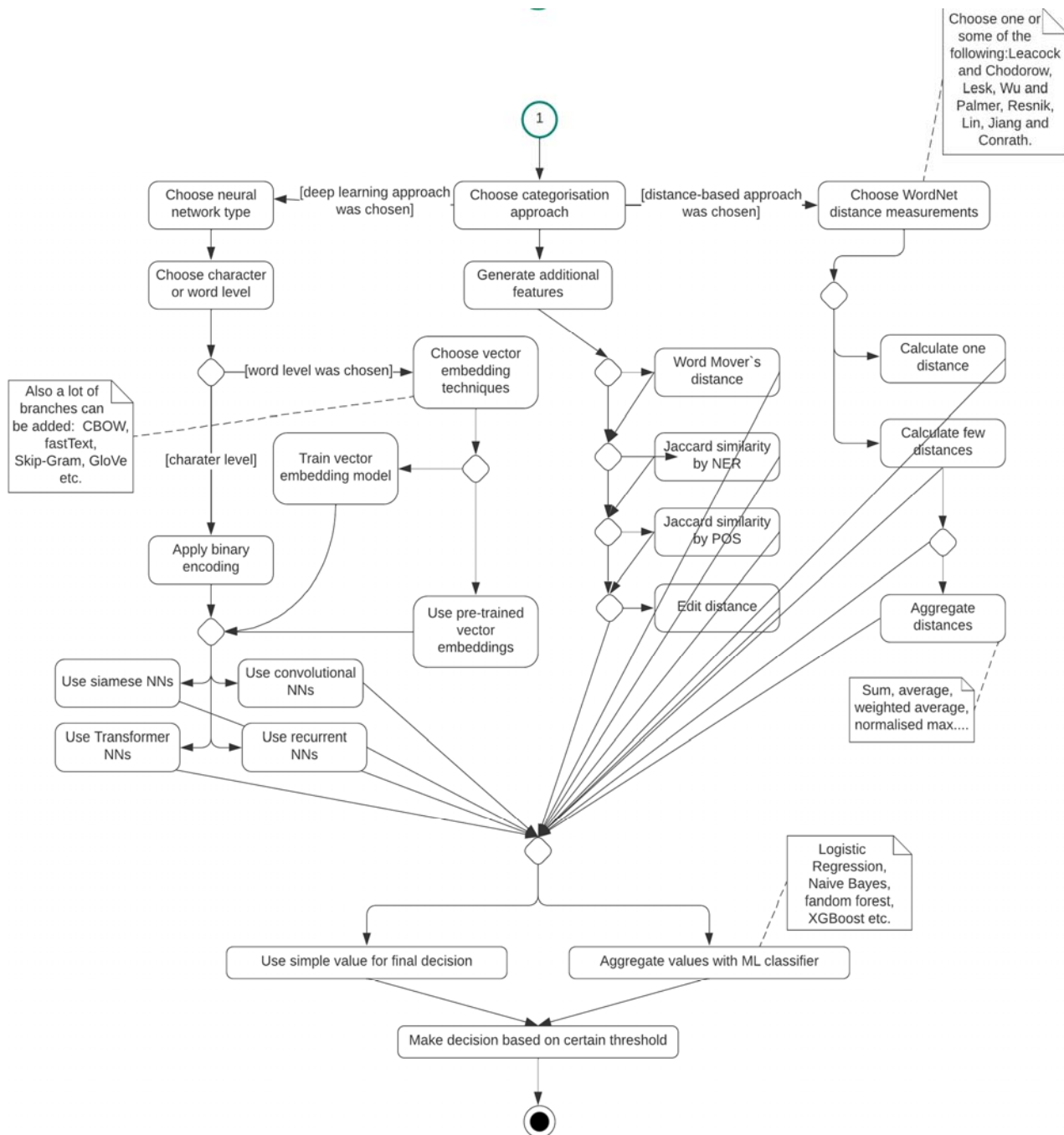


Рисунок 5 – Друга частина діаграми активності створення системи виявлення перефразувань

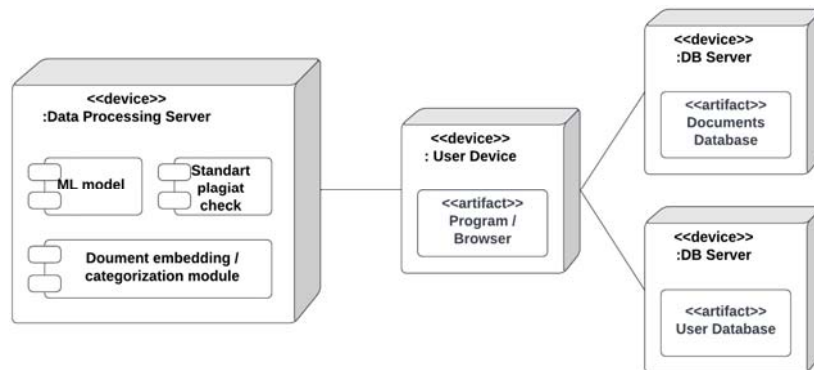


Рисунок 6 – Діаграма розгортання автоматичного виявлення перефразувань

відстань між векторними поданнями речень, Word Mover's Distance [43], відстані за словниками WordNet [1–2], передбачення двох ML-моделей: сіамської NN на основі рекурентних та типу Transformer – RoBERTa [46].

N-грам – послідовність з n слів. У контексті виявлення перефразувань у тексті кількість спільних N-грамів, нормована на загальну кількість N-грам у обох реченнях, допомагає виявити семантично подібні речення, що є близькими за семантичним навантаженням, однак не є перефразуваними, оскільки друге речення отримане шляхом перестановки слів у першому реченні, і, відповідно, має зовсім відмінне значення. Коефіцієнт Жаккара для двох множин розраховуємо наступним чином:

$$J(A, B) = |A \cap B| / |A \cup B|.$$

Для розрахунку N-грам у кожному реченні речення спочатку приводяться до нижнього регістру, видаляються усі розділові і додаткові знаки. Усього розраховані коефіцієнти Жаккара для 2-, 3-, 4-грам.

Є велика різноманітність методів отримання векторних вкладень слів GloVe, Word2Vec: CBOW / Skip-Gram, fastText. У дослідженні використані векторні вкладення моделі глибокого навчання BERT [42] для NLP архітектури Transformer. Базова модель BERT має 110 мільйонів параметрів, що налаштовуються, розширена версія – 345.

Особливістю архітектури Transformer є наявність механізму уваги, завдяки чому дані опрацьовують одночасно (на протигагу рекурентним NN, де дані сприймаються послідовно). Окрім того, особливістю BERT є попереднє навчання NN для вирішення двох завдань: передбачення певного слова у реченні і визначення того, чи є друге речення логічним продовженням першого. У [42] ця ML-модель попередньо навчена на нерозмічених даних з BooksCorpus (800 мільйонів слів) та English Wikipedia (2 500 мільйонів слів). Попереднє навчання та механізм уваги дають змогу отримати контекстні векторні вкладення слів.

Використовуючи попередньо навчену модель, матриця векторного подання для кожного речення має таку розмірність: torch.Size ([1, 128, 768]), де 1 – кількість речень у batch, 128 – максимальна довжина

речення, 768 – розмірність векторного вкладення. Для векторного подання речень дані усереднені для кожного слова у реченні, в результаті отримуємо вектор вкладення довжиною 768 значень.

Формула розрахунку косинусної відстані:

$$C = (\sum A_i B_i) / ((\sum A_i^2)^{1/2} (\sum B_i^2)^{1/2}).$$

Дана ознака потенційно виявляє семантично подібні речення, однак речення з однаковими словами, що не є перефразуваними, матимуть невелику косинусну відстань.

Word Mover's Distance застосовує векторні вкладення для розрахунку семантичної відстані між реченнями. WMD-відстань вимірює різницю між двома текстовими документами як мінімальну відстань, яку векторні вкладення слів одного документа повинні «полати», щоб досягти точок векторного вкладення слів іншого документа. Аналогічно до попередньої ознаки, така метрика відстані дозволить виявити семантично подібні пари слів, однак не допоможе виявити приклади неперефразованих речень із переставленими словами.

Для виміру семантичної подібності речень використовують дві метрики семантичної відстані синсетів за словниками WordNet: Leacock and Chodorow [1], Wu and Palmer [2]. Семантична подібність слів за Leacock and Chodorow [7] визначається за допомогою наступної формули:

$$\text{sim}_{lch} = -\log (\text{length} / (2 * D)).$$

де length – довжина найкоротшого шляху між двома концептами (кількість вузлів), D – максимальна глибина відповідної таксономії.

Функція семантичної відстані Wu and Palmer [9] залежить від глибини двох концептів $d(\text{concept}_i)$ у таксономії та глибини їх найближчого спільного предка $d(LCS)$:

$$\text{sim}_{wu_p} = 2 * d(LCS) / (d(\text{concept}_1) + d(\text{concept}_2)).$$

Оскільки за WordNet відстань розраховується для кожної пари синсетів окремо, для подання відстані між двома реченнями використаний підхід з [3]. Для впровадження двонапрямленості використовують середнє арифметичне двох значень певної відстані, значення якої залежить від максимальної подібності

пари слів. Однак, у даному випадку не враховується специфічність слів (відсутнє множення на значення TF-IDF). Таким чином, для розрахунку відстаней між парою речень потрібно порівняти кожне слово першого із кожним словом другого.

Сіамські NN використовують особливі функції втрат, оскільки даний тип NN вивчає такі внутрішні векторні подання даних, що однакові записи матимуть малу косинусну або Евклідову відстань. У даному дослідженні у якості функції втрат contrastive loss розрахована для Евклідової відстані:

$$L = 0,5 (1-Y) E^2 + 0,5 Y \{\max (0, m-E)\}^2,$$

де E – значення Евклідової відстані між передбаченими векторними поданнями записів, Y – істинне значення, m – поріг приналежності записів до одного класу: передбачена відстань між векторними поданнями різних класів має бути не $< m, m = 2$.

4 ЕКСПЕРИМЕНТИ

Оскільки навчальна вибірка набору даних PAWS-Wiki містить 49 401 запис і розрахунок кожної ознаки вимагає значних обчислювальних ресурсів, то для кожної ознаки спочатку створений репозиторій із відповідним скриптом опрацювання та результатами розрахунків. Проект має таку структуру (рис. 7).

Для кожної ознаки є репозиторій distances, що містить відповідні результати опрацювання. Кожний такий репозиторій містить по три файли – результати опрацювання тестової, навчальної, валідаційної вибірок відповідно. Для навчання NN (сіамської з рекурентними зв'язками та RoBERTa) використані обчислювальні ресурси Google Colab, тому передбачення сіамської NN завантажені локально, а у випадку RoBERTa завантажені лише збережені ваги у форматі .ckpt, опрацювання і класифікація відбувалась локально. Файл merge.py створений для об'єднання усіх ознак (рис. 8), в результаті утворюється тип даних бібліотеки Pandas DataFrame. Ці ж дані зберігаються у форматі .csv у репозиторії data/features для подальшого опрацювання.

З рис. 9–13 видно, що передбачення NN RoBERTa майже повністю співпадають з істинними мітками, що відповідають парі слів. Оскільки інші показники не є цілочисельними і позначають за своїм принципом зовсім різні ознаки, для їх об'єднання був обраний метод саме логістичної регресії. Її навчання прописане у файлі train_model.py, а сама модель (як і

інші класичні ML-методи, що порівнювались) імпортована з бібліотеки sklearn. Файл model_all_features.pkl містить ваги логістичної регресії, модель якої була навчена на всіх згенерованих ознаках. Подальше опрацювання текстів за допомогою натренованих ML-моделей вимагає об'єднання розрахунку усіх ознак, що виконано у файлі main.py. Із відповідних файлів імпортовані відповідні функції і попередньо навчені моделі.

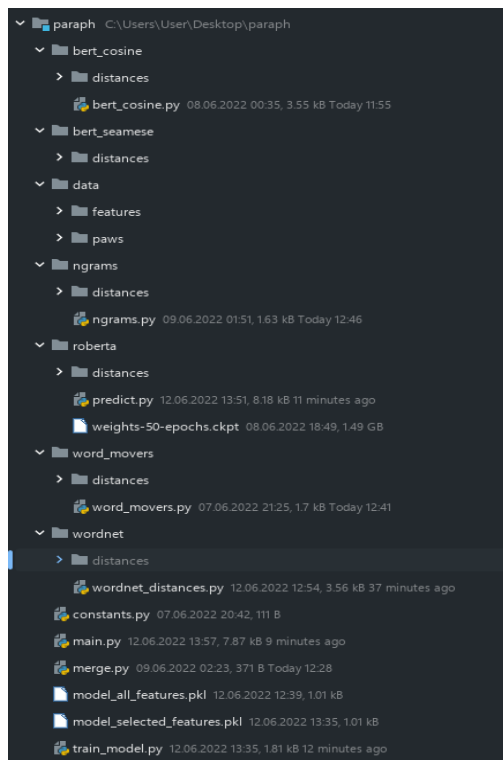


Рисунок 7 – Структура репозиторію проекту

```
import glob
import pandas as pd

for PART in ['dev', 'test', 'train']:
    files = glob.glob(f'C:/Users/User/Desktop/paraph/**/distances/{PART}*.csv',
                    recursive=True)
    df = pd.concat(map(pd.read_csv, files), axis=1)
    df.drop(columns='id', inplace=True)
    df.to_csv(f'data/features/{PART}.csv', index=False)
    print('ready')
```

Рисунок 8 – Об'єднання файлів з ознаками

id	sentence1	sentence2	label
1	This was a series of nested angular standards...	This was a series of nested polar scales , so...	0
2	His father emigrated to Missouri in 1868 but ...	His father emigrated to America in 1868 , but...	0
3	In January 2011 , the Deputy Secretary Genera...	In January 2011 , FIBA Asia deputy secretary ...	1
4	Steiner argued that , in the right circumstan...	Steiner held that the spiritual world can be ...	0
5	Luciano Willames Dias (born July 25 , 1970 ...	Luciano Willames Dias (born 25 July 1970) ...	0
6	During her sophomore , junior and senior summ...	During her second , junior and senior summers...	1
7	The smallest number that can be represented i...	The smallest number that can be represented a...	0
8	His father emigrated to Missouri in 1868 , bu...	His father emigrated to Missouri in 1868 but ...	1
9	The Villa Pesquera facilities are owned by th...	The facilities of Villa Pesquera are operated...	0
10	It is situated south of Köroğlu Mountains and...	It is situated south of Köroğlu - mountains a...	1

Рисунок 9 – Приклад вхідних даних

bert_cosine_distance	prediction_seamse_bert	3_grams_jaccard	2_grams_jaccard	4_grams_jaccard
0.99233836	0.2205543518066406	0.4117647058823529	0.5625	0.3142857142857143
0.98792297	0.4695497751235962	0.5517241379310345	0.8076923076923077	0.4827586206896552
0.9849439	0.6481984257698059	0.1724137931034483	0.44	0.06666666666666666
0.9759501	0.6250963807106018	0.2162162162162162	0.3823529411764705	0.1025641025641025
0.9834332	0.9546312689781188	0.375	0.5	0.25
0.9871587	0.7835149765014648	0.5769230769230769	0.72	0.5
0.98474044	0.856931209564209	0.2758620689655172	0.3928571428571428	0.2068965517241379
0.9874414	0.1621454060077667	0.8	0.88	0.72
0.9863758	1.0148042440414429	0.3478260869565217	0.4761904761904761	0.16
0.98016256	0.9825689196586608	0.4	0.5333333333333333	0.3571428571428571

Рисунок 10 – Відповідні вхідним даним ознаки

s_jaccard	predictions_raw	predictions	shortest_path_distance	wup_similarity	wm_distance
57142857143	0.0003076210268773	0	0.8808753618444751	0.9334199974323896	0.1638981500140085
86206896552	0.0001363550400128	0	0.9666656444455284	0.9749989694455368	0.1046792674400867
66666666666	0.9953380823135376	1	0.7899293829589504	0.8570820710088407	0.1082155853879519
41025641025	0.0146335149183869	0	0.8833327625003693	0.9286204319799678	0.258732279284138
0.25	0.004495037253946	0	0.9999987500015626	0.9999987500015626	0.0
0.5	0.9972121119499208	1	0.8849394652984164	0.9198043154376784	0.1023514166322542
65517241379	0.0016741530271247	0	0.9220770838260276	0.9545445867776484	0.1253511271784937
0.72	0.9963889122009276	1	0.9666656444455284	0.9749989694455368	0.1046792674400867
0.16	0.0004754920082632	0	0.999998819445843	0.999998819445843	0.0
28571428571	0.996845006942749	1	0.999998000004	0.999998000004	0.0

Рисунок 11 – Відповідні вхідним даним ознаки

```

from roberta.predict import Pairs_Dataset, config, Classifier_Model, predict
from transformers import AutoTokenizer
model_name = config['model_name']

tokenizer = AutoTokenizer.from_pretrained(model_name)

dataset_test = Pairs_Dataset(path, tokenizer, 'Quality', '#1 String', '#2 String')

BATCH_SIZE = config['batch_size']
test_loader = DataLoader(dataset_test, BATCH_SIZE, shuffle=False)

model_roberta = Classifier_Model.load_from_checkpoint('roberta/weights-50-epochs.ckpt', config=config)
model_roberta.to(device)

predictions_train_raw, predictions_train_int = predict(test_loader, model_roberta)
    
```

Рисунок 12 – Приклад класифікації за допомогою ML-моделі RoBERTa і імпортованих з інших файлів функцій

```

cos = get_cosine(df, 'sentence1', 'sentence2')
gr_2 = n_gr_sim(df, 2, 'sentence1', 'sentence2')
gr_3 = n_gr_sim(df, 3, 'sentence1', 'sentence2')
gr_4 = n_gr_sim(df, 4, 'sentence1', 'sentence2')
wn = wordnet_distance(df, 'sentence1', 'sentence2')
wm = word_movers(df, 'sentence1', 'sentence2')

X_test = pd.DataFrame({
    'bert_cosine_distance': cos,
    '2_grams_jaccard': gr_2,
    '3_grams_jaccard': gr_3,
    '4_grams_jaccard': gr_4,
    'predictions_raw': predictions_train_raw,
    'predictions': predictions_train_int,
    'shortest_path_distance': wn,
    'wm_distance': wm
})
    
```

Рисунок 13 – Створення таблиці з ознаками

NN містить два шари двонапрямленої довгої короткочасної пам'яті (LSTM), два прихованих шари повнозв'язних нейронів (512 та 128 нейронів), вихідний шар, що містить 16 нейронів. Векторні вкладення слів були отримані шляхом прямого поширення у попередньо натренованій NN BERT. NN навчалась протягом 30 епох на навчальній вибірці із валідацією після кожної епохи (рис. 14).

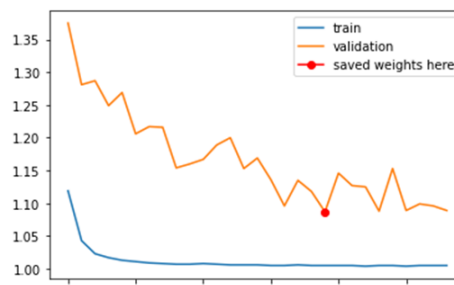


Рисунок 14 – Навчання сіамської NN

Значна різниця між функціями втрат на навчальній та валідаційній вибірці свідчить про недостатню комплексність моделі (можливо, про недостатню кількість прихованих шарів або нейронів). Окрім того, точність класифікації може залежати від обраних гіперпараметрів, налаштування яких вимагає додаткового навчання і тестування моделі. Для подальшої класифікації ваги NN збережені за найменшого значення функції втрат (19-та епоха).

Попередньо навчена NN RoBERTa є покращеним аналогом BERT: основна її відмінність полягає у підборі гіперпараметрів під час попереднього навчання, збільшенні об'єму навчального корпусу

(16GB речень з Books Corpus та English Wikipedia, CommonCrawl News dataset (63 мільйони статей, 76 GB), Web text corpus (38 GB), Stories from Common Crawl (31 GB)), застосуванні динамічного маскування токенів для задачі передбачення слова у реченні: слово, яке необхідно передбачити у певному реченні, змінюється з кожною епохою. RoBERTa має 124 мільйони налаштовуваних параметрів (рис. 15). Оскільки результатом прямого поширення є матриця векторного вкядення речень розмірністю [batch size, max sentence length, embedding dimension], для подальшої класифікації додано прихований та повнозв'язний вихідний шар з 256 та 1 нейронами відповідно. Для знаходження векторного подання речень отримано середнє значення для кожної координати векторів вкядень. Результуюча матриця ваг між отриманими векторними поданнями слів та прихованим шаром із 256 нейронами матиме розмірність 768*256, між прихованим та вихідним шаром – 256*1. Активаційною функцією останнього шару є сигмоїда, функцією втрат – бінарна крос-ентропія.

	Name	Type	Params
0	pretrained_model	RobertaModel	124 M
1	dropout_layer	Dropout	0
2	hidden_layer	Linear	196 K
3	output_layer	Linear	257
4	loss_function	BCEWithLogitsLoss	0

Рисунок 15 – Загальна кількість параметрів NN

Попередньо навчена NN додатково налаштована (fine-tuned) для задачі виявлення перефразувань у тексті за допомогою навчальної та валідаційної вибірок. Усього NN додатково навчалась протягом 50 епох, ваги збережені за умови найменшого значення функції втрат на валідаційній вибірці (рис. 16).

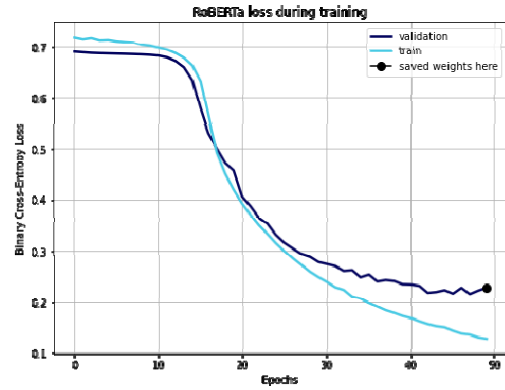


Рисунок 16 – Функція втрат під час навчання NN

Для фінального об'єднання ознак обрані як ймовірності приналежності двох записів до певного класу, так і самі передбачення. Поріг прийняття рішення = 0,5, тобто $\hat{y} = 1$ якщо $p(y') > 0,5$. Фінальна таблиця містить 10 ознак із вищезазначеними метриками для кожної пари речень, наявних у обраному наборі даних. Ці показники розраховані для навчальної, тестової і валідаційної вибірки.

5 РЕЗУЛЬТАТИ

Для зменшення навантаження на обчислювальні ресурси навчено іншу модель логістичної регресії без таких ознак: передбачення сіамської NN та відстані WordNet за Wu and Palmer. Точність класифікації на тестовій вибірці збільшилась з 92,4625% до 92,5%.

Для класифікації за допомогою NN RoBERTa дані подаються у наступному форматі: індекси слів, бінарна маска, що позначає частину корисного сигналу, істинна мітка, що відповідає парі записів як на рис. 17. Для отримання інших ознак дані приводяться до нижнього регістру, видаляються усі символи, окрім літер (рис. 18).

```
In [5]: dataset_test[0]
Out[5]: {'input_ids': tensor([ 0, 713, 21, 10, 651, 9, 46902, 42970, 2820, 2156,
    98, 14, 19851, 11, 15001, 757, 5914, 8, 25361, 115,
    28, 626, 2024, 11, 13744, 34721, 5407, 7, 5, 364,
    3998, 10455, 636, 479, 2, 2, 713, 21, 10, 651,
    9, 46902, 13744, 21423, 2156, 98, 14, 19851, 11, 15001,
    757, 5914, 8, 25361, 115, 28, 3744, 2024, 11, 42970,
    34721, 5407, 7, 5, 364, 3998, 10455, 636, 479, 2,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1]),
  'attention_mask': tensor([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0]),
  'labels': tensor(0.)}
```

Рисунок 17 – Формат вхідних даних до нейронної мережі RoBERTa

bert_cosine_distance	2_grams_jaccard	3_grams_jaccard	4_grams_jaccard	predictions_raw	predictions	shortest_path_distance	wm_distance
0.992338	0.5625	0.411765	0.314286	0.000308	0	0	0.163898

Рисунок 18 – Отримані ознаки для тестової пари речень

Для передбачення завантажуються ваги попередньо навченого класифікатора (логістичної регресії) як показано на рис. 19, рис. 20.

```
import pickle
loaded_model = pickle.load(open('model_selected_features.pkl', 'rb'))
predicted = loaded_model.predict(X_test)
print('Predicted label', predicted)
```

Рисунок 19 – Попередньо навчена модель

```
Predicted label [0]
```

Рисунок 20 – Результат передбачення

```
['This was a series of nested angular standards , so that measurements in azimuth and elevation could be done directly in polar coordinates relative to the ecliptic .']
['This was a series of nested polar scales , so that measurements in azimuth and elevation could be performed directly in angular coordinates relative to the ecliptic .']
```

Рисунок 21 – Перша тестова пара речень

```
Predicted label 1
In [8]: print(df.loc[2, 'sentence1'])
...: print(df.loc[2, 'sentence2'])
...:
In January 2011 , the Deputy Secretary General of FIBA Asia , Hagop Khajirian , inspected the venue together with SBP - President Manuel V. Pangilinan .
In January 2011 , FIBA Asia deputy secretary general Hagop Khajirian along with SBP president Manuel V. Pangilinan inspected the venue .
```

Рисунок 22 – Друга тестова пара речень

Програма не потребує реалізації користувацького інтерфейсу, оскільки її використання заплановано лише у якості модулю системи виявлення плагіату або об'єднання згенерованого користувачами контенту.

6 ОБГОВОРЕННЯ

Для об'єднання ознак і фінальної класифікації обраний класичний ML-алгоритм – логістична регресія. Отримані результати такі результати:

– Точність на тестовому наборі даних – 92,462%, площа під ROC-кривою становить 97,05%, під кривою Precision-Recall – 94,96%.

– Точність на валідаційному наборі даних – 93,71% площа під ROC-кривою становить 97,66%, під кривою Precision-Recall – 96,12%.

Відповідно до показників влучності, повноти, $F_{0.5}$ -міри (рис. 23) і матриці невідповідностей (рис. 24) результату класифікації тестового набору даних, логістична регресія помилково позначила майже вдвічі більше негативних записів як позитивні, ніж навпаки, позитивні – негативними. Даний результат може бути наслідком складності визначення не перефразованої пари речень, що були утворені в результаті заміни кількох слів місцями. Такі речення є дуже близькими семантично, однак мають абсолютно різне значення.

	precision	recall	f1-score	support
0	0.95	0.91	0.93	4464
1	0.89	0.94	0.92	3536
accuracy			0.92	8000
macro avg	0.92	0.93	0.92	8000
weighted avg	0.93	0.92	0.92	8000

Рисунок 23 – Повнота, влучність, F-міра для тестового набору даних

Результат класифікації, отриманий для наступного речення, подано на рис. 21. Логістична мережа правильно передбачила, що два речення не є перефразуваннями один одного, навіть попри те, що більшість слів у реченнях є спільними (рис. 22). Для наступного речення логістична регресія також правильно передбачила мітку, у цьому випадку пара речень є перефразуваннями один одного – подано на рис. 22.

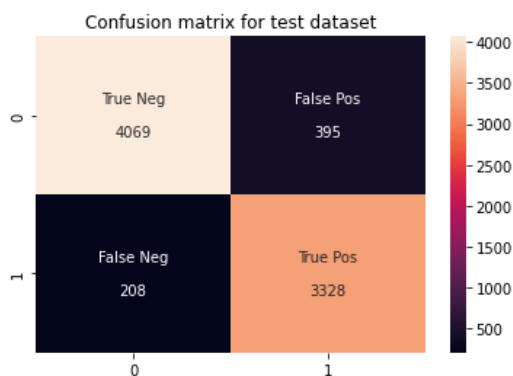


Рисунок 24 – Матриця невідповідностей для тестового набору даних

Відповідно до стовпчикової діаграми на рис. 25, найбільш важливими ознаками є передбачення ML-моделі RoBERTa, косинусна відстань між векторними поданнями рішень та найкоротший шлях між синетами словника WordNet за Leacock and Chodorow. Значний коефіцієнт логістичної регресії (рис. 26), що відповідає передбаченням ML-моделі, свідчить про її потенційну можливість самостійно якісно класифікувати тексти без необхідності розрахунку додаткових ознак.

Без застосування методів глибинного навчання, використовуючи лише метод логістичної регресії для об'єднання результатів, отримуємо (рис. 27, рис. 28):

– Точність класифікації на тестовому наборі даних – 71,15%, площа під кривою Precision-Recall – 73,6%.

Найбільш важливою ознакою для класифікації є коефіцієнт Жаккарда для 3-грам (нормована кількість спільних 3-грам) та косинусна відстань між векторними поданнями речень.

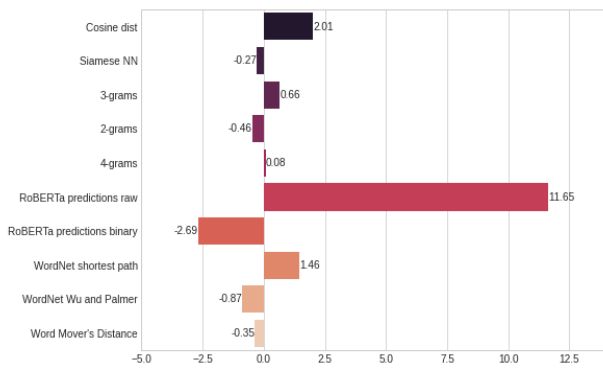


Рисунок 25 – Вагові коефіцієнти логістичної регресії, що відповідають певним ознакам семантичної подібності пари речень

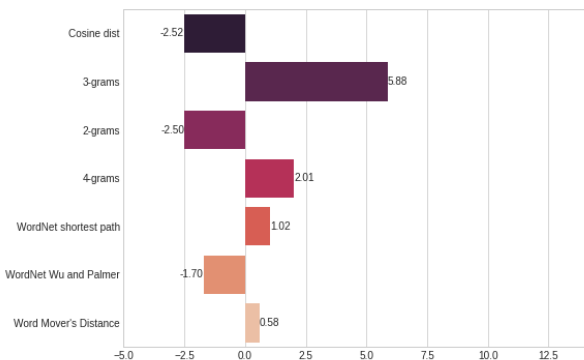


Рисунок 26 – Вагові коефіцієнти логістичної регресії, що відповідають певним ознакам семантичної подібності пари речень

На відміну від попереднього результату класифікації із використанням методів глибокого навчання, у цьому випадку вже більша частина позитивних випадків була класифікована як негативні, тобто модель логістичної регресії «має труднощі» із визначенням перефразувань.

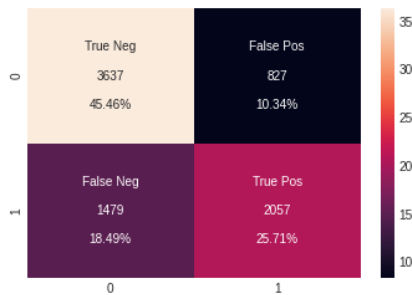


Рисунок 27 – Матриця невідповідностей для тестового набору даних

Використання лише однієї ознаки не є достатнім для якісної класифікації пари речень, оскільки передбачення такої моделі близькі за ймовірністю до «сліпого» вгадування: точність результатів класифікації тестового набору даних з використанням Word Mover's Distance становить 55,7%, косинусної відстані між векторними поданнями речень – 55,7%, передбачень сіамської NN – 58%, у той час як поєднання цих же метрик із коефіцієнтом Жаккарда

для 3-грамів збільшує точність класифікації до 70%, а застосування коефіцієнту Жаккарда для 2-грам додає ще 1% точності.

Попри те, що дані ознаки можуть свідчити про семантичну подібність речень і можуть бути використані для виявлення пар подібних речень, вищезазначеної точності не є достатньо для повноцінного і якісного виявлення перефразувань.

Самостійне передбачення навченої ML-моделі RoBERTa є досить якісним і має високі показники якості: точність – 91,96%, площа під кривою Precision-Recall – 96,34%.

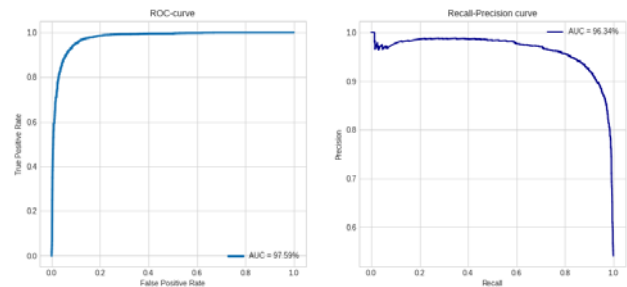


Рисунок 28 – Криві ROC та Precision-Recall для результатів класифікації моделі глибокого навчання RoBERTa

При цьому, RoBERTa значно більше (у 6 разів) класифікує негативні класи як позитивні (рис. 29), при цьому значно зменшуючи чутливість / повноту (recall) до пар речень, що не є перефразуваннями один одного.

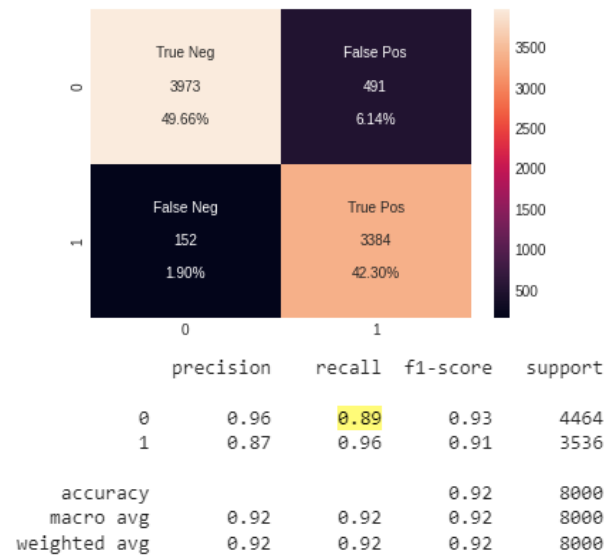


Рисунок 29 – Матриця невідповідностей, влучність, повнота, F-міра

При використанні такої моделі для виявлення плагиату більше робіт будуть неправильно вказані як першопочаткові джерела, при використанні для об'єднання згенерованого користувачами контенту (запитання, записи, теми на форумі) зростає ймовірність неправильно співвіднести певну кількість

записів. Окрім порівняння ML-методів також одночасно протестовано різну кількість ознак. Ознаки обрані в залежності від відповідної ваги моделі логістичної регресії. Найкраща точність була отримана для повного набору ознак та вищезазначеного алгоритму. Інші класичні ML-методи (протестовано наївний Байєсів класифікатор, метод опорних векторів, випадковий ліс, k найближчих сусідів багатосаровий перцептрон) не дають суттєвого приросту точності навчання. Щодо розробки проекту можна зробити такі висновки:

– для методу стекингу моделей і ознак отримані високі показники точності, влучності, повноти, $F_{0.5}$ -міри, площ під кривими ROC та Precision-Recall;

– «класичні» ознаки семантичної подібності, що використовуються у багатьох дослідженнях, справді здатні виявити семантично подібні речення, однак вони є «безсилимими» у випадках, коли у реченні є переставлені слова (або є багато спільних слів) і, відповідно, друге речення не є перефразуванням першого і має зовсім інший зміст;

– основною передумовою такого результату є те, що сконструйовані ознаки не беруть до уваги порядок слів у реченні. Для боротьби з цим використовується коефіцієнт Жаккарда і об'єднання ознак за допомогою класичних ML-методів;

– попередньо навчені ML-моделі на основі архітектури Transformer показують відмінну точність класифікації. При цьому, навчена у процесі виконання роботи NN RoBERTa (із додатковими повнозв'язними шарами) має меншу чутливість до пар речень, що не є перефразуваннями один одного. Така специфічність моделі може сприяти неправильному звинуваченню у плагіаті або некоректному об'єднанню згенерованого користувачами контенту;

– перед розробниками ІС виявлення антиплагіату може постати питання вибору між точністю класифікації і збереженням обчислювальних ресурсів, оскільки розрахунок ознак також сприяє додатковому навантаженню на обчислювальний пристрій;

– NN архітектури Transformer не вимагають додаткової генерації ознак і здатні виявляти перефразування із досить високою точністю. Недоліком такого типу NN є значна кількість параметрів (великий час розрахунку результатів);

– Перевагою NN типу Transformer є попереднє навчання моделей для «розуміння мови» за допомогою завдань передбачення найбільш ймовірних слів у реченні та визначення того, чи є друге речення ідейним продовженням другого;

– для задачі виявлення перефразувань NN «дотренована» (fine-tuned) на наборі даних Paraphrase Adversaries from Word Scrambling. Перевагами обраного набору даних є 1) велика кількість навчальних записів – 49 тис. 2) збалансованість класів: 44.2% з усіх пар речень є перефразуваннями один одного. 3) частина прикладів була утворена шляхом заміни або перестановки слів, у такому

випадку речення є близькими семантично, однак мають зовсім інші значення;

– NN типу Transformer застосовуються у багатьох NLP-задачах, їх також можна успішно використовувати для виявлення перефразувань у тексті з високою точністю. Єдиним недоліком такого типу мереж є значна кількість налаштовуваних параметрів – 110+ мільйонів, навчання такої NN і її застосування вимагають наявності значних обчислювальних ресурсів.

ВИСНОВКИ

Результатом роботи є розроблена ML-модель для виявлення перефразувань шляхом бінарної класифікації пари текстів. Розроблене ПЗ використовує принцип стекингу моделей і інжиніринг ознак (feature engineering). Додаткові ознаки вказують на семантичну приналежність речень або нормовану кількість спільних N-грам. Створена модель показує відмінні результати класифікації на тестових даних PAWS: зважена влучність (precision) – 93%, зважена повнота (recall) – 92%, F-міра (F1-score) – 92%. Результати дослідження показали, що NN типу Transformer можуть бути успішно застосовані для виявлення перефразувань у парі текстів із досить високою точністю без потреби додаткового генерування ознак.

Додатково налаштована (fine-tuned) NN RoBERTa (із додатковими повнозв'язними шарами) має меншу чутливість до пар речень, що не є перефразуваннями один одного. Така специфічність моделі може сприяти неправильному звинуваченню у плагіаті або некоректному об'єднанню згенерованого користувачами контенту. Додаткові ознаки збільшують як загальну точність класифікації, так і чутливість моделі до пар тих речень, що не є перефразуваннями один одного.

ПОДЯКИ

Роботу виконано в рамках держбюджетної теми «Методи та засоби функціонування систем підтримки прийняття рішень на основі онтологій» (ID:839 2017-05-15 09:20:01 (2459-315)). Дослідження провадилося в межах спільних наукових досліджень кафедри інформаційних систем та мереж НУ «Львівська політехніка» на тему «Дослідження, розроблення і впровадження інтелектуальних розподілених інформаційних технологій та систем на основі ресурсів баз даних, сховищ даних, просторів даних та знань з метою прискорення процесів формування сучасного інформаційного суспільства». Наукові дослідження провадилися також в рамках ініціативної тематики досліджень кафедри ІСМ НУ «Львівська політехніка» на тему «Розроблення інтелектуальних розподілених систем на основі онтологічного підходу з метою інтеграції інформаційних ресурсів».

ЛІТЕРАТУРА / LITERATURE

1. Salton G. A vector space model for automatic indexing / G. Salton, A. Wong, C.-S. Yang // *Communications of the ACM*. – 1975. – Vol. 18(11). – P. 613–620. DOI: 10.1145/361219.361220
2. Turney P. D. From Frequency to Meaning: Vector Space Models of Semantics / P. D. Turney, P. Pantel // *Journal of Artificial Intelligence Research*. – 2010. – Vol. 37(1). – P. 141–188. DOI: 10.1613/jair.2934
3. Efficient Estimation of Word Representations in Vector Space / [T. Mikolov, K. Chen, G. s Corrado, J. Dean] // *ArXiv*. – 2013. DOI: 10.48550/arXiv.1301.3781
4. Do Online Plagiarism Checkers Identify Paraphrased Content?. – DotNek Software Development, 2021. – <https://www.dotnek.com/Blog/Marketing/do-online-plagiarism-checkers-identify-paraph>
5. Introduction to WordNet: An On-line Lexical Database/ [G. Miller, R. Beckwith, C. Fellbaum, D. Gross, K. Miller] // *International Journal of Lexicography*. – 1990. – Vol. 3(4). – P. 235–244. DOI: 10.1093/ijl/3.4.235
6. Corley C. Measuring the Semantic Similarity of Texts / C. Corley, R. Mihalcea // *ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, Ann Arbor, Michigan, Jun. 2005 : proceedings. – P. 13–18.
7. Leacock C. Combining Local Context and WordNet Similarity for Word Sense Identification / C. Leacock, M. Chodorow // *WordNet: An Electronic Lexical Database*. – 1998. – Vol. 49(2). – P. 265–283. DOI: 10.7551/mitpress/7287.003.0018
8. Lesk M. E. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone / M. E. Lesk // *SIGDOC '86: the 5th Annual International Conference on Systems documentation*, Toronto, Ontario, Canada, June 1986 : proceedings. – P. 24–26. DOI: 10.1145/318723.318728
9. Wu Z. Verbs Semantics and Lexical Selection / Z. Wu, M. Palmer // *ACL '94: the 32nd annual meeting on Association for Computational Linguistics*, Las Cruces, New Mexico, June 27 – 30, 1994 : proceedings. – P. 133–138. DOI: 10.3115/981732.981751.
10. Resnik P. Using Information Content to Evaluate Semantic Similarity in a Taxonomy / P. Resnik // *ArXiv*. – 1995. DOI: 10.48550/arXiv.cmp-lg/9511007
11. Lin D. An Information-Theoretic Definition of Similarity / D. Lin // *ICML, 1998*. – <https://www.cse.iitb.ac.in/~cs626-449/Papers/WordSimilarity/3.pdf>
12. Jiang J. J. Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy / J. J. Jiang, D. W. Conrath // *10th Research on Computational Linguistics International Conference*, Taipei, Taiwan, Aug. 1997 : proceedings. – P. 19–33.
13. Mihalcea R. Corpus-based and Knowledge-based Measures of Text Semantic Similarity / R. Mihalcea, C. Corley, C. Strapparava // *AAAI'06: Proceedings of the 21st national conference on Artificial intelligence*, Boston, Massachusetts, July 2006 : proceedings. – Vol. 1. – P. 775–780.
14. Hassan S. Semantic Relatedness Using Salient Semantic Analysis / S. Hassan, R. Mihalcea // *AAAI 2011: Twenty-Fifth AAAI Conference on Artificial Intelligence*, San Francisco, California, USA, August 7–11, 2011 : proceedings. – <https://web.eecs.umich.edu/~mihalcea/papers/hassan.aaai11.pdf>
15. Fernando S. A Semantic Similarity Approach to Paraphrase Detection / S. Fernando, M. Stevenson // *11th Annual Research Colloquium of the UK Special Interest Group for Computational Linguistics*, May 2008 : proceedings. – P. 45–52.
16. Milajevs D. Evaluating Neural Word Representations in Tensor-Based Compositional Settings / [D. Milajevs, D. Kartsaklis, M. Sadrzadeh, M. Purver] // *ArXiv*. – 2014. DOI: 10.48550/arXiv.1408.6179
17. Islam A. Semantic similarity of short texts / A. Islam, D. Inkpen // *Current Issues in Linguistic Theory: Recent Advances in Natural Language Processing*. – 2009. – Vol. 309. – P. 227–236. DOI: 10.1075/cilt.309.18isl
18. Chong M. Using Natural Language Processing for Automatic Detection of Plagiarism / M. Chong, L. Specia, R. Mitkov // *IPC2010 : 4th International Plagiarism Conference*, Newcastle-upon-Tyne, May 2010 : proceedings. – https://www.academia.edu/326444/Using_Natural_Language_Processing_for_Automatic_Detection_of_Plagiarism
19. TakeLab: Systems for Measuring Semantic Text Similarity / [F. Šarić, G. Glavaš, M. Karan, J. Šnajder, B. Dalbelo Bašić] // **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, Montréal, Canada, May 2012 : proceedings. – P. 441–448.
20. SemEval-2012 Task 6: A Pilot on Semantic Textual Similarity / [E. Agirre, D. Cer, M. Diab, A. Gonzalez-Agirre] // **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, Montréal, Canada, May 2012 : proceedings. – P. 385–393.
21. Detecting High Obfuscation Plagiarism: Exploring Multi-Features Fusion via Machine Learning / [L. Kong, Z. Lu, H. Qi, Z. Han] // *International Journal of u- and e- Service, Science and Technology*. – 2014. – Vol. 7. – P. 385–396. DOI: 10.14257/ijunnesst.2014.7.4.35
22. Yin W. Convolutional Neural Network for Paraphrase Identification / W. Yin, H. Schütz // *North American Chapter of the Association for Computational Linguistics: Human Language Technologies Conferences*, Denver, Colorado, May 2015 : proceedings. – P. 901–911. DOI: 10.3115/v1/N15-1091.
23. Qiu L. Paraphrase Recognition via Dissimilarity Significance Classification / L. Qiu, M.-Y. Kan, T.-S. Chua // *Empirical Methods in Natural Language Processing Conference*, Sydney, Australia, Jul. 2006 : proceedings. – P. 18–26.
24. Kozareva Z. Paraphrase Identification on the Basis of Supervised Machine Learning Techniques / Z. Kozareva, A. Montoyo // *Advances in Natural Language Processing. FinTAL 2006. Lecture Notes in Computer Science*. – 2006. – Vol. 4139. – P. 524–533. DOI: 10.1007/11816508_52
25. Finch A. Using machine translation evaluation techniques to determine sentence-level semantic equivalence / A. Finch, E. Sumita // *IWP2005 : 3rd International Workshop on Paraphrasing*, May 2005 : proceedings. – P. 17–24.
26. Madnani N. Re-examining Machine Translation Metrics for Paraphrase Identification / N. Madnani, J. Tetreault, M. Chodorow // *North American Chapter of the Association for Computational Linguistics: Human Language*

- Technologies Conference of the, Montréal, Canada, June 2012 : proceedings. – P. 182–190.
27. A Deep Network Model for Paraphrase Detection in Short Text Messages / B. Agarwal, H. Ramampiaro, H. Langseth, M. Ruocco // ArXiv. – 2017. DOI: 10.48550/arXiv.1712.02820
28. Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection / R. Socher, E. H.-C. Huang, J. Pennington, A. Ng, C. D. Manning // NIPS'11: the 24th International Conference on Neural Information Processing Systems, Granada Spain, December 2011 : proceedings. – P. 801–809.
29. Thyagarajan A. Siamese Recurrent Architectures for Learning Sentence Similarity / A. Thyagarajan // The Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, Arizona, USA, February 12–17, 2016 : proceedings. – Vol. 30(1). – P. 2786–2792. DOI: 10.1609/aaai.v30i1.10350
30. Neculoiu P. Learning Text Similarity with Siamese Recurrent Networks / P. Neculoiu, M. Versteegh, M. Rotaru // Workshop on Representation Learning for NLP, Berlin, Germany, August 2016 : proceedings. – P. 148–157. DOI: 10.18653/v1/W16-1617
31. Ranasinghe T. Semantic Textual Similarity with Siamese Neural Networks / T. Ranasinghe, C. Orasan, R. Mitkov // RANLP 2019 : International Conference on Recent Advances in Natural Language Processing, Varna, Bulgaria, Sep. 2019 : proceedings. – P. 1004–1011. DOI: 10.26615/978-954-452-056-4_116
32. Mahmoud A. BLSTM-API: Bi-LSTM Recurrent Neural Network-Based Approach for Arabic Paraphrase Identification / A. Mahmoud, M. Zrigui // Arabian Journal for Science and Engineering. – 2021. – Vol. 46. – P. 4163–4174. DOI: 10.1007/s13369-020-05320-w
33. Reddy D. LSTM Based Paraphrase Identification Using Combined Word Embedding Features / D. Reddy, M. Kumar, S. Kp // Computing and Signal Processing. Advances in Intelligent Systems and Computing. – 2019. – Vol. 898. – P. 385–394. DOI: 10.1007/978-981-13-3393-4_40
34. Paraphrase Generation with Deep Reinforcement Learning / [Z. Li, X. Jiang, L. Shang, H. Li] // ArXiv. – 2017. DOI: 10.48550/arXiv.1711.00279
35. Goma W. SimAll: A flexible tool for text similarity / W. Goma, A. Fahmy // ESOLEC' 2017 : The Seventeenth Conference on Language Engineering, December 2017 : proceedings. – P. 182–190. DOI: https://www.academia.edu/35381793/SimAll_A_flexible_tool_for_text_similarity
36. Ahmed M. Improving Tree-LSTM with Tree Attention / M. Ahmed, M. R. Samee, R. E. Mercer // ArXiv. – 2019. DOI: 10.48550/arXiv.1901.00066
37. Pontes E. L. Predicting the Semantic Textual Similarity with Siamese CNN and LSTM / E. L. Pontes, S. Huet, A. C. Linhares, J.-M. Torres-Moreno // Actes de la Conférence TALN, Rennes, France, May 2018 : proceedings. – P. 311–320.
38. Are Neural Language Models Good Plagiarists? A Benchmark for Neural Paraphrase Detection / [J. P. Wahle, T. Ruas, N. Meuschke, B. Gipp] // ArXiv. – 2021. DOI: 10.48550/arXiv.2103.12450
39. Attention Is All You Need / [A. Vaswani, N. Shazeer, N. Parmar et al.] // ArXiv. – 2017. DOI: 10.48550/arXiv.1706.0376
40. Nighojkar A. Improving Paraphrase Detection with the Adversarial Paraphrasing Task / A. Nighojkar, J. Licato // ArXiv. – 2021. DOI: 10.48550/arXiv.2106.07691
41. Arase Y. Transfer fine-tuning of BERT with phrasal paraphrases / Y. Arase, J. Tsujii // ArXiv. – 2021. DOI: 10.48550/arXiv.1909.00931
42. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding / [J. Devlin, M.-W. Chang, K. Lee, K. Toutanova] // ArXiv. – 2018. DOI: 10.48550/arXiv.1810.04805
43. From Word Embeddings to Document Distances / [M. J. Kusner, Y. Sun, N. I. Kolkin, K. Q. Weinberger] // JMLR: W&CP. – 2015. – Vol. 37. – P. 957–966.
44. Zhang Y. PAWS: Paraphrase Adversaries from Word Scrambling / Y. Zhang, J. Baldridge, L. He // ArXiv. – 2019. DOI: 10.48550/arXiv.1904.01130
45. Propaganda Detection in Text Data Based on NLP and Machine Learning / [V.-A. Oliinyk, V. Vysotska, Y. Burov, et al.] // Modern Machine Learning Technologies and Data Science (MoMLeT+DS 2020) : Workshop, Lviv-Shatsk, 2–3 June 2020 : CEUR workshop proceedings. – Aachen: CEUR-WS.org, 2020. – Vol. 2631. – P. 132–144.
46. RoBERTa: A Robustly Optimized BERT Pretraining Approach / [Y. Liu, M. Ott, N. Goyal et al.] // ArXiv. – 2019. DOI: 10.48550/arXiv.1907.11692

Стаття надійшла до редакції 23.06.2022.
Після доробки 28.09.2022.

УДК 004.9

ТЕХНОЛОГИЯ ИДЕНТИФИКАЦИИ РЕРАЙТА В ТЕКСТОВОМ КОНТЕНТЕ НА ОСНОВЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ

Холодна Н. М. – студент кафедри «Інформаційні системи і мережі», Національний університет «Львівська політехніка», Україна.

Висоцька В. А. – канд. техн. наук, доцент, доцент кафедри «Інформаційні системи і мережі», Національний університет «Львівська політехніка», Україна.

АННОТАЦІЯ

Актуальність. Перефразований текстовий контент або рерайт є однією з складних проблем виявлення академічного плагіату. Більшість систем ідентифікації плагіату призначені для виявлення загальних слів, послідовності лінгвістических одиниць і незначительних змін, але не здатні виявити суттєві семантичні та структурні зміни. Тому більшість випадків плагіату з використанням перефразування залишаються незамеченими.

Целью исследования является разработка технологии обнаружения перефразировок в тексте на основе модели классификации и методов машинного обучения через использование сиамской нейронной сети на основе рекуррентных и типа Transformer – RoBERTa для анализа уровня подобия предложений текстового контента.

Метод. Для данного исследования в качестве признаков выбраны следующие метрики семантического подобия или показатели: коэффициент Жаккара для общих N-грамм, косинусное расстояние между векторными представлениями предложений, Word Mover’s Distance, расстояния по словарям WordNet, предсказание двух ML-моделей: сиамской нейронной сети на основе рекуррентных и типа Transformer – RoBERTa.

Результаты. Разработана интеллектуальная система выявления перефразировок в тексте на основе модели классификации и методов машинного обучения. Разработанная система использует принцип стекинг-моделей и инжиниринг признаков (feature engineering). Дополнительные признаки указывают на семантическую принадлежность предложений или нормированное количество общих N-грамм. Дополнительно настроенная (fine-tuned) нейронная сеть RoBERTa (с дополнительными полносвязными слоями) имеет меньшую чувствительность к парам предложений, не являющимся перефразированием друг друга. Такая специфичность модели может способствовать неправильному обвинению в плагиате или некорректному объединении сгенерированного пользователями контента. Дополнительные признаки увеличивают как общую точность классификации, так и чувствительность модели к парам тех предложений, которые не являются перефразированием друг друга.

Выводы. Созданная модель показывает отличные результаты классификации на тестовых данных PAWS: взвешенная точность (precision) – 93%, взвешенная полнота (recall) – 92%, F-мера (F1-score) – 92%, точность (accuracy) – 92%. Результаты исследования показали, что NN типа Transformer могут быть успешно применены для обнаружения перефразирования в паре текстов с достаточно высокой точностью без необходимости дополнительного генерирования признаков.

КЛЮЧЕВЫЕ СЛОВА: обработка природного языка, NLP, идентификация рерайта, выявление перефразировок в тексте, машинное обучение с учителем, глубинное обучение, классификация текста, анализ текста, векторное вложение слов, WordNet, семантическое сходство.

UDC 004.9

REWRITING IDENTIFICATION TECHNOLOGY FOR TEXT CONTENT BASED ON MACHINE LEARNING METHODS

Kholodna N. – Student of Information Systems and Networks Department, Lviv Polytechnic National University, Lviv, Ukraine.

Vysotska V. – PhD, Associate Professor of Information Systems and Networks Department, Lviv Polytechnic National University, Lviv, Ukraine.

ABSTRACT

Context. Paraphrased textual content or rewriting is one of the difficult problems of detecting academic plagiarism. Most plagiarism detection systems are designed to detect common words, sequences of linguistic units, and minor changes, but are unable to detect significant semantic and structural changes. Therefore, most cases of plagiarism using paraphrasing remain unnoticed.

Objective of the study is to develop a technology for detecting paraphrasing in text based on a classification model and machine learning methods through the use of Siamese neural network based on recurrent and Transformer type – RoBERTa to analyze the level of similarity of sentences of text content.

Method. For this study, the following semantic similarity metrics or indicators were chosen as features: Jacquard coefficient for shared N-grams, cosine distance between vector representations of sentences, Word Mover’s Distance, distances according to WordNet dictionaries, prediction of two ML models: Siamese neural network based on recurrent and Transformer type - RoBERTa.

Results. An intelligent system for detecting paraphrasing in text based on a classification model and machine learning methods has been developed. The developed system uses the principle of model stacking and feature engineering. Additional features indicate the semantic affiliation of the sentences or the normalized number of common N-grams. An additional fine-tuned RoBERTa neural network (with additional fully connected layers) is less sensitive to pairs of sentences that are not paraphrases of each other. This specificity of the model may contribute to incorrect accusations of plagiarism or incorrect association of user-generated content. Additional features increase both the overall classification accuracy and the model’s sensitivity to pairs of sentences that are not paraphrases of each other.

Conclusions. The created model shows excellent classification results on PAWS test data: precision – 93%, recall – 92%, F1-score – 92%, accuracy – 92%. The results of the study showed that Transformer-type NNs can be successfully applied to detect paraphrasing in a pair of texts with fairly high accuracy without the need for additional feature generation.

KEYWORDS: natural language processing, NLP, rewrite identification, detection of paraphrasing in text, supervised machine learning, deep learning, text classification, text analysis, word embeddings, WordNet, semantic similarity.

REFERENCES

1. Salton G., Wong A., Yang C.-S. A vector space model for automatic indexing, *Communications of the ACM*, 1975, Vol. 18(11), pp. 613–620. DOI: 10.1145/361219.361220
2. Turney P. D., Pantel P. From Frequency to Meaning: Vector Space Models of Semantics, *Journal of Artificial Intelligence Research*, 2010, Vol. 37(1), pp. 141–188. DOI: 10.1613/jair.2934
3. Mikolov T., Chen K., Corrado G. s, Dean J. Efficient Estimation of Word Representations in Vector Space, *ArXiv*, 2013. DOI: 10.48550/arXiv.1301.3781
4. Do Online Plagiarism Checkers Identify Paraphrased Content? *DotNek Software Development*, 2021, <https://www.dotnek.com/Blog/Marketing/do-online-plagiarism-checkers-identify-paraph>
5. Miller G., Beckwith R., Fellbaum C., Gross D., Miller K. Introduction to WordNet: An On-line Lexical Database, *International Journal of Lexicography*, 1990, Vol. 3(4), pp. 235–244. DOI: 10.1093/ijl/3.4.235
6. Corley C., Mihalcea R. Measuring the Semantic Similarity of Texts, *ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, Ann Arbor, Michigan, Jun. 2005, *proceedings*, pp. 13–18.
7. Leacock C., Chodorow M. Combining Local Context and WordNet Similarity for Word Sense Identification, *WordNet: An Electronic Lexical Database*, 1998, Vol. 49(2), pp. 265–283. DOI: 10.7551/mitpress/7287.003.0018
8. Lesk M. E. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone, *SIGDOC '86: the 5th Annual International Conference on Systems documentation*. Toronto, Ontario, Canada, June 1986, *proceedings*, pp. 24–26. DOI: 10.1145/318723.318728
9. Wu Z., Palmer M. Verbs Semantics and Lexical Selection, *ACL '94: the 32nd annual meeting on Association for Computational Linguistics*. Las Cruces, New Mexico, June 27–30, 1994, *proceedings*, pp. 133–138. DOI: 10.3115/981732.981751.
10. Resnik P. Using Information Content to Evaluate Semantic Similarity in a Taxonomy, *ArXiv*, 1995. DOI: 10.48550/arXiv.cmp-lg/9511007
11. Lin D. An Information-Theoretic Definition of Similarity, *ICML*, 1998, <https://www.cse.iitb.ac.in/~cs626-449/Papers/WordSimilarity/3.pdf>
12. Jiang J. J., Conrath D. W. Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy, *10th Research on Computational Linguistics International Conference*. Taipei, Taiwan, Aug. 1997, *proceedings*, pp. 19–33.
13. Mihalcea R. Corley C., Strapparava C. Corpus-based and Knowledge-based Measures of Text Semantic Similarity, *AAAI'06: Proceedings of the 21st national conference on Artificial intelligence*, Boston, Massachusetts, July 2006 : *proceedings*, Vol. 1, pp. 775–780.
14. Hassan S., Mihalcea R. Semantic Relatedness Using Salient Semantic Analysis, *AAAI 2011: Twenty-Fifth AAAI Conference on Artificial Intelligence*, San Francisco, California, USA, August 7–11, 2011, *proceedings*, <https://web.eecs.umich.edu/~mihalcea/papers/hassan.aaai11.pdf>
15. Fernando S., Stevenson M. A Semantic Similarity Approach to Paraphrase Detection, *11th Annual Research Colloquium of the UK Special Interest Group for Computational Linguistics*, May 2008, *proceedings*, pp. 45–52.
16. Milajevs D., Kartsaklis D., Sadrzadeh M., Purver M. Evaluating Neural Word Representations in Tensor-Based Compositional Settings, *ArXiv*, 2014. DOI: 10.48550/arXiv.1408.6179
17. Islam A., Inkpen D. Semantic similarity of short texts, *Current Issues in Linguistic Theory: Recent Advances in Natural Language Processing*, 2009, Vol. 309, pp. 227–236. DOI: 10.1075/cilt.309.18isl
18. Chong M., Specia L., Mitkov R. Using Natural Language Processing for Automatic Detection of Plagiarism, *IPC2010, 4th International Plagiarism Conference, Newcastle-upon-Tyne, May 2010, proceedings*, https://www.academia.edu/326444/Using_Natural_Language_Processing_for_Automatic_Detection_of_Plagiarism
19. Šarić F., Glavaš G., Karan M., Šnajder J., Dalbello Bašić B. TakeLab: Systems for Measuring Semantic Text Similarity, **SEM 2012: The First Joint Conference on Lexical and Computational Semantics*, Volume 1, Proceedings of the main conference and the shared task, and Volume 2, Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012). Montréal, Canada, May 2012, *proceedings*, pp. 441–448.
20. Agirre E., Cer D., Diab M., Gonzalez-Agirre A. SemEval-2012 Task 6: A Pilot on Semantic Textual Similarity, **SEM 2012: The First Joint Conference on Lexical and Computational Semantics*, Volume 1, Proceedings of the main conference and the shared task, and Volume 2, Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012). Montréal, Canada, May 2012, *proceedings*, pp. 385–393.
21. Kong L., Lu Z., Qi H., Han Z. Detecting High Obfuscation Plagiarism: Exploring Multi-Features Fusion via Machine Learning, *International Journal of u- and e- Service, Science and Technology*, 2014, Vol. 7, pp. 385–396. DOI: 10.14257/ijunesst.2014.7.4.35
22. Yin W., Schütz H. Convolutional Neural Network for Paraphrase Identification, *North American Chapter of the Association for Computational Linguistics: Human Language Technologies Conferences*. Denver, Colorado, May 2015, *proceedings*, pp. 901–911. DOI: 10.3115/v1/N15-1091.
23. Qiu L., Kan M.-Y., Chua T.-S. Paraphrase Recognition via Dissimilarity Significance Classification, *Empirical Methods in Natural Language Processing Conference*. Sydney, Australia, Jul. 2006, *proceedings*, pp. 18–26.
24. Kozareva Z., Montoyo A. Paraphrase Identification on the Basis of Supervised Machine Learning Techniques, *Advances in Natural Language Processing. FinTAL 2006. Lecture Notes in Computer Science*, 2006, Vol. 4139, pp. 524–533. DOI: 10.1007/11816508_52
25. Finch A., Sumita E. Using machine translation evaluation techniques to determine sentence-level semantic equivalence, *IWP2005, 3rd International Workshop on Paraphrasing, May 2005, proceedings*, pp. 17–24.
26. Madnani N., Tetreault J., Chodorow M. Re-examining Machine Translation Metrics for Paraphrase Identification, *North American Chapter of the Association for Computational Linguistics, Human Language Technologies Conference of the, Montréal*. Canada, June 2012, *proceedings*, pp. 182–190.
27. Agarwal B., Ramampiaro H., Langseth H., Ruocco M. A Deep Network Model for Paraphrase Detection in Short Text Messages, *ArXiv*, 2017. DOI: 10.48550/arXiv.1712.02820

28. Socher R., Huang E. H.-C., Pennington J., Ng A., Manning C. D. Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection, *NIPS'11: the 24th International Conference on Neural Information Processing Systems*, Granada Spain, December 2011, proceedings, pp. 801–809.
29. Thyagarajan A. Siamese Recurrent Architectures for Learning Sentence Similarity, *The Thirtieth AAAI Conference on Artificial Intelligence*, Phoenix, Arizona, USA, February 12–17, 2016, proceedings, Vol. 30(1), pp. 2786–2792. DOI: 10.1609/aaai.v30i1.10350
30. Neculoiu P., Versteegh M., Rotaru M. Learning Text Similarity with Siamese Recurrent Networks, *Workshop on Representation Learning for NLP*. Berlin, Germany, August 2016, proceedings, pp. 148–157. DOI: 10.18653/v1/W16-1617
31. Ranasinghe T., Orasan C., Mitkov R. Semantic Textual Similarity with Siamese Neural Networks, *RANLP 2019, International Conference on Recent Advances in Natural Language Processing*. Varna, Bulgaria, Sep. 2019, proceedings, pp. 1004–1011. DOI: 10.26615/978-954-452-056-4_116
32. Mahmoud A., Zrigui M. BLSTM-API: Bi-LSTM Recurrent Neural Network-Based Approach for Arabic Paraphrase Identification, *Arabian Journal for Science and Engineering*, 2021, Vol. 46, pp. 4163–4174. DOI: 10.1007/s13369-020-05320-w
33. Reddy D., Kumar M., Kp S. LSTM Based Paraphrase Identification Using Combined Word Embedding Features, *Computing and Signal Processing. Advances in Intelligent Systems and Computing*, 2019, Vol. 898, pp. 385–394. DOI: 10.1007/978-981-13-3393-4_40
34. Li Z., Jiang X., Shang L., Li H. Paraphrase Generation with Deep Reinforcement Learning, *ArXiv*, 2017. DOI: 10.48550/arXiv.1711.00279
35. Goma W., Fahmy A. SimAll: A flexible tool for text similarity, *ESOLEC' 2017, The Seventeenth Conference on Language Engineering, December 2017*, proceedings. https://www.academia.edu/35381793/SimAll_A_flexible_tool_for_text_similarity
36. Ahmed M., Samee M. R., Mercer R. E. Improving Tree-LSTM with Tree Attention, *ArXiv*, 2019. DOI: 10.48550/arXiv.1901.00066
37. Pontes E. L., Huet S., Linhares A. C., Torres-Moreno J.-M. Predicting the Semantic Textual Similarity with Siamese CNN and LSTM, *Actes de la Conférence TALN*. Rennes, France, May 2018, proceedings, pp. 311–320.
38. Wahle J. P., Ruas T., Meuschke N., Gipp B. Are Neural Language Models Good Plagiarists? A Benchmark for Neural Paraphrase Detection, *ArXiv*, 2021. DOI: 10.48550/arXiv.2103.12450
39. Vaswani A., Shazeer N., Parmar N., J. Uszkoreit, Jones L., Gomez A. N., Kaiser Ł., Polosukhin I. Attention Is All You Need, *ArXiv*, 2017. DOI: 10.48550/arXiv.1706.0376
40. Nighojkar A., Licato J. Improving Paraphrase Detection with the Adversarial Paraphrasing Task, *ArXiv*, 2021. DOI: 10.48550/arXiv.2106.07691
41. Arase Y., Tsujii J. Transfer fine-tuning of BERT with phrasal paraphrases, *ArXiv*, 2021. DOI: 10.48550/arXiv.1909.00931
42. Devlin J., Chang M.-W., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, *ArXiv*, 2018. DOI: 10.48550/arXiv.1810.04805
43. Kusner M. J., Sun Y., Kolkin N. I., Weinberger K. Q. From Word Embeddings to Document Distances, *JMLR: W&CP*, 2015, Vol. 37, pp. 957–966.
44. Zhang Y., Baldrige J., He L. PAWS: Paraphrase Adversaries from Word Scrambling, *ArXiv*, 2019. DOI: 10.48550/arXiv.1904.01130
45. Oliinyk V.-A., Vysotska V., Burov Y., Mykich K., Fernandes V. B. Propaganda Detection in Text Data Based on NLP and Machine Learning, *Modern Machine Learning Technologies and Data Science (MoMLeT+DS 2020), Workshop, Lviv-Shatsk, 2–3 June 2020, CEUR workshop proceedings*. Aachen, CEUR-WS.org, 2020, Vol. 2631, pp. 132–144.
46. Liu Y., Ott M., Goyal N., Du J., Joshi M., Chen D., Levy O., Lewis M., Zettlemoyer L., Stoyanov V. RoBERTa: A Robustly Optimized BERT Pretraining Approach, *ArXiv*, 2019. DOI: 10.48550/arXiv.1907.11692

УПРАВЛІННЯ У ТЕХНІЧНИХ СИСТЕМАХ

CONTROL IN TECHNICAL SYSTEMS

УПРАВЛЕНИЕ В ТЕХНИЧЕСКИХ СИСТЕМАХ

UDC 004.94

RISK ASSESSMENT MODELING OF ERP-SYSTEMS

Kozhukhivskiyi A. D. – Dr. Sc., Professor, Professor Department of Information and Cybernetic security of State University of Telecommunications, Kyiv, Ukraine.

Kozhukhivska O. A. – Dr. Sc., Associate Professor Department of Information and Cybernetic security of State University of Telecommunications, Kyiv, Ukraine.

ABSTRACT

Context. Because assessing security risks is a complex and complete uncertainty process, and uncertainties are a major factor influencing valuation performance, it is advisable to use fuzzy methods and models that are adaptive to noncomputed data. The formation of vague assessments of risk factors is subjective, and risk assessment depends on the practical results obtained in the process of processing the risks of threats that have already arisen during the functioning of the organization and experience of security professionals. Therefore, it will be advisable to use models that can adequately assess fuzzy factors and have the ability to adjust their impact on risk assessment. The greatest performance indicators for solving such problems are neuro-fuzzy models that combine methods of fuzzy logic and artificial neural networks and systems, i.e. “human-like” style of considerations of fuzzy systems with training and simulation of mental phenomena of neural networks. To build a model for calculating the risk assessment of security, it is proposed to use a fuzzy product model. Fuzzy product models (Rule-Based Fuzzy Models/Systems) this is a common type of fuzzy models used to describe, analyze and simulate complex systems and processes that are poorly formalized.

Objective. Development of a fuzzy model of quality of security risk assessment and protection of ERP systems through the use of fuzzy neural models.

Method. To build a model for calculating the risk assessment of security, it is proposed to use a fuzzy product model. Fuzzy product models are a common kind of fuzzy models used to describe, analyze and model complex systems and processes that are poorly formalized.

Results. Identified factors influencing risk assessment suggest the use of linguistic variables to describe them and use fuzzy variables to assess their qualities, as well as a system of qualitative assessments. The choice of parameters was substantiated and a fuzzy product model of risk assessment and a database of rules of fuzzy logical conclusion using the MATLAB application package and the Fuzzy Logic Toolbox extension package was implemented, as well as improved by introducing the adaptability of the model to experimental data by introducing neuro-fuzzy components into the model. The use of fuzzy models to solve the problems of security risk assessment, as well as the concept and construction of ERP systems and the analyzed problems of their security and vulnerabilities are considered.

Conclusions. A fuzzy model has been developed risk assessment of the ERP system. Selected a list of factors affecting the risk of security. Methods of risk assessment of information resources and ERP-systems in general, assessment of financial losses from the implementation of threats, determination of the type of risk according to its assessment for the formation of recommendations on their processing in order to maintain the level of protection of the ERP-system are proposed. The list of linguistic variables of the model is defined. The structure of the database of fuzzy product rules – MISO-structure is chosen. The structure of the fuzzy model was built. Fuzzy variable models have been identified.

KEYWORDS: Security, fuzzy logic, fuzzy product model, risk assessment, security, ERP-system.

ABBREVIATIONS

ANFIS is a Adaptive Network-based Fuzzy Inference System;

DB is a Database;

DSTU is a State standard of Ukraine;

ERP is an Enterprise Resources Planning;

ERP-System is an Enterprise Recourses Planning System;

MISO is a Structure (Multi Inputs – Single Output);

FIS is a Fuzzy Inference System;

ARL is an acceptable risk level;

MRL is a middle risk level;

HRL is a high-risk level;

VLR is a very low risk;

LR is a low risk;

AR is an average risk;

HR is a High risk;
VHR is a Very high risk;
CVSS is a Common Vulnerability Scoring System;
NVD is a National Vulnerability Database;
CVE is a Common Vulnerabilities and Exposures.

NOMENCLATURE

R_{ij} is a risk of the i -th resource in the implementation of the j -th threat;

A_{ij} is an expected loss from the one-time implementation of the j -th threat to for the i -th resource;

P_j^t is a probability of occurrence of j -th threat;

P_{ij}^v is a vulnerability of the i -resource to the j -th threat;

IR is a resource set of system;

Th is a set of threats to the system.

A_i^V is a value of the i -st resource;

F_{ij}^e is an impact consequences in the implementation of the j -th threat on the i -th resource, or the propensity of the i -th resource to the j -th threat;

R_i is a risk of the i -th resource in the implementation of threats;

R_{ik} is a risk of the i -th resource in the implementation of the k -th threat;

Th_i is a set of risks for the i -resource;

R_g is a general system risk;

R_{ig} is a risk of the i -th resource at general system risk;

FL_i is a financial loss of the i -th resource;

R_i is a risk of the i -st resource;

Co_i is a cost of the i -th resource;

FL is a total financial loss;

RL is a risk level type;

\min_R is a minimum value of risk assessment;

\max_R is a maximum value of risk assessment;

Pr_1 is a parameter, maximum value of risk assessment of acceptable type;

Pr_2 is a parameter, the maximum value of the risk assessment of the average type;

x_I is an incoming Variables (can be either clear or fuzzy);

X_I is a definition area appropriate prerequisites;

y is a fuzzy output variable;

Y is a definition area the conclusion;

A_{ij}, B_i are fuzzy sets defined that are defined by X_j and Y with affiliation functions $\mu_{A_{ij}}(x_j) \in [0;1]$ and $\mu_{B_i}(y) \in [0;1]$ respectively;

p_i, q_i, r_i are affiliation functions options;

k is an example from many examples of training sampling;

$x_m^{(k)}$ are input variable values x_m ;

$y^{(k)}$ is a reference value of the source variable y in the k -th example;

K is a total number of examples, size of Training sample;

$E^{(k)}$ is an error k -th example from many examples of educational sample;

E is an error;

$y^{1(k)}$ is an installed the value of the source variable y in the k -th example;

ε is an installed threshold;

$\mu(x, \sigma, c)$ is a bell function – Gauss distribution function;

x is a degree of belonging to the term;

σ is a standard deviation, function steepness;

c is a shift peak bell Curve from Zero;

l_m are number of functions belonging to specify variables X_1, X_2, X_3, X_4 ;

l_y is a number of affiliation functions for the source variable Y .

INTRODUCTION

The basis of activity of any organization is business processes, which are determined by the goals and objectives of the entity. The business process broadly understands the structured sequence of actions to perform a certain type of activity at all stages of the life cycle of the subject of activity. Each business process has a start (login), output, and sequence of procedures that ensure that operations are grouped by the appropriate types. In general, the calculation of the risks of security of ERP systems should be carried out in relation to each critical business process and only on those vulnerabilities that are relevant to a particular business process, and it should be borne in mind that a number of vulnerabilities may be the same for all business processes.

Each vulnerability in the current list of vulnerabilities is correlated with the threat that this vulnerability may be, and for each pair, the probability of its occurrence is assessed and the impact of the pair's implementation on integrity, confidentiality, accessibility, and observability is assessed.

We will use the following definitions. Probability is a conditional number that determines the frequency of such a threat / vulnerability of a pair. Privacy is a property of information that is that information cannot be obtained by an unauthorized user and/or process. Integrity is a property of information, which is that information cannot be modified by an unauthorized user and/or process. System integrity – system property, which is that none of its components can be eliminated, modified or added in violation of security policy. Accessibility – the property of the system resource, which is that the user and/or process,

which has the appropriate powers, can use the resource in accordance with the rules established by the security policy, without waiting longer for a specified (small) period of time, that is, when it is in the form required by the user, in the place required by the user, and at the time when it is necessary. Observation – system property, which allows to record the activities of users and processes, the use of passive objects, as well as to unequivocally establish identifiers of users involved in certain events and processes in order to prevent violations of security policies and/or to ensure liability actions.

The object of the study is the modeling of a fuzzy model of the ERP system.

The subject of the study is neuro-fuzzy models that combine methods of fuzzy logic and artificial neural networks and systems.

The purpose of the work is to improve the quality of assessment of security risks and protection of ERP systems through the use of fuzzy neural models.

1 PROBLEM STATEMENT

Security risk modeling is an important element of the overall security risk management process, which is the process of ensuring that the organization's position is within acceptable limits defined by senior management and consists of four main stages: security risk assessment, testing and supervision, mitigation, and operational security [1].

Risk managers and organizers use risk assessment to determine which risks to reduce through control and which to accept or transfer. Modeling of information security risks is a process of identifying vulnerable situations, threats, the likelihood of their occurrence, the level of risks and consequences associated with the assets of organization, as well as control, which can mitigate threats and their consequences. Modeling includes: assessing the likelihood of threats and vulnerabilities that are possible; calculation of the impact that can be a threat to each asset; determination of quantitative (measurable) or qualitative (described) cost of risk.

The full process of risk assessment modeling should also include recommendations for control and evaluation of results.

Information risk assessment can be carried out by modeling. The methodology for modeling information security risk assessment understands the systematized sequence of actions (step-by-step instructions) that need to be implemented and the tool (software product) for risk assessment at the enterprise.

Also, to assess security risks, manager documents containing theoretical descriptions can be used and provide guidelines on the risk assessment process, but no specific technologies for their implementation are provided. At present, the following standards apply on the territory of Ukraine: ISO 27001, ISO 27002, ISO 27003, ISO 27004 and ISO 27005 [2–6].

Recently, methods of analysis and risk assessment, based on elements of fuzzy logic, have been intensively developed. Such methods allow you to change the test table of methods of rough risk assessment to the mathematical

method, as well as significantly expand the possibilities of risk modeling [7–11].

To build a risk assessment model, we will use the ratio of risk factors, according to the formulas [11]:

$$R_{ij} = A_{ij} \cdot P_j^t \cdot P_{ij}^v, i \in IR, j \in Th. \quad (1)$$

$$A_{ij} = A_i^V \cdot F_{ij}^e, i \in IR, j \in Th. \quad (2)$$

The general ratio of risks assessment factors (1) and (2) is represented by the expression:

$$R_{ij} = A_i^V \cdot F_{ij}^e \cdot P_j^t \cdot P_{ij}^v, i \in IR, j \in Th. \quad (3)$$

As for each information resource many risks (from one to all) can be defined, the estimation of the general risk on an information resource will be defined as the maximum estimation among risks:

$$R_i = \max(R_{ik}), k \in Th_i. \quad (4)$$

In turn, the system-wide risk assessment will be defined as the maximum assessment among resource risk assessments:

$$R = \max(R_i), i \in IR. \quad (5)$$

Total financial loss is defined as the sum of financial losses on all resources:

$$FL = \sum_i FL_i, i \in IR. \quad (6)$$

Thus, the overall risk assessment of the ERP system can be expressed as follows:

$$Y = f_Y(X_1, X_2, X_3, X_4). \quad (7)$$

Based on the analysis and the formed ratio of risk factors (3), a fuzzy model with four input parameters (X1, X2, X3, X4) and one output Y (MISO structure [11]) is proposed to assess each of the risks. The number of input parameters is selected according to the number of factors influencing the degree of risk (3).

Important processes are the implementation of the model using the MATLAB application package and the Fuzzy Logic Toolbox extension package, as well as improvements by introducing the adaptability of the model to experimental data by introducing neuro-fuzzy components into the model.

As a result of modeling the process of obtaining risk assessments of the ERP system and analyzing the results, a fairly high accuracy and low error of the developed model were established.

The proposed model and approach to assessing the security risks of the ERP system may be further developed and underlie the development of an information risk management system.

2 REVIEW OF THE LITERATURE

The security risk analysis study begins in the mid-1980s, and in the early 90s R. Baskerville identified risk analysis checklists for tools used to design information system security measures [12]. Over time, complex tools

are developed to analyze risks, such as: Facilitated Risk Assessment Process [13]; The Operationally Critical Threat, Asset, and Vulnerability Evaluation [14]; CO-RAS [15]; Method of Risk analysis of business model [16]; Security Risk Analysis Method [17]; Risk Watch method [18]; Consultative Objective and Bifunctional Risk Analysis [19]; CRAMM [20].

In addition, since the early 2000s, some other security risk modeling techniques have also been used in the risk forecasting industry, which have provided good performance and are commonly referred to as “soft computing models”, including gray relational approach, fuzzy number arithmetic, information entropy, fuzzy weighted average approach, fuzzy measure and theory of evidence, method of fuzzy analysis of the hierarchical process.

The development and application of soft computing and hybrid models are considered to be modern areas of research to assess security risks.

Soft computing components include: Neural networks – computational systems that assess the risks of security through similar functioning of biological neural networks and learning tasks (gradually improving performance of these networks), considering examples, in general, without special programming for the task; Rough sets an effective mathematical analysis tool to address uncertainty in the field of solution analysis; Grey sets; Fuzzy systems – based on the algorithm for obtaining fuzzy conclusions based on fuzzy preconditions; Generic algorithms – belong to the largest class Evolutionary algorithms and generate solutions to optimization problems using methods borrowed from the theory of evolution, such as inheritance, mutation, selection and crossover; Method of reference vectors – the data analysis method for classification and regression analysis using managed learning models is used when input is either not defined or when only some data is determined by their preprocessing; Bayesian network – used to identify cause-and-effect relationships of risk factors and predict the likelihood of security risk.

Hybrid models represent a combination of two or more technologies to develop robust risk assessment information systems. The most common hybrid model is the neuro-fuzzy network.

To determine the level of risk, it is advisable to use the apparatus of the theory of fuzzy sets, which allows you to describe vague concepts and knowledge, operate them and draw vague conclusions. The theory of fuzzy sets is used precisely to solve problems in which inputs are unreliable and poorly formalized, as in the case of the problem solved in this work. To assess the risk, it is appropriate to use the mechanism of a vague logical conclusion – obtaining a conclusion in the form of a fuzzy set corresponding to the current values of input variables, using a fuzzy knowledge base and fuzzy operations.

Most often, Mamdani and Sugeno algorithms are used in practice. The main difference between them is the method of create the value of the source variable in the rules that make up the knowledge base. In systems like Mam-

dani, the values of input variables are set by fuzzy terms, in systems like Sugeno – as a linear combination of input variables. For tasks in which identification is important, models of fuzzy conclusion Mamdani, Sugeno, Larsen, Tsukamoto [20] have been developed. Most often, Mamdani and Sugeno algorithms are used in practice. The main difference between them is the method of applying the value of the source variable in the rules that make up the knowledge base. In systems like Mamdani, the values of input variables are set by fuzzy terms, in systems like Sugeno – as a linear combination of input variables. For tasks in which identification is more important, it is advisable to use the Sugeno algorithm, and for tasks in which the explanation and justification of the decision is more important, Mamdani’s algorithm will have an advantage.

3 MATERIALS AND METHODS

To build a structure a model for calculating security risk assessment, it is proposed to use Rule-Based Fuzzy Models / Systems.

Under the Rule-Based Fuzzy Models / Systems understand the agreed a lot of individual fuzzy product rules of the type “if A, then B” where A is the prerequisite (parcel, antecedent) of a certain rule, and B – the conclusion (action, consequent) of the rule in the form of fuzzy statements. The model is designed to determine the degree of truthfulness of the conclusions of fuzzy product rules. The degree of truth is determined on the basis of preconditions with a certain degree of truthfulness of the relevant rules.

When building a fuzzy product model should take into account: the method of fuzzy inference; fuzzy product rules database; the order of introduction of fuzzy cations; the procedure for aggregating the degree of truth of the preconditions for each of the rules of fuzzy product; activation procedure for each of the rules of the odd product; the procedure for eliminating activated inclusions of all fuzzy product rules for each source variable; diffusion procedure for clarity of each aggregate output variable; procedure for optimizing the parameters of the final base of fuzzy rules.

At present, many different types of fuzzy product models are offered on the basis of different combinations of these components.

Rule-Based Fuzzy Models / Systems are used in solving a number of problems in which information about the system, its parameters, as well as the inputs, outputs and states of the system is unreliable and poorly formalized. Together with the advantages of describing the model in a language close to natural, in the versatility and efficiency of the model, Rule-Based Fuzzy Models / Systems are characterized by certain disadvantages: the wording of the original set of fuzzy rules is carried out with the help of an expert, so it may be incomplete or contradictory; the choice of the type and parameters of the functions of belonging in fuzzy statements of the rules is subjective; automatic acquisition of knowledge cannot be performed.

To eliminate these shortcomings, it is proposed to use an adaptive fuzzy production model, which in the process and on the results of functioning corrects both the composition of the rules in the base and the parameters of the functions of belonging, as well as to implement various components of this model on the basis of neuronet technology.

Determine the incoming and outgoing parameters of the model.

To build a risk assessment calculation model, we will use the risk factor ratio according to formula (1).

Under the expected damage from a one-time implementation of the threat we understand the cost (or value) of the asset, which is mathematically expressed as follows (see (2)).

Taking into account (1) and (2), we obtain the general ratio of factors for risk assessment (see (3)).

Since many risks can be identified for each information resource (one to all), the assessment of the total risk by the information resource will be defined as the maximum risk assessment of the resource (see (4)).

In turn, the assessment of system risk will be defined as the maximum assessment among resource risk assessments (see 5)).

In turn, the total financial loss will be determined as the amount of financial losses on all resources (see 6)).

We will apply a linguistic approach to the description of security risk factors. Suppose as the values of factors and characteristics of relations between them not only quantitative assessment, but also qualitative, sentences of natural language. Then this approach will provide a quantitative description of the elements of the model in the conditions of vague information about the value of the risk level, the cost of the resource, the impact of the consequence of, the likelihood of a threat, the vulnerability, of resource protection and ways to avoid negative impact from the implementation of risk.

Each risk factor of security and the risk itself be described by linguistic variables $X \in \bar{X}$. The value of described by linguistic variables of the model \bar{X} is: $\bar{X} = \{\text{“Resource Price”}, \text{“Impact of the consequence”}, \text{“Probability the emergence of Threat”}, \text{“Resource Vulnerability”}, \text{“Risk”}\}$.

Thus, information security risk assessment can be expressed as (see 7)).

Based on the analysis [21] and the formed ratio of risk factors (3) for the assessment of each of the risks, a fuzzy model with four input parameters (X_1, X_2, X_3, X_4) and one Y output (MISO structure [22]) is proposed. The number of input parameters is selected according to the number of factors influencing the degree of risk (3).

To maintain the level of security of the ERP system, it is necessary to determine what risks, according to the level of their assessment – risk level (RL), require processing according to certain recommendations. To do this, we will introduce 3 types of risk levels:

- acceptable risk – ARL – will be considered insignificant, the processing of such a risk is not required;
- medium risk – MRL – recommended for processing in order to minimize it;
- high risk-HRL– we will consider it essential and its processing is mandatory.

Determination of the type of risk will be carried out as follows:

$$RL = \begin{cases} ARL, R_{ij} \in (\min_R; Pr_1); \\ MRL, R_{ig} \in (Pr_1; Pr_2); \quad i \in IR, j \in Th, \\ HRL, R_{ij} \in (Pr_2; \max_R). \end{cases} \quad (8)$$

Parameters – the maximum value of the assessment of acceptable and medium risk – $[Pr_1]$ and $[Pr_2]$ respectively – are set by experts.

We will create a structure and build bases of fuzzy product rules.

The structure of the rules should correspond to the structure of the model, namely the number of fuzzy statements in the prerequisites and conclusions. The database of rules that has the structure of MISO, in general, has the following rule structure [22]:

$$P_i: \text{If } x_1 \text{ is } A_{i1} \text{ and } \dots \text{ and } x_j \text{ is } A_{ij} \text{ and } \dots \text{ and } x_m \text{ is } A_{im}, \text{ then } y \text{ is } B_i. \quad (9)$$

When creating a fuzzy model, both apriori data coming from experts and data obtained as result of measurements can be used.

In the first case, if there is no need to agree on the opinions of experts, it is assumed that the tasks of ensuring completeness and inconsistency of the database of fuzzy rules are solved in advance. If only experimental data are known, these tasks can be attributed to the tasks of system identification. In practice, there may also be a mixed case when the initial database of fuzzy rules is built on the basis of heuristic assumptions, and its clarification is carried out using experimental data.

ANFIS, the adaptive network fuzzy output system proposed by Chang in 1992, will be used to represent the fuzzy production model and algorithm of fuzzy output in the form of a fuzzy network [23].

Since the fuzzy ANFIS network is presented multilayer structure with a direct signal propagation, and the value of the source variable can be changed by adjusting the parameters of layer elements, then to teach this network you can use an algorithm for reverse spreading the error, which belongs to the class of classic gradient algorithms.

Consider the problem of fuzzy neural network of anfis type, which implements the algorithm of fuzzy output of Takagi-Sugeno [24].

Let the rules of this form be given:

$$\begin{aligned} P_1: \text{If } x_1 \text{ is } A_{11} \text{ and } x_2 \text{ is } A_{12} \text{ then} \\ y_1 = a_1 x_1 + b_1 x_2; \\ P_2: \text{If } x_1 \text{ is } A_{21} \text{ and } x_2 \text{ is } A_{22} \text{ then} \\ y_2 = a_2 x_1 + b_2 x_2. \end{aligned} \quad (10)$$

Let's define a linguistic variable Y “Risk”. To evaluate the linguistic variable Y , we will use the term set $T(Y)$ of five quality terms: $T(Y) = \{\text{«Very low risk (VLR)»}, \text{«Low risk (LR)»}, \text{«Medium risk (MR)»}, \text{«High risk (HR)»}, \text{«Very high (VHR)»}\}$.

risk (VHR)». Definition Areas of E_Y of the linguistic variable Y will be set at the interval [0; 100] [25].

The value of information will be defined as the relationship between the type of confidentiality and criticality – criticality (C) of the information. Value estimation is formed as the sum of points corresponding to each type and level of criticality of information. Estimates of the value of information are given in Table 1.

The criticality of the information will be determined, taking into account the assessment of the consequences of violation of the properties of information. To evaluate the linguistic variable X_1 “Resource price”, we will use the term set $T(X_1)$ of three high – quality therms: Basis of the development of information risk management systems. $T(X_1) = \{ \text{Low Price (LP); Average Price (AP);$

Table 1 – Definition of value assessment of information

Type of information	Criticality of information (C)		
	Insignificant (1–3 points)	Significant (4–9 points)	Critical (10–15 points)
Open (1 point)	2–4	5–10	11–16
For internal use (2 points)	3–5	6–11	12–17
Confidential (3 points)	4–6	7–12	13–18
Strictly Confidential (4 points)	5–7	8–13	14–19

High Price (HP)». The Definition Area of E_{X_1} of the linguistic variable X_1 be set at the interval [4;19] [26].

The value level assessment scale for each linguistic variable is determined by values 4, 11 and 19, respectively.

To evaluate the linguistic variable X_3 “Threat probability level”, we will use the set $T(X_3)$ of five quality therms: $T(X_3) = \{ \text{Very low probability of threat (VLT); Low probability of threat (LT); Average threat probability (MT); High probability of threat (HT); Very high probability (VHT). Definition Areas } E_{X_3} \text{ of the linguistic variable } X_3 \text{ beset at the interval [0, 05; 365].$

The VLT term corresponds to a situation where the threat is almost never realized or implemented no more than 2–3 times in five years (frequency in the range [0, 0,6]). The term LT corresponds to the situation when the threat occurs 1–2 times a year (frequency in the range [1, 2]). The term MT corresponds to the situation when the threat occurs once every 2–3 months (frequency in the range [4, 6]). The HT term corresponds to the situation when the threat occurs 1–2 times a month (frequency in the range [12, 24]). The VHT term corresponds to a situation where a threat occurs from 1 time per week to 1 ti-me per day (frequency in the range [52, 365]).

When evaluating the linguistic variable X_4 “Resources Vulnerability”, we will rely on the common vulnerability assessment system (CVSS), which makes it possible to fix the basic characteristics of the vulnerability and create a numerical score that reflects its criticality [27]. CVSS is a free and open industry standard for assessing the severity of a computer system security vulnerability, allowing users to prioritize resources according to threat. The CVSS assess-
©Kozhukhivskiy A. D., Kozhukhivska O. A., 2022
DOI 10.15588/1607-3274-2022-4-12

ment system consists of three indicators 26]: basic metric – reflects the main qualities and characteristics of the vulnerability; time indicators – reflects the following characteristics of the vulnerability, which change over time, develop over the vulnerable period; context metrics – displays vulnerability characteristics that are unique to the user environment. Each group of indicators has a certain numerical score in the range from 0 to 10 and a dot representing the value of all indicators in the form of a block of text.

To obtain vulnerability indicators, we will use the National Vulnerability Assessment System (NVD) [28]. NVD is an information database of the U.S. National Standardization Authority, the National Institute of Standards and Technology, supported by the U.S. Government. In the NVD database, the security level values of the vulnerability are calculated by values from 0 to 10 (according to CVSS) and are described linguistically by the term None, Low, Medium, High, and Critical [28].

According to the linguistic therms of the NVD data-base, we will use the $T(X_4)$ term set of four quality the-rms to evaluate the linguistic variable X_4 “Resource Vulnerability”: $T(X_4) = \{ \text{Low vulnerability (LV); Medium vunerability (MV); High vulnerabilidad (HV); Critical vulnerability (CV). Definition Area } E_{X_4} \text{ of the linguistic variables } X_4 \text{ set at the interval [0, 10].$

Table 2 describes NVD vulnerability scores by points and linguistically [29], description of the impact of exploitation, and corresponding levels of resource vulnerability according to the term sets $T(X_4)$.

Table 2 – Resource Vulnerability Rating Scalt

Level by NVD	Score by NVD	Description of the vulnerability level	Vulnerability level
None	0.0	Vulnerability has no effect on resource	
Low	0.1–3.9	A vulnerability that has little impact on the resource does not Affect the availability, integrity and confidentiality of information	LV
Medium	4.0–6.9	A vulnerability that may have some impact on the resource but has a complexity of implementation or does not cause serious consequences. It is possible to access confidential information, change some information, but there is no control over the information, or the scale of losses is small. Resource availability failures occur	MV
Higt	7.0–8.9	A vulnerability that has a significant impact on the resource, possible access to confidential information, changes in informations and control over information. Significant resource availability failures and performance reductions	HV
Critical	9.0–10.0	Vulnerability, the consequence of the exploitation of which has a serious impact on the resource: complete loss of availability and integrity of information, full disclosure of confidential information	CV

4 EXPERIMENTS

To develop a fuzzy model, we will use the Fuzzy Logic Toolbox tool from the MATLAB package version R2020a.

Fuzzy Logic Toolbox is a MATLAB extension package that contains tools for designing fuzzy logic systems. The package allows you to create expert systems based on fuzzy logic, develop clustering with fuzzy algorithms, as well as design fuzzy neural networks. The package includes a graphical interface for interactive step-by-step design of fuzzy systems, command line functions for software development, as well as special blocks for building fuzzy logic systems. All functions of the package are implemented in the open language MATLAB, which allows you to control of the execution of algorithms, change the source code, as well as create your own functions and procedure [30].

In accordance with the developed structure of the fuzzy model (see (7)) using the Fuzzy Logic Designer GUI of the Fuzzy Logic Toolbox package, a fuzzy product model was developed, the structure of which is shown in Fig. 1.

The developed fuzzy model has a MISO structure: four inputs (risk assessment factors) and one output (risk assessment).

Among the fuzzy Logic Toolbox models available, using Mamdani or Sugeno fuzzy conclusion algorithms, the Sugeno model was chosen as the only one that has the ability to use fuzzy natural production networks based on it, namely the ANFIS network.

For each input of the model according to the developed structure (7), the ranges of the areas for determining the numerical value of the parameter, quantity, type, name and parameters of the membership functions were adjusted:

- the range of the input parameter definition area corresponds to the ranges of estimates of the corresponding risk factor;
- number of affiliation functions corresponds to the number of terms of the linguistic variable of the parameter;
- the names of the functions of the affiliation correspond of the abbreviated names of the term;
- the type of the function of belonging is a kolokolobrazna curve – the function of the Gauss distribution:

$$\mu(x, \sigma, c) = e^{-\frac{(x-c)^2}{2\sigma^2}} \quad (11)$$

- parameters of affiliation functions were selected in accordance with the center of values of parameter evaluations by term, parameters σ are selected so that functions of the affiliation overlap at the level of 0.5.

The results of configuring the source and input data using the Membership Function Editor are shown in Figs 2 and 3, respective.

The list of selected parameters for model data and affiliation functions is shown in Table 3.

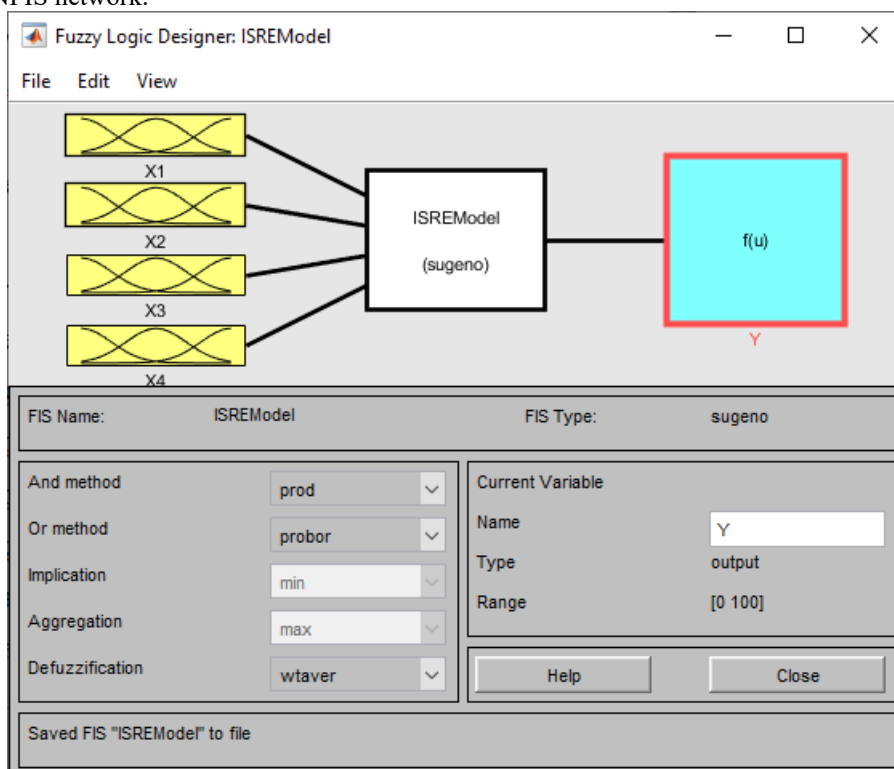


Figure 1 – Structure of fuzzy production model

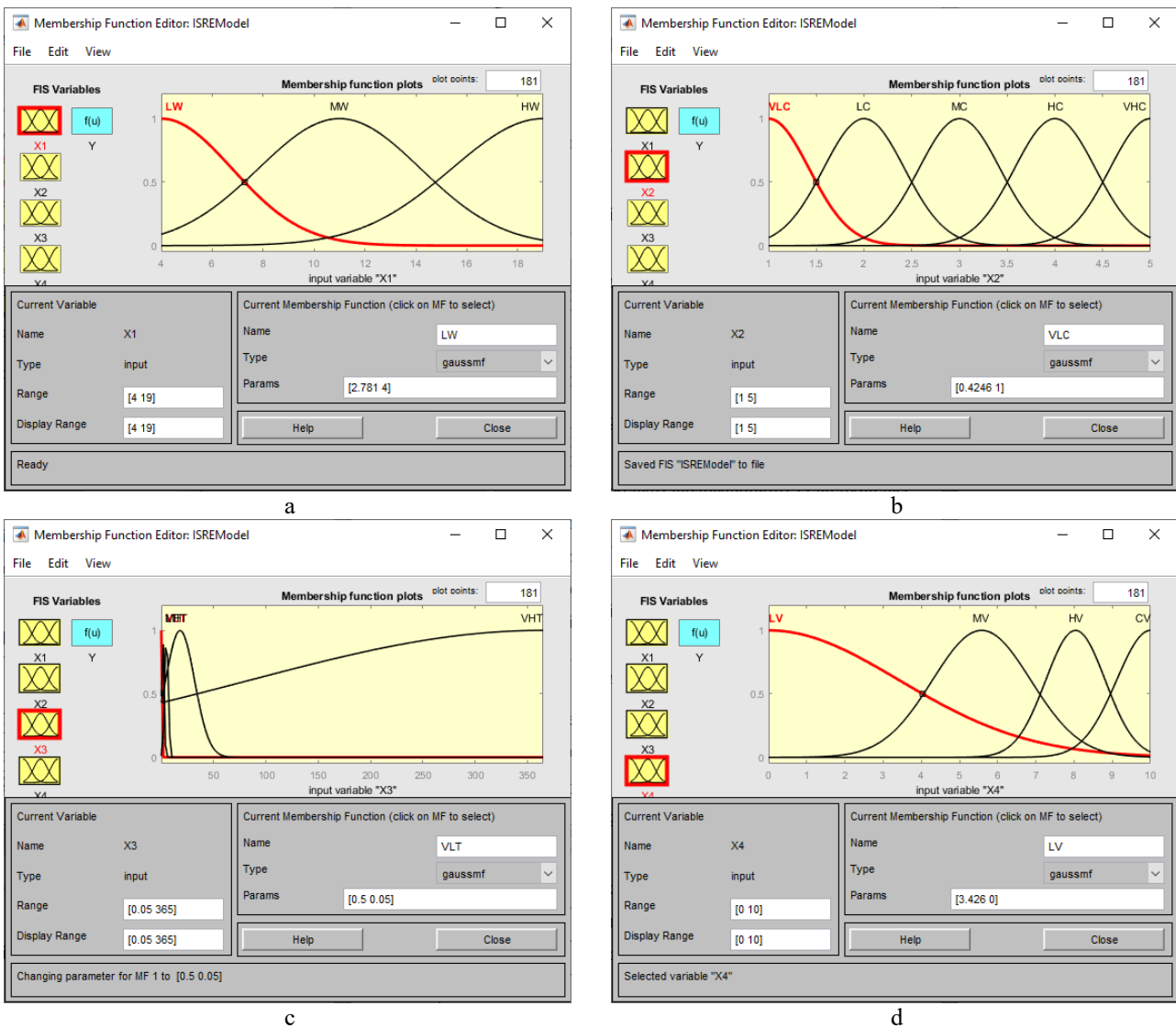


Figure 2 – Model input data configuration results:

a – Input parameter X_1 “Resource Value”, b – Input parameter X_2 “Impact the consequence”, c – Input parameter X_3 “Probability the emergence of threat”, d – Input parameter X_4 “Resource Vulnerability”

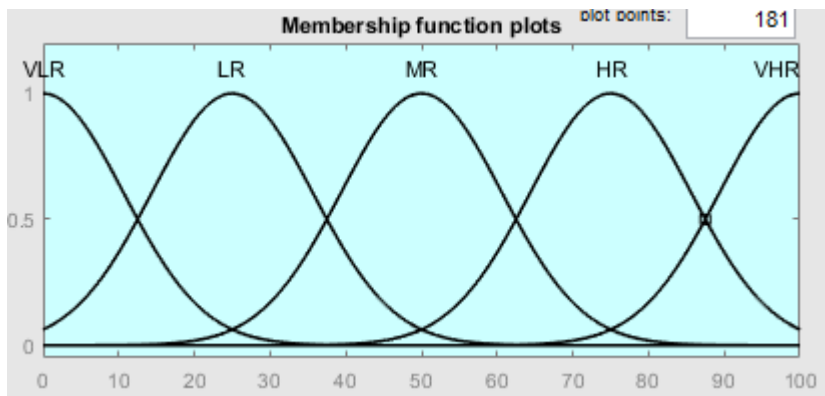


Figure 3 – The results of configuring the original model parameter

Table 3 – Model Data Options

Input Parameter Definition Driver Ratings)	Affiliation function	Therm	Deviation (σ)	Center (peak) (s)
$X_1 \in [4;19]$				
	$\mu_1(X_1)$	LW	2.781	4
	$\mu_2(X_1)$	MV	3.184	11
	$\mu_3(X_1)$	HV	3.589	19
$X_2 \in [1;5]$				
	$\mu_1(X_2)$	VLC	0.4246	1
	$\mu_2(X_2)$	LC	0.4246	2
	$\mu_3(X_2)$	MC	0.4246	3
	$\mu_4(X_2)$	HC	0.4246	4
	$\mu_5(X_2)$	VHC	0.4246	5
$X_3 \in [0.05;365]$				
	$\mu_1(X_3)$	VLT	0.5	0.05
	$\mu_2(X_3)$	LT	1	1.5
	$\mu_3(X_3)$	MT	1.8	5
	$\mu_4(X_3)$	HT	13.86	18
	$\mu_5(X_3)$	VHT	280.8	365
$X_4 \in [0;10]$				
	$\mu_1(X_4)$	LV	3.426	0
	$\mu_2(X_4)$	MV	1.29	5.579
	$\mu_3(X_4)$	HV	0.78	8.037
	$\mu_4(X_4)$	CV	0.8875	10
$Y \in [0;100]$				
	$\mu_1(Y)$	VLR	10.62	0
	$\mu_2(Y)$	LR	10.62	25
	$\mu_3(Y)$	MR	10.62	50
	$\mu_4(Y)$	HR	10.62	75
	$\mu_5(Y)$	VHR	10.62	100

To form the initial base of rules, we will use an approach based on the generation of many rules, based on possible combinations of vague statements in the prerequisites and conclusions of the rules, according to which the maximum number of rules in the database is determined [27]:

$$I = I_1 \cdot \dots \cdot I_m \cdot I_y \quad (12)$$

Thus, for the model being developed, the number of rules in the initial base will be $3 \cdot 5 \cdot 5 \cdot 4 = 300$ rules.

According to the structure of the rules (10), which for the developed model have a general look of:

$$P_i : \text{It } X_1 \text{ is } T_i(X_1) \text{ and } X_2 \text{ is } T_i(X_2) \text{ and } X_3 \text{ is } T_i(X_3) \text{ and } X_4 \text{ is } T_i(X_4), \text{ then } Y \text{ is } T_i(Y) \quad (14)$$

the initial database of rules was formed, consisting of 300 rules, fragment of which of the 10 rules is shown in Figure 4.

Tools are allowed when creating a rule to indicate weight, that is, the significance of the rule, which has a definition area $[0;1]$. In the built database, all rules, by default, have the same weight of 1.

1. If (X1 is LW) and (X2 is VLC) and (X3 is VLT) and (X4 is LV) then (Y is VLR) (1)
2. If (X1 is LW) and (X2 is VLC) and (X3 is VLT) and (X4 is MV) then (Y is VLR) (1)
3. If (X1 is LW) and (X2 is VLC) and (X3 is VLT) and (X4 is HV) then (Y is LR) (1)
4. If (X1 is LW) and (X2 is VLC) and (X3 is VLT) and (X4 is CV) then (Y is LR) (1)
5. If (X1 is LW) and (X2 is VLC) and (X3 is LT) and (X4 is LV) then (Y is VLR) (1)
6. If (X1 is LW) and (X2 is VLC) and (X3 is LT) and (X4 is MV) then (Y is VLR) (1)
7. If (X1 is LW) and (X2 is VLC) and (X3 is LT) and (X4 is HV) then (Y is LR) (1)
8. If (X1 is LW) and (X2 is VLC) and (X3 is LT) and (X4 is CV) then (Y is LR) (1)
9. If (X1 is LW) and (X2 is VLC) and (X3 is MT) and (X4 is LV) then (Y is VLR) (1)
10. If (X1 is LW) and (X2 is VLC) and (X3 is MT) and (X4 is MV) then (Y is VLR) (1)

Figure 4 – Fragment of model product rules base

5 RESULTS

The use of a fuzzy model provides a more flexible processing of inaccurate /substandard factors of security risk and allows you to proceed to the numerical representation of any characteristics. The proposed fuzzy model and methods can be used both to assess specific types of security risks of ERP system resources and to the overall risk of security of the ERP system.

In a real enterprise, the use of a fuzzy model involves the implementation of a certain block of preparatory work such as: identify specific objects of protection of the, ERP-system; make a list of threats and possible vulnerabilities; to make a list of current threat/vulnerability for ERP-systems (taking into account the peculiarities of business processes); assess the probabilities of implementing a threat using the specified vulnerability; to assess the consequences of the threat, the impact of the threat on the integrity, confidentiality, availability and observation of information; perform a risk assessment from the implementation of the threat; determine the level of risk and provide recommendations for the need to process it; assess security risk by asset and business process.

The prospect of developing the proposed model is the use of adaptive neuro-fuzzy product model, which will allow reassessing risk in case of changes in the values of factors, changes in the product base of rules or in case of new risk.

The use of a linguistic approach ensures the possibility of using quantitative description of both all and individual elements of the model, provided that there is only about the value of fuzzy security risk factors, which provides opportunities, if necessary, to separate and rank risk factors and their consequences. Such actions may be useful in determining ways to avoid and / or reduce the negative impact of risk.

The use of neuro-fuzzy system components gives the model flexibility. Setting up the model by training in accordance with the obtained knowledge base allows you to perform risk reassessment in case of changes in the values of factors, changes in the product base of rules or the emergence of new risks. This provides an opportunity to shape and adapt the model to a specific ERP system.

6 DISCUSSIONS

Violation of security, including noncompliance with regulatory standards, can lead to financial and reputational consequences that are best avoided for any organization, regardless of size, scope or form of ownership.

The operating procedures and business applications that support them must be strategically managed and monitored to ensure the integrity, availability and confidentiality of the data that the organization owns.

Currently, the vast majority of organizations rely on ERP Systems to implement business processes and integrate financial data. The ERP system is an application system that implements a strategy of comprehensive resource planning that integrates the company's business processes and financial data into one platform. Integration provides better quality and availability of information, but it also increases the risk of fraud from within the organization by users and malicious attacks from outside. This dependency increases the security value of the ERP-system to protect your organization's information assets.

A key aspect of any security strategy is the ability to achieve a level of security that adequately demonstrates the organization's commitment to security and data security regulations collected from its customers and partners. Too little security increases the risk of violations, while too much can lead to unnecessary costs for information technology, software and hardware, deteriorating system performance, and slowing down business processes. There is no optimal security solution for any ERP-system. Each organization needs to assess risks and set goals related to their environment and the type of information it processes.

The peculiarity of risk assessment tasks is that most of the data on risk factors has signs of imperfection and uncertainty: contradiction, inaccuracy, unreliability or incompleteness, are nonlinear and dynamically variable. For effective assessment in case of uncertainty of input data, fuzzy logic methods and neuro-fuzzy networks are used to use linguistic variables and statements to describe risk factors and be adaptive at the expense of the neuro-network component.

CONCLUSIONS

The developed fuzzy evaluation model of the ERP-system was practically implemented using the Matlab en-

vironment. The implemented model was improved by using a fuzzy output algorithm in the form of a fuzzy production network, namely the ANFIS system of the Fuzzy Logic Toolbox package of the Matlab environment, which implements the Sugeno fuzzy output algorithm. Model training was conducted on different volumes and content of educational data, as well as for different number of learning epochs. The data obtained as a result of the analysis showed:

1) ANFIS-systems have a much lower estimate of error in obtaining the result of a logical conclusion;

2) increasing the size of the sample and increasing the number of learning epochs both individually and together improve the quality of the conclusion by increasing the accuracy of the result.

ACKNOWLEDGEMENTS

The work was performed at the Department of Information and cybernetic security of the State University of telecommunications within scientific researches conducted by the department.

REFERENCES

1. Leighton J. Security Controls Evaluation, Testing and Assessment Handbook. Syngress, 2016, 678 p.
2. Methody zahysty systemy upravlinnia informaciiou Bezpeku [Tekst], DSTU ISO/IES 27001, 2015. Chyn. 2017.01.01. Kyiv, DP "UkrNDNC", 2016, 22 p.
3. Informaciini tehnologii. Metody zahystu. Zvid praktyk shchodo zahodiv informaciiou bezpeky [Tekst], ISO/IES 27002:2015, 2015, Chyn. 2017.01.01. Kyiv, DP "UkrNDNC", 2016.
4. Informaciini tehnologii. Metody zahystu. Systemy ke-ruvannia informaciiou bezpekiou. Nastanova [Tekst], DSTU ISO/IES 27003, 2018, Chyn. 2018.01. 01. Kyiv, DP "UkrNDNC", 2018.
5. Informaciini tehnologii. Metody zahystu. Systemy ke-ruvannia informaciiou bezpekiou. Monitoring, Vy-miriuivannia, analisuivannia ta ociniuvannia [Tekst], DSTU ISO / IES27004, 2015, 2018, Chyn. 2018.01. 01. Kyiv, DP "UkrNDNC", 2018.
6. Informaciini tehnologii. Metody zahystu. Upravlinnia Rysykamy informaciiou bezpeki [Tekst], DSTU ISO / IES 27001: 2015, Chyn. 2015.01.01. Kyiv, DP "Ukr-NDNC", 2016.
7. Ehlakov Yu. P. Nechyotkaya model ocenki riskov Prodvizheniya programnyh produktov, *Biznes-informatika*, 2014, No. 3 (29), pp. 69–78.
8. Gladyshev S. V. Predstavlenie znaniy ob upravlenii in-Cyudentami informacionnoj bezopasnosti posredstvom Nechyotkich vremennyh raskrashennyh Setei Petri, *Mizhnarodnyi naukovotekhnichnyi zhurnal "Informaciini tehnologii ta kompyuterna inzheneriia"*, 2010, No. 1 (17), 2010, pp. 57–64.
9. Nieto-Morote A. A., RuzVila F. Fuzzy approach to construction Project risk assessment, *International Journal of Project Management*, 2011, Vol. 29, Issue 2, pp. 220–231.
10. Kozhukhivskiy A. D., Kozhukhivska O. A. ERP-System Risk Assessment Methods and Models (Tekst), *Radio Electronics, Computer Science, Control*, 2020, No. 4(55), pp. 151–162. DOI 10.15588/1607-3274-2020-4-15
11. Kozhukhivskiy A. D., Kozhukhivska O. A. Developing a Fuzzy Risk Assessment Model for ERP-Systems (Tekst) *Radio Electronics, Computer Science, Control*, 2022, No. 1, pp. 106–119. DOI 10.15588/1607-3274-2022-1-12
12. Baskerville R. An analysis survey of information system security design methods: Implications for Information Systems Development, *ACM Computing Survey*, 1993, pp. 375–414.
13. Peltier T. R. Facilitated risk analysis process (FRAP). Auerbach Publication, CRC Press LLC, 2000, 21 p.
14. Alberts C., Dorofee A. Managing Information Security Risks: The Octave Approach. Addison-Wesley Professional, 2002, 512 p.
15. Stolen K., Den Braber F., Dimitrakos T. Model-based risk assessment – the CORAS approach [Elektronnyi resurs], 2002, Rezhim dostupu: <http://folk.uio.no/nik/2002/Stolen.pdf>
16. Suh B., Han I. The IS risk analysis based on business model, *Information and Management*, 2003, Vol. 41, No. 2, pp. 149–158.
17. Karabacaka B., Songukpinar I. ISRAM: Information security risk analysis method, *Computer & Security, March*, 2005, pp. 147–169.
18. Goel S., Chen V. Information security risk analysis – a matrix-based approach [Elektronnyi resurs], University at Albany, SUNY, 2005, Rezhim dostupu: <https://www.albany.edu/~goel/publications/goelchen2005.pdf>
19. Elky S. An introduction to information system risk management [Elektronnyi resurs], *SANS Institute InfoSec Reading Room*, 2006, Rezhim dostupu: <https://www.sans.org/reading-room/whitepapers/auditing/introduction -information-system-risk-management-1204>.
20. Yazar Z. A. Qualitative risk analysis and management tool – CRAMM [Elektronnyi resurs], *SANS Institute InfoSec Reading Room*, 2011. Rezhim dostupu: <https://www.sans.org/reading-room/whitepapers/auditing/qualitative -risk-analysis-management-tool-cramm-83>
21. Korchenko A. G. Postroenie system zashchity informacii na nechetkikh mnozhestvah. Teoriya i prakticheskie resheniya. Kyiv, MK-Press, 2006, 320 p.: IL.
22. Karpenko A.C. Lohika Lukasevicha i proste chisla. Moscow, Nauka, 2000, 319 p.
23. Teoriya algoritmov ta matematychna lohika [Elektronnyi resurs], *Materialy dystancinogo navchmya sumskogo derzhavnogo universytetu*. Rezhim dostupu: <https://dl.sumdu.edu.ua/textbooks/85292/354091/index.html>
24. Kruglov V. V., Borisov V. V., Fedulov A. C. Nechitki modeli i seti. Moscow, Goriachaya liniya, Telekom, 2012, 284 p. IL.
25. Kruglov V. V., Borysov V. V. Iskusstvennye neironnye seti. Teoriya i praktika. Moscow, Goriachaya liniya, Telekom, 2002, 382 p.: IL.
26. Zade L. Ponyatie lingvisticheskoi pemonnoi i ego Primenenie k ponyatiyu priblizhonnykh reshenii, Per. s Angl. Moscow, Mir, 1976, 166 p.
27. Jang J.-S. R. ANFIS: Adaptive Network – based Fuzzy Inference System, *IEEE Trans. On System, Man and Cybernetics*, 1993, Vol. 23, No. 3, pp. 665–685.
28. Common Vulnerability Scoring System version 3.1: Specification Document. CVSS Version 3.1 Release [Elektronnyi resurs], *Forum of Incident Response and Security Teams*. Rezhim dostupu: <https://www.first.org/cvss/ specification-document>
29. National vulnerability database Release [Elektronnyi resurs], *National Institute of Standards and Technology*. Rezhim dostupu: <https://nvd.nist.gov>
30. FUZZY LOGIC TOOLBOX [Elektronnyi resurs], *Czentr Inzhenernykh Tekhnologii i Modelirovaniia Eksponenty*, Rezhim dostupu: <https://exponenta.ru/fuzzy-logic-toolbox>.

Received 10.08.2022.

Accepted 20.10.2022.

УДК 004.94

МОДЕЛЮВАННЯ ОЦІНКИ РИЗИКІВ ERP-СИСТЕМИ

Кожухівський А. Д. – д-р техн. наук, професор, професор кафедри інформаційної та кібернетичної безпеки Державного університету телекомунікацій, Київ, Україна.

©Kozhukhivskiy A. D., Kozhukhivska O. A., 2022

DOI 10.15588/1607-3274-2022-4-12

Кожухівська О. А. – д-р техн. наук, доцент кафедри інформаційної та кібернетичної безпеки Державного університету телекомунікацій, Київ, Україна.

АНОТАЦІЯ

Актуальність. Оскільки оцінка ризиків безпеки є складним і повним процесом невизначеності, а невизначеність є основним фактором, що впливає на ефективність оцінки, доцільно використовувати нечіткі методи та моделі, які є адаптивними до необчислюваних даних. Формування розпливчастих оцінок факторів ризику є су-б'єктивним, а оцінка ризиків залежить від практичних результатів, отриманих у процесі обробки ризиків загроз, які вже виникли під час функціонування організації та досвіду фахівців з безпеки. Тому доцільним буде використання моделей, що здатні адекватно оцінювати нечіткі фактори та мають можливість корегування їх впливу на оцінку ризику. Найбільші показники ефективності для вирішення таких задач мають нейро-нечіткі моделі, що комбінують методи нечіткої логіки та штучних нейронних мереж і систем, тобто «людиноподібного» стилю міркувань нечітких систем з навчанням та моделюванням розумових явищ нейронних мереж. Для побудови моделі розрахунку оцінки ризику безпеки пропонується використовувати нечітку продукційну модель. Нечіткі продукційні моделі (нечіткі моделі/системи на основі правил) це поширений тип нечітких моделей, які використовуються для опису, аналізу та моделювання складних систем і процесів, що слабо формалізуються.

Мета роботи – розробка нечіткої моделі оцінки ризиків безпеки та захисту систем ERP шляхом використання нечітких нейронних моделей.

Метод. Для побудови моделі розрахунку оцінки ризику безпеки пропонується використовувати нечітку продукційну модель. Нечіткі продукційні моделі це загальний вид нечітких моделей, які використовуються для опису, аналізу та моделювання складних систем і процесів, що слабо формалізуються.

Результати. Визначено фактори, що впливають на оцінку ризиків, запропоновано використання лінгвістичних змінних для їх опису та використання нечітких змінних для оцінки їх якостей, а також системи якісних оцінок. Обґрунтовано вибір параметрів та реалізовано нечітку продукційну модель оцінювання ризиків та бази правил нечіткого логічного висновку з використанням пакету прикладних програм MATLAB та пакету розширення Fuzzy Logic Toolbox, а також покращено за рахунок введення адаптивності моделі до експериментальних даних шляхом впровадження в модель нейро-нечітких компонентів. Розглянуто використання нечітких моделей для вирішення задач оцінки ризиків безпеки, а також концепцію та побудову ERP-систем та проаналізовано проблеми їх безпеки та вразливості.

Висновки. Розроблено нечітку модель оцінки ризиків ERP-системи. Обрано перелік факторів, що впливають на ризик безпеки. Запропоновано методи оцінки ризику інформаційних ресурсів та ERP-систем взагалі, оцінки фінансових збитків від реалізації загроз, визначення типу ризику за його оцінкою для формування рекомендацій відносно їх обробки з метою підтримки рівня захищеності ERP-системи. Визначено перелік лінгвістичних змінних моделі. Обрано структуру бази нечітких продукційних правил – MISO-структуру. Побудовано структуру нечіткої моделі. Визначено нечіткі змінні моделі.

КЛЮЧОВІ СЛОВА: безпека, нечітка логіка, нечітка продукційна модель, оцінка ризиків, захищеність, ERP-система.

УДК 004.94

МОДЕЛИРОВАНИЕ ОЦЕНКИ РИСКОВ ERP-СИСТЕМЫ

Кожуховский А. Д. – д-р техн. наук, профессор, профессор кафедры информационной та кібернетической безопасности Государственного университета телекоммуникаций, Киев, Украина.

Кожуховская О. А. – д-р техн. наук, доцент кафедры информационной та кібернетической безопасности Государственного университета телекоммуникаций, Киев, Украина.

АННОТАЦИЯ

Актуальность. Поскольку оценка рисков безопасности является сложным и полным процессом неопределенности, а неопределенность является одним из основных факторов, влияющих на эффективность оценки, целесообразно использовать нечеткие методы и модели, которые являются адаптивными к неучтенным данным. Формирование расплывчатых оценок факторов риска субъективно, а оценка рисков зависит от практических результатов, полученных в процессе обработки рисков угроз, которые уже возникли в ходе функционирования организации, и опыта специалистов по безопасности. Поэтому целесообразно использовать модели, которые могут адекватно оценивать нечеткие факторы и иметь возможность корректировать их влияние на оценку рисков. Наибольшими показателями эффективности для решения таких проблем являются нейро-нечеткие модели, сочетающими методы нечеткой логики и искусственные нейронные сети и системы, т.е. «человекоподобный» стиль соображений нечетких систем с обучением и моделированием психических явлений нейронных сетей. Для построения модели расчета оценки рисков безопасности предлагается использовать нечеткую модель продукта. Нечеткие модели продуктов (нечеткие модели/системы на основе правил) являются обычным типом нечетких моделей, используемых для описания, анализа и моделирования сложных систем и процессов, которые плохо формализованы.

Цель работы – разработка нечеткой модели оценки рисков безопасности и защиты систем ERP с использованием нечетких нейронных моделей.

Метод. Для построения модели расчета оценки рисков безопасности предлагается использовать нечеткую модель продукта. Нечеткие модели продуктов являются обычным видом нечетких моделей, используемых для описания, анализа и моделирования сложных систем и процессов, которые плохо формализованы.

Результаты. Выявленные факторы, влияющие на оценку риска, свидетельствуют об использовании лингвистических переменных для их описания и использования нечетких переменных для оценки их качеств, а также системы качественных оценок. Обоснован выбор параметров и реализованы нечеткая модель оценки рисков и основы правил нечеткого логического заключения с использованием пакета прикладных программ MATLAB и пакета расширения Fuzzy Logic Toolbox, а также улучшено за счет введения адаптивности модели к экспериментальным данным путем внедрения в модель нейро-нечетких

компонентов. Рассмотрено использование нечетких моделей для решения проблем оценки рисков безопасности, а также концепция и строительство систем ERP и проанализированы проблемы их безопасности и уязвимости.

Выводы. Разработана нечеткая модель оценки рисков системы ERP. Выбран перечень факторов, влияющих на риск безопасности. Предлагаются методы оценки рисков информационных ресурсов и ERP-систем в целом, оценка финансовых потерь от реализации угроз, определение вида риска в соответствии с его оценкой для формирования рекомендаций по их обработке в целях поддержания уровня защиты системы ERP. Определен список лингвистических переменных модели. Выбрана структура базы данных нечетких правил продукта – MISO-структура. Построена структура нечеткой модели. Выявлены нечеткие переменные модели.

КЛЮЧЕВЫЕ СЛОВА: безопасность, нечеткая логика, нечеткая производственная модель, оценка рисков, защищенность, ERP-система.

ЛІТЕРАТУРА / LITERATURE

1. Leighton J. Security Controls Evaluation, Testing and Assessment Handbook / J. Leighton. – Syngress, 2016. – 678 p.
2. Методи захисту системи управління інформаційною безпекою [Текст]: ДСТУ ISO/IEC 27001:2015. – 2016. – Чин. 2017.01.01. – Київ : ДП «УкрНДНЦ», 2016. – 22 с.
3. Інформаційні технології. Методи захисту. Звід практик щодо заходів інформаційної безпеки [Текст] : ДСТУ ISO/IEC 27002: 2015. – 2015. – Чин. 2017. 01.01. – Київ : ДП «УкрНДНЦ», 2016.
4. Інформаційні технології. Методи захисту. Системи керування інформаційною безпекою. Настанова [Текст] : ДСТУ ISO/IEC 27003: 2018.–2018. – Чин. 2018.10.01. – Київ : ДП «УкрНДНЦ», 2018.
5. Інформаційні технології. Методи захисту. Системи керування інформаційною безпекою. Моніторинг, вимірювання, аналізування та оцінювання [Текст]: ДСТУ ISO/IEC 27004: 2018. 2018. – Чин. 2018.10. 01. – Київ : ДП «УкрНДНЦ», 2018.
6. Інформаційні технології. Методи захисту. Управління ризиками інформаційної безпеки [Текст]: ДСТУ ISO/IEC 27005:2015.–2015.–Чин. 2017.10.01. – Київ : ДП «УкрНДНЦ», 2016.
7. Ехлаков Ю. П. Нечеткая модель оценки рисков продвижения программных продуктов / Ю. П. Ехлаков // Бизнес-информатика. – 2014. – №3 (29). – С. 69–78.
8. Гладыш С. В. Представление знаний об управлении инцидентами информационной безопасности посредством нечетких временных раскрашенных сетей Петри / С. В. Гладыш // Міжнародний науково-технічний журнал «Інформаційні технології та комп'ютерна інженерія». – 2010. – № 1(17). – С. 57–64.
9. Nieto-Morote A. A. Fuzzy approach to construction Project risk assessment / A. Nieto-Morote, F. RuzVila // International Journal of Project Management. – 2011. – Vol. 29, Issue 2. – P. 220–231.
10. Kozhukhivskiy A. D. ERP-System Risk Assessment Methods and Models (Tekst) / A. D. Kozhukhivskiy, O. A. Kozhukhivska // Radio Electronics, Computer Science, Control. – 2020. – No. 4(55). – P. 151–162.
11. Kozhukhivskiy A. D. Developing a Fuzzy Risk Assessment Model for ERP – Systems (Tekst) /A.D. Kozhukhivskiy, O.A. Kozhukhivska // Radio Electronics, Computer Science, Control. – 2022. – No. 1. – P. 106–119. DOI 10. 15588/1607-3274-2022-1-12
12. Baskerville R. An analysis survey of information system security design methods: Implications for Information Systems Development / R. Baskerville // ACM Computing Survey. – 1993. – P. 375–414.
13. Peltier T. R. Facilitated risk analysis process (FRAP) / T. R. Peltier. – Auerbach Publication. – CRC Press LLC, 2000. – 21 p.
14. Alberts C. Managing Information Security Risks: The Octave Approach / C. Alberts, A. Dorofee. – Addison-Wesley Professional, 2002. – 512 p.
15. Stolen K. Model-based risk assessment – the CORAS approach [Elektronnyi resurs] / K Stolen, F. den Braber, T. Dimitrakos. – 2002. – Rezhim dostupu: <http://folk.uio.no/nik/2002/Stolen.pdf>
16. Suh B. The IS risk analysis based on business model / B. Suh, I. Han // Information and Management. – 2003. – Vol. 41, No. 2. – P. 149–158.
17. Karabacaka B. ISRAM: Information security risk analysis method / B. Karabacaka, I. Songukpinar. – Computer & Security, March. – 2005. – P. 147–169.
18. Goel S. Information security risk analysis – a matrix-based approach [Elektronnyi resurs] / S. Goel, V. Chen. – University at Albany. – SUNY. – 2005.– Rezhim dostupu: <https://www.albany.edu/~goel/publications/goelchen2005.pdf>
19. Elky S. An introduction to information system risk management [Elektronnyi resurs] / S. Elky. – SANS Institute InfoSec Reading Room. – 2006. – Rezhim dostupu:<https://www.sans.org/reading-room/whitepapers/auditing/introduction-information-system-risk-management-1204>.
20. Yazar Z. A. Qualitative risk analysis and management tool – CRAMM [Elektronnyi resurs] / Z. A. Yazar. – SANS Institute InfoSec Reading Room. – 2011. – Rezhim dostupu: <https://www.sans.org/reading-room/whitepapers/auditing/qualitative-risk-analysis-management-tool-cramm-83>
21. Корченко А. Г. Построение систем защиты информации на нечетких множествах. Теория и практические решения / А. Г. Корченко – К. : МК-ПресС, 2006. – 320 с.: ил.
22. Карпенко А. С. Логика Лукасевича и простые числа / А. С. Карпенко. – М. : Наука, 2000. – 319 с.
23. Теорія алгоритмів та математична логіка [Електронний ресурс] / Матеріали дистанційного навчання Сумського державного університету. – Режим доступу: <https://dl.sumdu.edu.ua/textbooks/85292/354091/index.html>.
24. Круглов В.В. Нечеткие модели сети / В. В. Круглов, В. В. Борисов, А. С. Федулов. – М. : Горячая линия – Телеком, 2012. – 284 с.: ил.
25. Круглов В. В. Искусственные нейронные сети. Теория и практика / В. В. Круглов, В. В. Борисов. – М. : Горячая линия-Телеком, 2002. – 382 с.: ил.
26. Заде Л. Понятие лингвистической переменной и его применение к принятию приближенных решений / Л. Заде. – Пер. с англ. – М. : Мир, 1976. – 166 с.
27. Jang J.-S. R. ANFIS: Adaptive Network-based Fuzzy Inference System / J.-S.R. Jang // IEEE Trans. On Syst. Man and Cybernetics. – 1993. – Vol. 23, № 3. – P. 665– 685.
28. Common Vulnerability Scoring System version 3.1: Specification Document. CVSS Version 3.1 Release [Elektronnyi resurs] // Forum of Incident Response and Security Teams. – Rezhim dostupu: <https://www.first.org/cvss/ specification-document>
29. National vulnerability database Release [Elektronnyi resurs] // National Institute of Standards and Technology. – Rezhim dostupu: <https://nvd.nist.gov>
30. FUZZY LOGIC TOOLBOX [Elektronnyi resurs] // Центр инженерных технологий и моделирования экспоненты – Rezhim dostupa: <https://exponenta.ru/fuzzy-logic-toolbox>.

Наукове видання

**Радіоелектроніка,
інформатика,
управління**

№ 4/2022

Науковий журнал

Головний редактор – д-р техн. наук С. О. Субботін
Заст. головного редактора – д-р техн. наук Д. М. Піза

Комп'ютерне моделювання та верстання
Редактор англійських текстів

С. В. Зуб
С. О. Субботін

Оригінал-макет підготовлено у редакційно-видавничому відділі НУ «Запорізька політехніка»

Свідоцтво про державну реєстрацію
КВ № 24220-14060 ПР від 19.11.2019.

*Підписано до друку 25.11.2022. Формат 60×84/8.
Папір офс. Різогр. друк. Ум. друк. арк. 18,83.
Тираж 300 прим. Зам. № 877.*

69063, м. Запоріжжя, НУ «Запорізька політехніка», друкарня, вул. Жуковського, 64

Свідоцтво суб'єкта видавничої справи
ДК № 6952 від 22.10.2019.