ISSN 2710-0766
DOI 10.31891/CSIT

THE INTERNATIONAL SCIENTIFIC JOURNAL

COMPUTER SYSTEMS AND INFORMATION TECHNOLOGIES

No 2-2023

МІЖНАРОДНИЙ НАУКОВИЙ ЖУРНАЛ

КОМП'ЮТЕРНІ СИСТЕМИ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

2023

COMPUTER SYSTEMS AND INFORMATION TECHNOLOGIES

INTERNATIONAL SCIENTIFIC JOURNAL

Professional publication of category "B"

Published since 2020 year ***Four time a year***

Khmelnitskyi, 2023, № 2 (11)

Establishers: **Khmelnytskyi National University** (Ukraine)
Associated establisher: **Institute of Information Technologies** (Slovakia)
National Library og Ukraine named after V.I.Vernadsky http://nbuv.gov.ua/j-tit/csit

The journal is included in scientometric databases:

Index Copernicus https://journals.indexcopernicus.com/search/details?id=69998&lang=en
Google Scholar https://scholar.google.com.ua/citations?hl=uk&user=HW1XpMsAAAAJ
CrossRef http://doi.org/10.31891/CSIT

Recommended for publication by the decision of the Academic Council of Khmelnytskyi National University, protocol № 14 from 29.06.2023

Editorial board address: International scientific journal "Computer Systems and Information Technologies", Khmelnytskyi National University, Institutska str. 11, Khmelnytskyi, 29016, Ukraine

☎ **(0382) 67-51-08**

e-mail: csit.khnu@gmail.com

web: http://csitjournal.khmnu.edu.ua/
http://lib.khnu.km.ua/csit_khnu.htm

Registered by the Ministry of Justice of Ukraine
Certificate of state registration of the print media
Series KB № 24924-14864PR dated 12.07.2021

КОМП'ЮТЕРНІ СИСТЕМИ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

МІЖНАРОДНИЙ НАУКОВИЙ ЖУРНАЛ

Фахове видання категорії "Б". Наказ Міністерства освіти і науки України про затвердження рішень Атестаційної колегії №420 від 19.04.2021 року

Засновано в 2020 р. ***Виходить 4 рази на рік***

Хмельницький, 2023, № 2 (12)

Засновник і видавець: **Хмельницький національний університет** (Україна)
Асоційований співзасновник: **Інститут інформаційних технологій** (Словаччина)

Наукова бібліотека України ім. В.І. Вернадського http://nbuv.gov.ua/j-tit/csit

Журнал включено до наукометричних баз:
Index Copernicus https://journals.indexcopernicus.com/search/details?id=69998&lang=en
Google Scholar https://scholar.google.com.ua/citations?hl=uk&user=HW1XpMsAAAAJ
CrossRef http://doi.org/10.31891/CSIT

Рекомендовано до друку рішенням Вченої ради Хмельницького національного університету, протокол № 14 від 29.06.2023

Адреса редакції: **Україна, 29016,**
м. Хмельницький, вул. Інститутська, 11,
Хмельницький національний університет
редакція журналу "Комп'ютерні системи та інформаційні технології"

☎ **(0382) 67-51-08**

e-mail: csit.khnu@gmail.com

web: http://csitjournal.khmnu.edu.ua/
http://lib.khnu.km.ua/csit_khnu.htm

Зареєстровано Міністерством юстиції України
Свідоцтво про державну реєстрацію друкованого засобу масової інформації
Серія КВ № 24924-14864ПР від 12 липня 2021 року

CONTENTS

ЗМІСТ

https://doi.org/10.31891/csit-2023-2-1
UDC 519.6

Lesia MOCHURAD
Lviv Polytechnic National University

CUDA-BASED PARALLELIZATION OF GRADIENT BOOSTING AND BAGGING ALGORITHM FOR DIAGNOSING DIABETES

Data, its volume, structure, and form of presentation are among the most significant problems in working in the medical field. The probability of error is very high without innovative high-tech data analysis tools. It is easy to miss an important factor that is critical but lost among other, less important information. This work aims to study the proposed parallel gradient boosting algorithm in combination with the Bagging algorithm in the classification of diabetes to achieve greater stability and higher accuracy, reduce computational complexity and improve performance in medicine. The methods of parallelization of the Gradient Boosting algorithm in combination with the Bagging algorithm are investigated in the paper. Performance scores were obtained: approximately 7 using ThreadPoolExecutor and an eight-core computer system and 9.5 based on CUDA technology. Performance indicators that go to the unit are calculated. This, in turn, confirms the effectiveness of the proposed parallel algorithm. Another significant result of the study is improving algorithm accuracy by increasing the number of algorithms in the composition. The problem of diagnosing a patient's diabetes based on specific measurements included in the data set is considered. Detailed analysis and pre-processing of the selected dataset were performed. The parallelization of the proposed algorithm is implemented using the multi-core architecture of modern computers and CUDA technology. The process of learning models and training samples was parallelized. The theoretical estimation of the computational complexity of the offered parallel algorithm is given. A comparison of serial and parallel algorithm execution time using ThreadPoolExecutor when varying the number of threads and algorithms in the composition is presented. And also, the comparative analysis of time expenses at consecutive and parallel execution based on CPU and GPU is carried out.

Key words: ensemble of models, acceleration, graphic processor, CUDA technology, machine learning, classification problem.

Леся МОЧУРАД
Національний університет «Львівська політехніка»

РОЗПАРАЛЕЛЮВАННЯ АЛГОРИТМУ ГРАДІЄНТНОГО БУСТИНГУ ТА БЕГГІНГУ ДЛЯ ДІАГНОСТИКИ ДІАБЕТУ НА ОСНОВІ CUDA

Важливим аспектом, який відрізняє медичні дані від більшості інших, є те, що об'єктивність, точність, якість і своєчасність результатів є критично важливими і повинні постійно ставитися під сумнів.

Одну з найбільших складнощів в процесі роботи в медичній сфері становлять дані: їх об'єм, структура та форма представлення. Очевидно, що без інноваційних високотехнологічних інструментів аналізу даних, ймовірність помилки є дуже високою, адже легко пропустити якийсь важливий фактор, що є критичним, але загубленим серед іншої, менш важливої інформації. У роботі досліджено методи паралелізації алгоритму Градієнтного Бустингу у поєднанні з алгоритмом Беггінгу. Розглянуто задачу діагностичного прогнозування наявності у пацієнта діабету на основі певних вимірювань, включених до набору даних. Проведено детальний аналіз та попередню обробку обраного датасету. Розпаралелювання запропонованого алгоритму реалізовано за допомогою модуля Python – concurrent. Futures, а саме його класу – ThreadPoolExecutor та технології CUDA. Здійснено паралелізацію процесу навчання моделей та навчальної вибірки. Наведено теоретичну оцінку обчислювальної складності запропонованого паралельного алгоритму. Проведено порівняння часу виконання послідовного та паралельного алгоритмів при використанням ThreadPoolExecutor при варіації кількості потоків та алгоритмів в композиції. А також здійснено порівняльний аналіз часових витрат при послідовному та паралельному виконанні на основі CPU і GPU. Отримано показники прикорення: приблизно 7 при використанні ThreadPoolExecutor та восьмиядерної обчислювальної системи та 9,5 на основі технології CUDA. Обчислено показники ефективності, які прямують до одиниці. Що, в свою чергу, підтверджує ефективність запропонованого паралельного алгоритму. Ще одним важливим результатом дослідження є досягнення кращої точності алгоритму за рахунок збільшення кількості алгоритмів у композиції.

Ключові слова: ансамбль моделей, прискорення, графічний процесор, технологія CUDA, машинне навчання, задача класифікації.

Introduction

Today, one of the most important factors influencing the development of human society is artificial intelligence (AI). The latter is used in various fields of health and medicine [1-3]. An important aspect that distinguishes medical data from most others is that results' objectivity, accuracy, quality, and timeliness are critical and should be constantly questioned.

Various methods of machine learning for disease diagnosis have attracted attention in recent years [4]. With the increase in the number of people coming to hospitals, conventional traditional methods may not be enough. Thus, various forms of AI, such as natural language processing [5, 6], data processing algorithms [7, 8], clustering, and classification [9, 10], should be used to improve the efficiency of the medical system significantly.

The use of AI has some benefits: it reduces human error, saves time and money, and improves service delivery. Moreover, machine learning methods have repeatedly shown better accuracy compared to medical personnel.

The basis of medical diagnosis is the problem of classification. The diagnosis can be reduced to a problem with displaying data to one of the different results. In some cases, the task is only to determine based on data (such as radiographs or electrocardiography) whether the patient has a specific disease – (1) or not – (0).

Over the past few years, the growth of diabetes among people has become exponential [11, 12]. A health report shows that about 347 million people worldwide suffer from diabetes. Diabetes affects not only the elderly but also the younger generation. Detecting diabetes at an early stage is a big problem. Therefore, diagnosing diabetes at an early stage will be helpful in the decision-making process of the medical system and will help save lives. After all, prolonged diabetes leads to the risk of damage to vital organs of the human body.

Globally, the incidence of diabetes is snowballing, even if we consider the increase in population in recent decades. This, in turn, leads to rapid growth in the amount of medical data. This increase in health data means that AI will be increasingly used in this area. In this case, for the effective operation of AI methods, their accuracy and the speed of execution must be high, which makes the topic of the study relevant.

The healthcare information technology industry is changing daily. As a result of this rapid development, new scientific advances in machine learning have enabled the healthcare industry to utilize several revolutionary tools that use natural language processing, pattern recognition, and "deep" learning. Of course, enterprises still have work to do before machine learning and AI can meet the needs of modern medicine. Still, it is worth noting that innovative technologies are already leaving their mark in the environment of medical data analysis [13-16].

The main contribution of this article can be summarized as follows:

1. We have proposed an accurate and computationally efficient approach to the parallelization of the combination of two algorithms – Gradient Boosting and Bagging – to solve the diagnostic prediction of the presence of diabetes in patients based on specific measurements. This provides an opportunity to significantly optimize the computational process by parallelizing it, solving significant data processing in medicine;

2. We have developed an algorithmic implementation of the proposed approach. The relevance of the development of the multi-core architecture of modern CPUs and the use of CUDA technology for graphics processors was taken into account;

3. We have demonstrated the reduction of computational complexity using the proposed algorithm; we have an acceleration of approximately seven times for the octa-core architecture of the corresponding computing system when using ThreadPoolExecutor (this result can be significantly improved by choosing a CPU with more than eight cores) and 9.5 times for GPUs based on CUDA technology; the efficiency indicator which goes to the unit is calculated; achieving better algorithm accuracy by increasing the number of algorithms in the composition, which is an excellent opportunity to increase the efficiency of the model.

Related works

Since the problems investigated in this paper are relevant, an analysis of the literature related to this range of tasks was conducted. This provided an opportunity to study the studied algorithms in detail, determine the achievements in their use for the diagnosis of diabetes, and explore existing approaches to improve the Gradient Boosting algorithm.

In [17], the author used boosting and running to enhance the decision tree algorithm in predicting diabetes risk. An accuracy of 89.65% was achieved. Despite the high precision, the question of the execution time of the algorithms remains open. After all, small amounts of data (768 instances) [18] were used by the author, which suggests that the time will be too extensive with more data. So, in contrast to the previous one, in this paper, the dataset [19] that provides information about 10000 patients was chosen in order to find and test methods of accelerating the proposed algorithms.

Many machine learning methods have been discussed, starting from different basic algorithms such as the logistic regression, a modified support vector machine, a decision trees, to further classification including the Iterative Dichotomiser 3, C4.5, C5.0, J48 and Classification and Regression Tree and the naïve Bayes on diabetes detection [20]. Ensemble methods, such as Bagging, Boosting, and Random Forest (RF) regressors, are further used to enhance the accuracy and performance of models. These techniques have been implemented on all types of platforms such as Python or MATLAB, and the models have been analyzed using different parameters such as area under curve or confusion matrices or mathematical terms such as the Root-mean-square error or Mean absolute error. However, the work does not provide an analysis of parallel algorithms for solving the problem of big data analysis on diabetes detection. And today, the possibility of speeding up decision-making in medicine is extremely important.

A comparison of gradient boosting with two other machine learning algorithms: RF and neural networks on a dataset for diagnosing diabetes was carried out in [21]. Studies have shown the benefits of boosting. This article provided a better understanding of the benefits of the practical application of this method.

The research paper [22] presents a methodology forclassification of diabetic and normal heart rate variability signals using deep learning architectures. The classification system proposed can help the clinicians to diagnose diabetes using electrocardiogram signals with a very high accuracy of 95.7%. As the authors note, the highest value published for the automated diabetes detection with HRV as input data was obtained in the work. And further improvement in accuracy can be obtained using a very large-sized input dataset. However, this, in turn, requires modification of existing algorithms in order to obtain a solution in real-time.

In [23], the authors improved the parallel efficiency of the decision tree by proposing a new GBDT system – HarpGBDT. This approach includes a block strategy of parallelism and extension of the TopK tree growth method (which selects the best K candidates of tree nodes to allow more levels of parallelism without sacrificing the algorithm's accuracy). In addition to the description of this approach, the paper presents a comparative analysis with other parallel implementations, which states that despite the many advantages over other approaches, the operating time is longer. This, in turn, indicates the need to find a method of parallelization, which will give a significant acceleration among the many advantages presented in the paper.

The authors paralleled Boosting and Bagging [24]. The research was conducted using decision trees and two standard data sets. The main results are that the sample size limits the achieved accuracy, regardless of the computational time. Therefore, this work encourages experiments on larger data and attempts to improve accuracy. The paper also demonstrated the parallelization of Boosting and Bagging methods separately, confirming our proposed method's novelty.

In [25], the authors developed the idea of ensemble learning, combining adaptive Boosting and Bagging for binary classification. The algorithms were tested on different data sets, showing improved accuracy and reduced error rates. However, this algorithm has increased the computation time. This study further actualizes the problem of parallelization of the combination of Boosting and Bagging algorithms.

In the work [26] a stacked ensemble-based deep neural network approach is proposed for diabetes possibility assessment in the early stages. As a result, the proposed method achieved the highest success rate with 99.36% accuracy and 99.19% AUC. But the proposed approach was tested on a dataset of 520 patients. In the work, the authors did not investigate the problem of speed of decision-making and changes in accuracy when the number of patients increases.

Thus, an analysis of the literature has shown that the use of the Gradient Boosting algorithm to diagnose diabetes is not sufficiently studied. Researches usually use an insignificant volume of data, and the realized parallelization marches do not provide the expected acceleration. In addition, no works have been found that describe the development of a similar algorithm to what will be implemented and investigated in the following sections. This once again confirms the value and relevance of this work.

Methodology

Combining Gradient Boosting and Bagging

As is known [27], Begging classification technology uses compositions of algorithms, each of which learn independently. In this case, each model is studied on a separate sample formed from the original data set. The output of the ensemble is determined by averaging the outputs of the basic models. The method allows to improve the accuracy and stability of machine learning algorithms, reduce error variance and reduce the effect of retraining. The method was initially developed for classifiers based on decision trees, but now it can be used for any model. At any given time, the model results are weighted based on previous results. Correctly predicted results are given less weight, and those that do not meet the classification – more. Therefore, several subsets are randomly selected from the set of source data, containing the number of objects corresponding to the number of objects in the original set. Remember that because the selection is random, the set of objects will always be different: some examples fall into several subsets and some into none. A classifier is built based on each sample. The results of the classifiers are aggregated. Aggregation of results usually occurs by averaging or voting. Moreover, the first option is used in the regression problem, and the second – classification.

Advantages:

- a significant increase in the accuracy of the prediction of the ensemble relative to the basic classifiers (from 10% to 40%). It is achieved by reducing the variance of the predictions of the basic models in averaging. Scatter is the variance of the answers of our models. The scatter shows how sensitive to small changes in the sample the algorithms are;
- small offset. Offset – the deviation of the average answers of all models from the answers of the most optimal model. The offset characterizes how complex the family of algorithms is, how much it can restore complex patterns;
- the use of bagging reduces deviations.

Disadvantages:

- insufficient mathematical justification for improving the accuracy of predictions.

Like bagging, the main task of this method of aggregation is to convert a set of weak classifiers (i.e., those that assume many errors in the test sample) into a stronger one. But, unlike the previous one, the training takes place sequentially, and each subsequent classifier aims to compensate for the shortcomings of the previous one. The result is usually a weighted linear combination of responses of all algorithms used.

The general idea of Boosting algorithms is to consistently apply predictors so that each subsequent model minimizes the error of the previous one [26].

Advantages:

- as an ensemble model, boosting is an easy-to-read and interpretive algorithm that makes predictive interpretations easy to use;
- the ability to predict is effective through the use of cloning techniques, such as bags or RF, and decision trees;
- boosting is an elastic method that easily curbs re-equipment.

Disadvantages:

- emission sensitive, as each classifier is obliged to correct the errors of the predecessors;
- the method is almost impossible to increase. This is due to the fact that each evaluator bases his correctness on previous predictors, which complicates the ordering procedure.

The gradient boosting method works consistently by adding new ones to past models to correct the mistakes made by previous predictors.

This method changes the weights with each iteration, teaching new models on the residual error of the past (moving to a minimum loss function).

Here are the steps to implement gradient Boosting:

1. The model is based on data collection.
2. This model makes predictions for the entire data set.
3. Errors are calculated according to forecasts and true value.
4. The new model is built, taking into account errors as target variables. At the same time, we strive to find a better separation to minimize error.
5. The predictions made with this new model are combined with the previous ones.
6. Errors are recalculated using these predicted values and true values.
7. This process is repeated until the error function stops changing or until the maximum number of predictors is reached.

Basic algorithm learning is consistent. Suppose that at some point $N-1$ algorithms $b_1(x),...,b_{N-1}(x)$ are trained, i.e. the composition has the form: $a_{N-1}(x)=\sum_{n=1}^{N-1} b_n(x).$

Now another algorithm $b_N(x)$ is added to the current composition. This algorithm is trained to minimize the composition error in the training sample: $\sum_{i=1}^{l} L(y, a(x)+b(x)) \to \min_b$.

First, it makes sense to solve a simpler problem: to determine what values of $s_1,...,s_l$ should take the algorithm $b_N(x_i)=s_i$ on the objects of the training sample, so that the error in the training sample is minimal: $F(S)=\sum_{i=1}^{l} L(y, a(x)+s) \to \min_s$, where $s=(s_1,...,s_l)$ – vector of shifts.

In other words, it is necessary to find a shift vector s that will minimize the function $F(S)$. Since the direction of the greatest decrease of the function is given by the direction of the antigradient, it can be taken as a vector s: $s=-\nabla F=(-L_z(y_1,a_{N-1}(x_1)), ..., -L_z(y_l,a_{N-1}(x_l))).$

The components of the shift vector s are, in fact, the values that the new algorithm $b_N(x)$ must take on the objects of the training sample to minimize the composition error. Learning $b_N(x)$, thus, is a learning task on marked data, in which $\{(x, s)\}$ is a training sample.

It should be noted that the information about the initial loss function $L(y, z)$, which is not necessarily quadratic, is in the optimal shift vector s. Therefore, for most tasks in learning $b_N(x)$ we can use the quadratic loss function.

When it is necessary to solve a complex computational problem and no algorithm is ideal, ensembles are used [28]. Ensembles make it possible to combine several algorithms that learn simultaneously and correct each other's mistakes. To date, they give the most accurate results.

A total sample of elements is first taken, then divided into sub-samples. Then the sample data are fed parallelly to the input of the basic algorithms, as in the bagging. The difference is that the basic algorithms are gradient boosting algorithms. After completing the process of independent learning, the algorithms are combined into an ensemble model. By inputting test characteristics to the algorithms, we can obtain the final prediction of the model, taking the average value of all predictions.

Description of the proposed algorithm

1. Specify the required number of gradient boosting models that we want to build. Let it be n.
2. Divide the initial data N by n subsamples, where n is the number of models of gradient boosting, which is specified in step 1. We obtain $N_1, ..., N_n$ models of size m, where $m=N/n$ is the amount of data in each model of gradient boosting.

3. Next, we implement the Bagging algorithm, running the training for each model of gradient boosting separately. In this case, each model is given a corresponding sub-sample for training (the first model is the first sub-sample, etc.). We get $w_1(\cdot), w_2(\cdot), ..., w_n(\cdot)$ independent weak learners (one for each subsample).

4. At each iteration, we fit the weak learner to the gradient of the current selection error relative to the current ensemble model. $s_n(\cdot) = s_{n-1}(\cdot) - c_n * \nabla_{s_{n-1}} E(s_{n-1}),$ where $E(\cdot)$ is the fit error of this model, c_n is the coefficient corresponding to the step size, $-\nabla_{s_{n-1}} E(s_{n-1})$ is the gradient of the fit error relative to the ensemble model.

5. At the output of the Bagging algorithm, we obtain n trained models (classifiers) of the Gradient Boosting algorithm.

6. To perform the prediction (assigning an object to a certain class), we pass the object to each of the models, namely a number of its features that need to be classified.

7. As a result, each model of Gradient Boosting assigns an object to the class to which the probability of belonging of the object according to its calculations is the highest. Belonging of object x to each of the classes:

$$P(y=1|x) = \frac{1}{1+\exp(-a(x))}, \quad P(y=-1|x) = \frac{1}{1+\exp(a(x))}.$$

8. Then find the average value between all predictions obtained from n models.

9. Round off the determined average value to the nearest integer, which will be the result of the algorithm of combining Bagging and Gradient Boosting, and, consequently, the class to which the transferred object belongs.

It is possible to parallelize the combination of algorithms in step 3 of the proposed algorithm, during which the models are studied sequentially, independently of each other.

Since the models learn independently, we can run their training parallelly. This is one of the main reasons why we used the idea of combining the Bagging algorithm with Gradient Boosting.

For example, when classifying the Gradient Boosting model, several classifiers are also trained. Still, this process cannot be parallelized, because here each subsequent classifier uses the results of the previous one and does not work independently, as in Bagging.

Therefore, when parallelizing, we change step 3 in the algorithm for combining Begging with Gradient Boosting, running parallel training for each model of Gradient Boosting separately. In this case, each thread will work with its subsample and its model.

Parallelization

Gradient boosting is a sequential algorithm for learning trees because the result of the previous weak algorithm is the input for another. That is why it is impossible to perform training parallelly. However, using gradient boosting as a basic algorithm for bagging will allow us to perform training parallelly.

The parallelization of bagging is quite simple. The training set is divided equally among the available processors (streams). Each processor (thread) executes a sequential algorithm until the appropriate predictions are made. In general, it is a good idea to divide the training set randomly to make sure that the predictions created by each processor (thread) do not contain unnecessary biases.

Parameters to select for the parallel ensemble technique: sample size used and number of iterations.

We will implement parallelization using the Python module – concurrent Futures, namely its class – ThreadPoolExecutor and CUDA technology [29].

We will parallel the learning process. The training sample will also be parallelized. In a single-threaded system, the training vectors will be sent to the classifier one by one. In a parallel system, these learning vectors will be separated between streams.

Threads in Python are a form of parallel programming that allows a program to perform multiple procedures simultaneously. Flow-based parallelization is especially well suited for accelerating applications that work with large amounts of data.

ThreadPoolExecutor is a utility that is built into Python 3, is located in the concurrent Futures module [30] and is designed to distribute code execution among threads (a pool of threads is formed). You must first import it from the specified module, then initialize the ThreadPoolExecutor() object.

The map function was also used, the syntax of which is as follows: *map(func, *iterables).*

The map method applies the func function to one or more iterating objects; in this case, the function is to build and train a model of the Gradient Boosting algorithm, and iterating objects are the parts into which the total data sample is divided. In this case, each function call is started in a separate thread. The map method returns an iterator with the function results for each element of the iterating object. The number of threads in which the code will run is specified in the max_workers parameter when declaring the ThreadPoolExecutor() object, applied to the map function.

So, on input ThreadPoolExecutor().map accepts a function that needs to be parallelized and arguments that need to be transferred to it. On an output, we receive an array that consists of the models trained on subsamples.

Next, a quality check will be performed independently for each model, and their quality metrics will be calculated (score and f1_score). In the end, when we leave the parallel area of our program, we will re-check the quality for a set of all models. The prediction of such a composition will be averaged and rounded.

As is known [31], the GPU speeds up programs running on the CPU by unloading some computing and time-consuming parts of the code. The rest of the program is still running on the CPU. From the user's point of view, the program runs faster because it uses the powerful parallel processing power of the graphics processor to increase performance.

The CPU consists of several processor cores, while the GPU consists of hundreds of smaller cores. Together they work to process data. This highly parallel architecture is what gives the GPU high computing performance.

The system that executes the software implementations presented in this paper supports CUDA technology, which provides the ability to use GPUs for parallelization and acceleration of program execution.

We will use LightGBM to call CUDA technology [32]. This Python framework has been used to improve gradient boosting. It is designed to increase efficiency with the following benefits:

- Faster training speed and higher efficiency.
- Less memory usage.
- Better accuracy.
- Support of parallel, distributed and GPU learning.
- Able to process large-scale data.

Classification will be done by calling the LGBMClassifier() method from the LightGBM library described above.

The complexity of the Boosting algorithm: the complexity of the training time: $O(M*n*\log(n)*d)$, where M is the number of trees, d is the number of signs, $\log(n)$ is the depth of the tree; n is the total amount of data.

The complexity of Boosting in conjunction with Bagging: $O(n)+O(K*M*m*\log(m)*d)$, where m is the sample size of the data used, $O(n)$ is the difficulty of sampling, n is the total data size. $O(K*M*m*\log(m)*d)$ – Boosting complexity for K iterations (K – number of Gradient Boosting algorithms).

The difficulty of one round of parallel learning for Boosting in conjunction with Bagging can be expressed in the formula: $O\left(\frac{n}{P}, m\right)+O(M*m*\log(m)*d)$, where m is the size of the data subsample used, $O\left(\frac{n}{P}, m\right)$ is the complexity of sampling (which depends on both the size of the local data set in the round and the size of the selected set according to the specified number of threads), n is total data size, P – number of threads. $O(M*m*\log(m)*d)$ – Boosting complexity in one round.

Data review and analysis

The paper uses a data set based on a long-term survey conducted among residents of Framingham, Massachusetts [19]. The purpose of the classification is to predict whether a patient is at risk of developing diabetes. The dataset provides information about 10000 patients and contains 15 attributes.

Each attribute is one of the potential risk factors: demographic, behavioral, and medical risk factors.

<u>Demographic attributes:</u>

- Sex: man or woman (nominal value);
- Age: age of the patient (continuous values).

<u>Behavioral attributes:</u>

- Current Smoker: whether the patient was a smoker at the time of the examination (nominal value);
- Cigs Per Day: the number of cigarettes a person smoked on average in one day (continuous value).

<u>Medical attributes:</u>

- BP Meds: whether the patient was receiving medication for blood pressure (nominal value);
- Prelevant Stroke: whether the patient has suffered a stroke (nominal value);
- Prevalent Hyp: whether the patient had a hypertensive crisis (nominal value);
- Risk of coronary heart disease (CHD): whether the patient was at risk of coronary heart disease (nominal value);
- Tot Chol: total cholesterol (continuous);
- Systolic blood pressure (Sys BP) (continuous);
- Diastolic blood pressure (Dia BP) (continuous);
- Body Mass Index (BMI) (continuous value);
- Heart Rate (continuous value – in medical research, variables such as heart rate, although discrete, are considered continuous due to a large number of possible values);
- Glucose: Glucose level (continuous).

<u>Variable for prediction:</u>

- Diabetes: a binary value of "1" means that diabetes has been detected and "0" has not been detected.

We performed the pre-processing and analysis of data to achieve the best quality of the model and detailed acquaintance with the data.

Experiments

Time costs of sequential and parallel implementations

Let's present the execution time results of the sequential and the proposed parallel algorithm using the module ThreadPoolExecutor and analyze them, where K – the number of algorithms in the composition.

Table 1

Program execution time for sequential and parallel processing (ThreadPoolExecutor), s

K	Sequential	Parallel (threads)			
		2	4	8	16
1	0.8053	0.8185	0.8109	0.8523	0.7743
2	1.6932	0.9073	0.8598	0.9075	0.8426
4	3.1604	1.7761	0.9813	1.1275	1.0676
8	6.5957	3.3760	1.9017	1.5271	1.6136
16	12.8843	6.5524	4.0012	3.4891	3.1071
30	23.9632	12.7330	9.4793	6.1386	5.9924

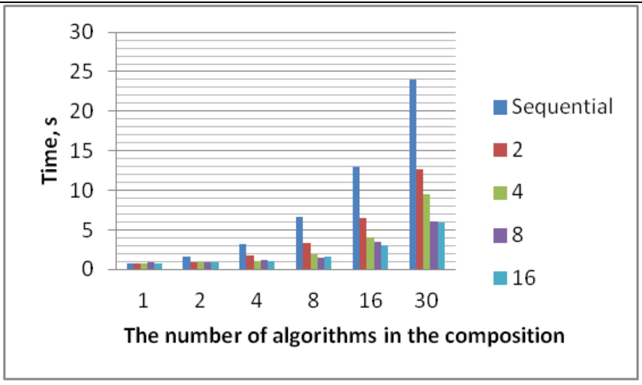


Fig. 1. Visualization of comparison of execution time of sequential and parallel algorithm (using ThreadPoolExecutor) with variation of flows and number of algorithms in a composition

From the graph shown in Figure 1 and Table 1, it can be seen that in comparison with the sequential execution of the program, run time is significantly reduced by parallelization. The greater is the number of threads, the lower is the time. Only when using more than eight threads, the time value hardly changes. This is due to the capabilities of the system's architecture on which this program was implemented (only four cores and eight logical processors, i.e., the maximum efficiency can be obtained by using eight threads).

Now let's perform a comparative analysis of the execution time of the sequential and the proposed parallel algorithm based on CPU and GPU.

Table 2

Program execution time for sequential and parallel processing on the CPU and GPU, s

K	Sequential	Parallel (CPU)	Parallel (GPU)
1	0.8053	0.7743	0.0842
2	1.6932	0.8426	0.2444
4	3.1604	1.0676	0.3494
8	6.5957	1.5271	0.7865
16	12.8843	3.1071	1.4916
30	23.9632	5.9924	2.8142

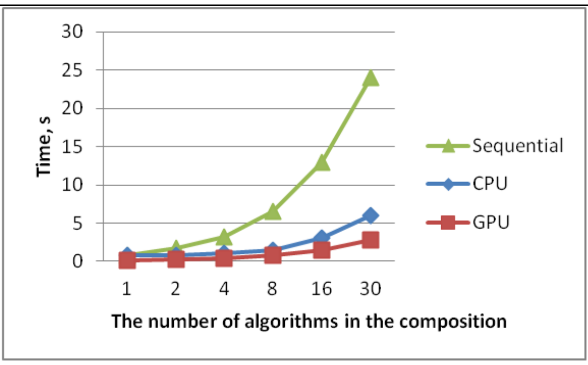


Fig. 2. Visualization comparing the execution time of sequential and parallel algorithms (using CPU and GPU) with variation in the number of algorithms in the composition

Figure 2 and Table 2 shows how rapidly the program execution time decreases when using GPU-based parallelization. To compare the time spent working with the CPU and GPU, we used the best results achieved on the CPU and still got a fairly significant acceleration on the GPU. This confirms that the use of GPUs in parallelization is very efficient and can provide impressive results. In this case, with 24 seconds of sequential execution, the graphics processor provided the ability to speed up the program to 3 seconds, which is a significant improvement.

Accuracy of the model

Now let's evaluate the quality of the model. For a more accurate estimate, we use two metrics – Accuracy and F1-score:

- Accuracy – an indicator that describes the overall accuracy of model predictions for all classes.
- F1-score is the harmonic mean value of Precision and Recall metrics, normalized between 0 and 1. If F1 score = 1, it indicates an ideal balance.

Table 3

Values of metrics Accuracy and F1-score for different number of algorithms studied parallelly

Metrics	K					
	1	2	4	8	16	30
Accuracy	0.986	0.987	0.991	0.925	0.992	0.998
F1-score	0.711	0.778	0.795	0.838	0.873	0.881

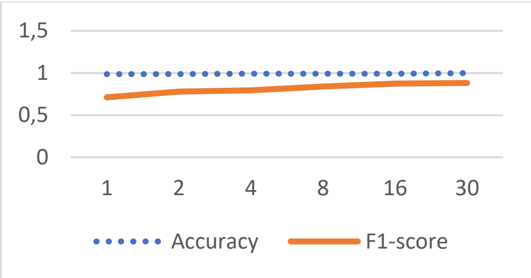


Fig. 3. Visualization of Accuracy and F1-score metrics depending on the number of algorithms

Analyzing Figure 3 and Table 3, we conclude that the accuracy is very high even when using only one algorithm but still increases slightly with an increasing number of algorithms. The value of the F1-score metric is smaller than the accuracy. Still, here we see more clearly the influence of the number of algorithms on the model's accuracy – the more algorithms in the composition, the higher the value of the metric.

Thus, the use of a parallel algorithm for combining Gradient Boosting with Begging tends to increase the accuracy of the constructed model. And the accuracy results at different values of 10 fold cross validation of the proposed algorithm were given in Table 4.

Table 4

Cross Validation Results

Fold	Loss	Accuracy(%)
1	0.015	99.685
2	0.016	99.789
3	0.011	99.899
4	0.019	98.987
5	0.013	99.843
6	0.091	97.345
7	0.026	99.341
8	0.051	98.562
9	0.016	99.876
10	0.065	97.560
Average	0.0323	99.0887

According to the results obtained (see Table 4), the proposed method reached 99.09% accuracy with 10 fold cross validation and also showed higher performance with an accuracy rate of 99.80%, although the percentage split (75:25).

Discussion of research results

Now we calculate the experimental indicators of acceleration and efficiency of parallel algorithms with different numbers of boosting algorithms in the bagging composition.

To calculate these indicators of acceleration and efficiency, we will use the following formulas: $S_p(n) = T_1(n)/T_p(n)$, $E_p(n) = S_p(n)/p$, respectively, where $T_1(n)$ is the time complexity of the sequential execution of the algorithm, $T_p(n)$ is the time complexity of the parallel execution of the algorithm for p processors (threads).

Table 5

Indicators of acceleration of the parallel algorithm with different number of threads and variation in the number of algorithms in the composition (CPU)

K	Number of threads			
	2	4	8	16
1	1.0164	1.0069	1.0583	0.9615
2	0.5358	1.9693	1.8658	2.0095
4	1.7794	3.2206	2.8030	2.9603
8	1.9537	3.4683	4.3191	4.0876
16	1.9663	3.2201	5.6927	5.1467
30	1.9819	3.5280	6.9041	6.9989

Table 5 shows the acceleration rates by parallel execution for different numbers of threads and algorithms in the composition when working on the CPU.

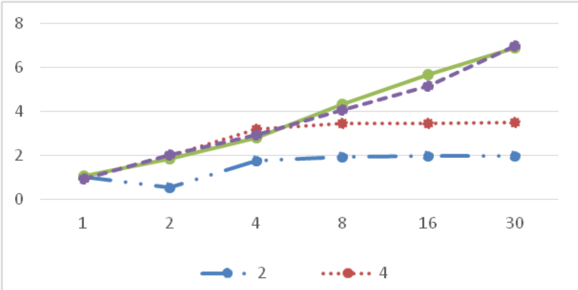


Fig. 4. Visualization of acceleration values for different number of threads and different number of algorithms in the composition (CPU)

From Figures 4, we can see that the value of the acceleration index increases with an increasing number of threads and increasing the number of algorithms in the composition. With a few algorithms, the results are ambiguous and do not reflect a stable and significant increase in acceleration. When a small number of algorithms are used, more time is allocated to the distribution of data between streams, i.e., to parallelization, than to the operation of the algorithms themselves. In addition, the acceleration goes to the number of streams used, which is in line with the basic idea of the acceleration rate. The highest acceleration was recorded when using 16 threads and 30 boosting algorithms in the bagging composition.

Table 6

Indicators of the efficiency of the parallel algorithm with different number of threads and variations in the number of algorithms in the composition

K	Number of threads			
	2	4	8	16
1	0.5082	0.2517	0.1323	0.1202
2	0.2679	0.4923	0.2332	0.2512
4	0.8897	0.8052	0.3504	0.3700
8	0.9769	0.8671	0.5399	0.5112
16	0.9832	0.8050	0.7116	0.6433
30	0.9911	0.8820	0.8630	0.8749

Table 6 shows the performance indicators using parallel execution for different numbers of threads and algorithms in the composition when working on the CPU.

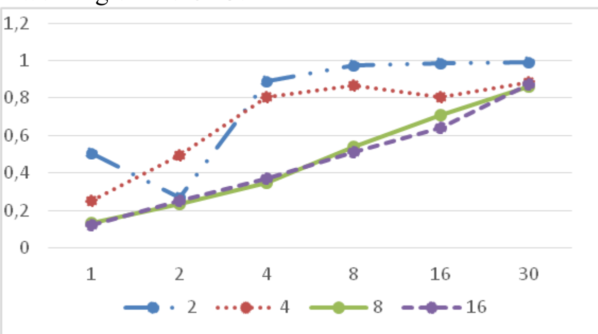


Fig. 5. Visualization of performance indicators for different number of threads and different number of algorithms in the composition

Analyzing Table 6 and Figures 5, we can say that the efficiency decreases with an increasing number of threads in contrast to the acceleration. This decrease in efficiency is due to an increase in the load on the system when calling more threads. However, with the increase in the number of algorithms, the efficiency also increases and, at the same time, goes to 1, which indicates the quality of the parallelization method.

Let's compare the work of parallel execution of the program using the CPU and GPU.

Table 7

Acceleration rates when running a program parallelly using the CPU and GPU

K	CPU	GPU
1	1.0583	9.1641
2	1.8658	6.9279
4	2.8030	8.0452
8	4.3191	8.3861
16	5.6927	8.6379
30	6.9041	9.5151

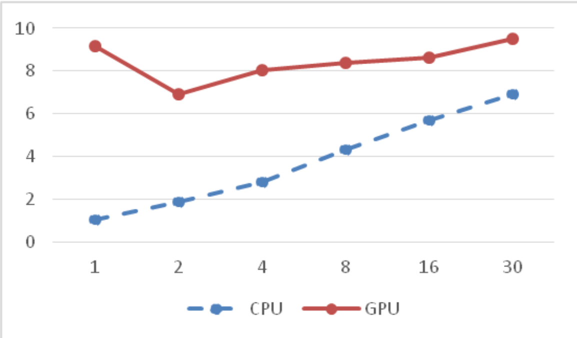


Fig. 6. Graphs of acceleration when using CPU and GPU with different number of algorithms in the composition

From Table 7 and Figure 6 we see that as the number of algorithms in the composition increases, the value of acceleration increases both when working on the CPU and the GPU, but the use of GPU gives higher values of the acceleration index, which indicates the high efficiency of CUDA technology.

Therefore, the results of experiments showed that the parallel implementation of the algorithm is most effective when using graphics processors based on CUDA technology.

Conclusions

Modern technologies can improve the standard of living of humanity in various fields, and medicine is no exception. The paper considered the relevance of the research topic: the use of data mining methods for diagnosing the disease in a patient on a set of indicators, such as symptoms, test results, and more.

A pre-processed Framingham dataset (with the number of patients equal to 10 000) was used for the study. The choice of this data set is primarily related to the need to test the proposed parallel algorithm on big data sets and obtain the best performance indicators. A search and analysis of significant features and patterns between different factors influencing the disease were conducted.

In addition, this paper proposed a parallel Gradient Boosting algorithm with Bagging to improve accuracy in predicting disease risk and significantly speed up execution time by using a multi-core architecture of modern CPUs and GPUs. So, the accuracy of the model with 10 cross validation is equal 99.09%, and for percentage split (75:25) – 99.80%. Parallelization is performed using two technologies – a pool of threads through the Python ThreadPoolExecutor utility on the CPU and CUDA on the GPU. High rates of acceleration and efficiency were achieved. Thus, with eight threads, an acceleration of approximately seven times is obtained, which indicates the reliability of the obtained results and the possibility of significant improvement of this indicator by choosing the architecture of a computer with more cores. The latter is especially relevant in modern trends in multi-core architecture. When using CUDA technology on GPUs, the acceleration is approximately 9.5 times. As for efficiency indicators, with the increase in the number of algorithms, the efficiency goes to 1, indicating the parallelization method's quality.

After analyzing the results, we concluded that CUDA works much more efficiently and prevails over the pool of threads for the selected method and data set. This is because GPUs are designed with thousands of processor cores running simultaneously and thus provide massive parallelism when each core is focused on efficient computing. Another significant result of the study is the achievement of better algorithm accuracy by increasing the number of algorithms in the composition. Using more algorithms that are learned and whose values are then averaged, we get even more accuracy values. Therefore this is a great opportunity to increase the efficiency of the model.

So, using several training algorithms to get the best forecasting efficiency, namely combining Boosting with Begging, is a great solution. This ensemble allows using different methods of accelerating and improving algorithms, which is very important today to make real-time decisions.

References

1. Buch V., Ahmed I., Maruthappu M. Artificial intelligence in medicine: current trends and future possibilities. *The British journal of general practice: the journal of the Royal College of General Practitioners*, 2018; 68(668):143–144.

2. Briganti G., Le Moine O. Artificial Intelligence in Medicine: Today and Tomorrow. *Frontiers in Medicine.* 2020; 7(27):1-6.

3. Chan Y.K., Chen Y.F., Pham T., Chang W., Hsieh M.Y. Artificial Intelligence in Medical Applications. *J Healthc Eng.*, 2018 Jul 15;2018:4827875.

4. Richens J.G., Lee C.M. & Johri S. Improving the accuracy of medical diagnosis with causal machine learning. *Nat Commun*, 2020, 11(3923):1-9.

5. Chary M., Parikh S., Manini A.F., et al. A Review of Natural Language Processing in Medical Education. West J. Emerg. Med. 2019, 20(1):78-86.

6. Le Glaz A, Haralambous Y, Kim-Dufor DH, Lenca P, Billot R, Ryan TC, Marsh J, DeVylder J, Walter M, Berrouiguet S, Lemey C. Machine Learning and Natural Language Processing in Mental Health: Systematic Review. *J. Med. Internet Res.*, 2021 May 4;23(5):e15708.

7. Dash S., Shakyawar S.K., Sharma M., et al. Big data in healthcare: management, analysis and future prospects. *J. Big Data*, 2019, 6(54):1-25.

8. Melnykova N., Shakhovska N., Gregušml M., Melnykov V. Using big data for formalization the patient's personalized data. *Paper presented at the Procedia Computer Science*, 2019, 155:624-629.

9. Mochurad L., Hladun Ya.. Modeling of Psychomotor Reactions of a Person Based on Modification of the Tapping Test. *International Journal of Computing*, 2021, 20(2):190-200.

10. Izonin I., Trostianchyn A., Duriagina Z., et al. The combined use of the wiener polynomial and SVM for material classification task in medical implants production. *International Journal of Intelligent Systems and Applications*, 2018, 10 (9):40-47.

11. Colás A., Vigil L., Vargas B., et al. Detrended Fluctuation Analysis in the prediction of type 2 diabetes mellitus in patients at risk: Model optimization and comparison with other metrics. *PLoS ONE*, 2019, 14(12):1-15.

12. Revathi Addala, Chandra Sekhar Vasamsetty. Diabetes Prediction with improved Data Preprocessing and using Linear Support Vector Classifier. *International Journal of All Research Education and Scientific Methods (IJARESM)*, 2021, 9(6): 33-40.

13. Alsuliman T., Humaidann D., Sliman L. Machine learning and artificial intelligence in the service of medicine: Necessity or potentiality? *Current Research in Translational Medicine*, 2020, 68(4):245-251.

14. Mukherjee S., Yadav G., Kumar R. Recent trends in stem cell-based therapies and applications of artificial intelligence in regenerative medicine. *World journal of stem cells*, 2021, 13(6):521–541.

15. Hussain A., Choi H.E., Kim H.J., et al. Forecast the Exacerbation in Patients of Chronic Obstructive Pulmonary Disease with Clinical Indicators Using Machine Learning Techniques. *Diagnostics (Basel, Switzerland)*, 2021, 11(5):829.

16. Melnykova N., Shakhovska N., Gregus M., et. al. Data-driven analytics for personalized medical decision making. *Mathematics*, 2020, 8(8):1211.

17. Taser P.Y. Application of Bagging and Boosting Approaches Using Decision Tree-Based Algorithms in Diabetes Risk Prediction. *Proceedings of the 7th International Management Information Systems Conference*, 2021, 74(1):6.

18. Diabetes Database. Available online: https://www.kaggle.com/uciml/pima-indians-diabetes-database (accessed on 9 April 2023).

19. Dataset Framingham. Available online: https://www.researchgate.net/publication/ 366485790_Diabetes (accessed on 9 April 2023).

20. Sharma, T., Shah, M. A comprehensive review of machine learning techniques on diabetes detection. *Vis. Comput. Ind. Biomed.*, 2021, 4(30):1-16.

21. Beschi R., Anitha R., Sujatha R., et al. Diabetics Prediction using Gradient Boosted Classifier. *International Journal of Engineering and Advanced Technology (IJEAT),* 2019, 9(1): 3181-3183.

22. Swapna G., Vinayakumar R. and Soman K. Diabetes detection using deep learning algorithms. ICT Express. 2018, 4(4): 243-246.

23. Peng B., et al. HarpGBDT: Optimizing Gradient Boosting Decision Tree for Parallel Efficiency. 2019 *IEEE International Conference on Cluster Computing (CLUSTER),* 2019:1-11.

24. Yu C., Skillicorn D.B. Parallelizing boosting and bagging. Queen's University, Kingston, Canada, *Tech. Rep.*; 2001, 22 p.

25. Arsov N., Pavlovski M., Basnarkov L., Kocarev L. Generating highly accurate prediction hypotheses through collaborative ensemble learning. *Sci Rep.,* 2017, 7:44649.

26. Yurttakal A.H., Baş H. Possibility Prediction Of Diabetes Mellitus At Early Stage Via Stacked Ensemble Deep Neural Network. *Afyon Kocatepe Üniversitesi Fen Ve Mühendislik Bilimleri Dergisi*, 2021, 21(4):812-819.

27. González S., García S., Del Ser J., Rokach L., Herrera F. A practical tutorial on bagging and boosting based ensembles for machine learning: Algorithms, software tools, performance study, practical perspectives and opportunities, *Information Fusion*, 2020, 64:205-237.

28. Ensemble methods: bagging, boosting and stacking. Available from: https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205 (accessed on 9 April 2023).

29. Mochurad L, Hladun Y, Tkachenko R. An Obstacle-Finding Approach for Autonomous Mobile Robots Using 2D LiDAR Data. *Big Data and Cognitive Computing*, 2023; 7(1):43.

30. Mochurad, L., Kryvinska, N. Parallelization of Finding the Current Coordinates of the Lidar Based on the Genetic Algorithm and OpenMP Technology. *Symmetry*, 2021, 13(666).

31. Felipe A. Quezada, Cristobal A. Navarro, Miguel Romero, Cristhian Aguilera. Modeling GPU Dynamic Parallelism for self similar density workloads. *Future Generation Computer Systems*, 2023, 145:239-253.

32. Shi Yu et al. Quantized Training of Gradient Boosting Decision Trees. *36th Conference on Neural Information Processing Systems (NeurIPS 2022)*, 2022:1-24, https://arxiv.org/pdf/2207.09682.pdf.

Lesia Mochurad **Леся Мочурад**	PhD, Associate Professor of Department of Artificial Intelligence, Lviv Polytechnic National University, Lviv, Ukraine, e-mail: lesia.i.mochurad@lpnu.ua. https://orcid.org/0000-0002-4957-1512 Scopus Author ID: 57210286606, ResearcherID: AAZ-9150-2020	кандидат технічних наук, доцент, доцент кафедри систем штучного інтелекту національного університету "Львівська політехніка", Львів, Україна.

https://doi.org/10.31891/csit-2023-2-2
UDC 004.912

Olesia BARKOVSKA, Anton HAVRASHENKO
Kharkiv National University of Radio Electronics, Ukraine

ANALYSIS OF WORD SEARCH ALGORITHMS IN THE DICTIONARIES OF MACHINE TRANSLATION SYSTEMS FOR ARTIFICIAL LANGUAGES

The main goal of the project is the analysis of word search algorithms in the dictionary for machine translation systems. To achieve the goal, the following tasks were solved:

- analysis of shortcomings of the proposed and developed artificial language machine translation system;

- improvement of the dictionaries included in the proposed system of machine translation of artificial languages, thanks to the algorithms of searching for words of n-grams and Knuth-Morice-Pratt;

- implementation of the possibility of using the prepared dictionary for translation;

- analysis of the impact of implemented improvements to word search methods on search accuracy, choice of dictionary structure, and search time.

The paper is devoted to the development of an organizational model of the machine translation system of artificial languages. The main goal is the analysis of word search algorithms in the dictionary, which are significant elements of the developed machine translation system at the stage of improvement of new dictionaries created on the basis of already existing dictionaries. In the course of the work was developed a model of the machine translation system, created dictionaries based on texts and based on already existing dictionaries using augmentation methods such as back translation and crossover; improved dictionary based on algorithms of n-grams, Knuth-Morris-Pratt and word search in the text (such as binary search, tree search, root decomposition search). In addition, the work implements the possibility of using the prepared dictionary for translation. The obtained results can improve existing systems of machine translation of the text of artificial languages. Namely, to reduce the operating time by approximately 20 times when switching from the balanced tree algorithm to other logarithmic algorithms. The practical significance of this work is the analysis and improvement of text augmentation algorithms using algorithms of binary search, hashes, search tree, and root decomposition.

Keywords: translation, prefix tree, dictionary, artificial language, hash, binary search, search tree

Олеся БАРКОВСЬКА, Антон ГАВРАШЕНКО
Харківський національний університет радіоелектроніки

АНАЛІЗ АЛГОРИТМІВ ПОШУКУ СЛІВ ДЛЯ ЗАСТОСУВАННЯ В СИСТЕМАХ МАШИННОГО ПЕРЕКЛАДУ ШТУЧНИХ МОВ

Основною метою проекту є аналіз алгоритмів пошуку слів у словнику для систем машинного перекладу. Для досягнення мети були вирішені наступні завдання:

- аналіз недоліків запропонованої та розробленої системи машинного перекладу штучної мови;

- удосконалення словників, що входять до складу запропонованої системи машинного перекладу штучних мов, завдяки алгоритмам пошуку слів n-грамм та Кнута-Моріса-Пратта;

- реалізація можливості використання підготовленого словника для перекладу;

- аналіз впливу впроваджених удосконалень методів пошуку слів на точність пошуку, вибір структури словника та час пошуку.

Робота присвячена розробці організаційної моделі системи машинного перекладу штучних мов.Головною метою є аналіз алгоритмів пошуку слова в словник, які є значущими елементами розробленої системи машинного перекладу на етапі вдосконалення створених нових словників на основі вже існуючих словників. В ході виконання роботи була розроблена модель системи машинного перекладу, створені словники на основі текстів та на основі вже існуючих словників методами аугментації такими, як зворотній переклад та кросовер; вдосконалено створений словник на основі алгоритмів n-грамм, Кнута-Моріса-Пратта та пошуку слів у тексті (таких, як бінарний пошук, пошук в дереві, пошук в кореневій декомпозиції). Окрім того, в роботі реалізована можливість використання підготовленого словнику для перекладу. Отримані результати можуть покращити існуючі системи машинного перекладу тексту штучних мов. А саме – зменшити час роботи приблизно у 20 разів при переході від алгоритму збалансованого дерева до інших логарифмічних алгоритмів. Практичною значущістю даної роботи є аналіз та покращення алгоритмів аугментації тексту за допомогою алгоритмів бінарного пошуку, хешів, дерева пошуку, кореневої декомпозиції.

Ключові слова: переклад, префіксне дерево, словник, штучна мова, хеш, бінарний пошук, дерево пошуку

Introduction

Machine translation (MT) – technology of automated translation of texts (written and spoken) from one natural language to another using a computer; the direction of scientific research related to the construction of automated translation systems [1,3].

At a basic level, the job of computer translation programs is to replace words or phrases from one language with words or phrases from another. However, the problem arises that such a replacement cannot provide a high-quality translation of the text, because it requires the definition and recognition of words and entire phrases from the original language. This encourages active researches in the field of computational linguistics.

Machine translation is one of the subgroups of computational linguistics that studies the use of software to translate text from one language to another [4,6]. At the initial level, MT performs the usual replacement of words from one language for words from another language, but usually the translation made in this way is not very high-quality, because in order to fully convey the meaning of the sentence and find the closest analogue in the "target" language) – to the language required by the translator, it is often necessary to translate an entire phrase. Commonly machine translation divided on 4 methods (Figure 1).

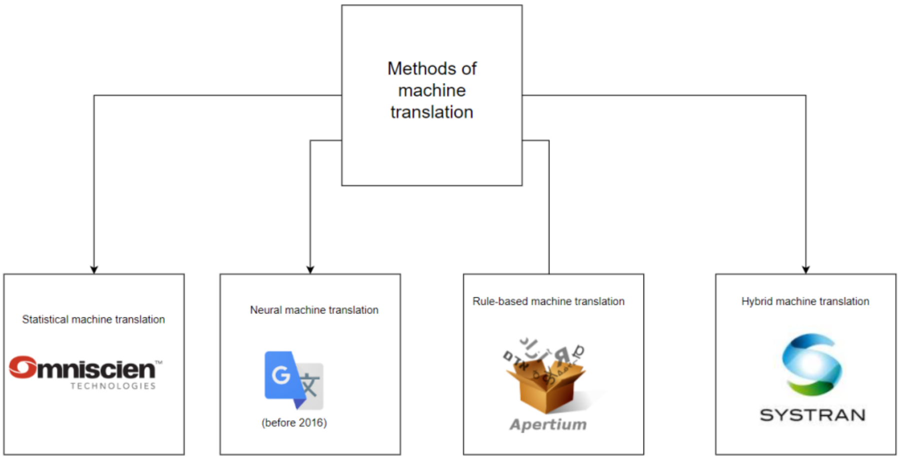


Fig. 1. Methods of machine translations

Let's take a closer look at statistical machine translation.

Statistical machine translation is a type of machine translation of text based on the comparison of large volumes of language pairs. Language pairs - texts containing sentences in one language and corresponding sentences in another can be both variants of writing two sentences by a person - a native speaker of two languages - and a set of sentences and their translations performed by a person. Thus, statistical machine translation has the property of "self-learning". The more language pairs the program has at its disposal and the more precisely they correspond to each other, the better the result of statistical machine translation [7,9].

Table 1

Comparison of translation methods

Criterion for comparison	Statistical machine translation	Neural machine translation	Rule-based machine translation	Hybrid machine translation
Translation speed	It takes the least amount of time	It takes a lot of time	It takes a lot of time	It takes the most time
Translation quality	High	The best	High	Quite high
Extensibility	There is a possibility of expansion	There is a possibility of expansion	There is a possibility of expansion, but it is more difficult than in other algorithms	There is a possibility of expansion, but it is more difficult than in other algorithms

The first ideas of statistical machine translation were presented by Warren Weaver in 1949, including the ideas of information theory applications of Claude Shannon. Statistical machine translation was reintroduced in the late 1980s and early 1990s by researchers at IBM's Thomas J. Watson Research Center and has contributed to a significant resurgence of interest in machine translation in recent years. Before the introduction of neural machine translation, it was by far the most researched method of machine translation [10,12].

In verbal translation, the basic unit of translation is a word of a certain natural language. As a rule, the number of words in translated sentences varies due to complex words, morphology and idioms. The ratio of the length of sequences of translated words is called fertility, which shows how many foreign words each native word creates. Information theory necessarily assumes that everyone embraces the same concept. In practice, this is not quite the case. For example, the English word corner can be translated into Spanish as rincón or esquina, depending on whether it means inside or outside corner.

We compare all methods according to such parameters as translation speed, quality and extensibility (Table 1).

After the comparison, we can conclude that each of the types of machine translation can be supported, but the best are neural and hybrid machine translation, which are currently used most often.

Related works

An example of a word-based translation system (Figure 2) is the freely available package GIZA++ (GPLed), which includes a tutorial for IBM models, HMM models, and Model 6 [13].

Verbal translation is not widely used today; phrasal systems are more common. Most phrase-based systems still use GIZA++ for corpus matching. Alignment is used to highlight phrases or derive syntactic rules. And matching words in double text is still a hotly debated issue in the community. Due to the prevalence of GIZA++, there are now several distributed implementations of it online.

In phrase-based translation, the goal is to reduce the limitations of word-based translation by translating whole sequences of words where the length can vary. Sequences of words are called blocks or phrases, but usually they are not linguistic phrases, but phrases found using statistical methods from corpora. It is shown that restricting phrases to language phrases (syntactically motivated groups of words) reduces the quality of translation.[14].

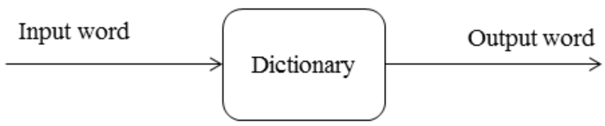


Fig.2. Diagram of the translation system based on words

The selected phrases are then displayed next to each other based on the phrase translation table and can be changed. This table can be learned from word alignment or directly from a parallel corpus.

Syntax-based translation is based on the idea of translating syntactic units rather than individual words or word strings (as in phrase-based MT), i.e. (partial) analysis of sentence/utterance trees. The idea of syntax-based translation is quite old in MT, although its statistical counterpart did not become widespread until the advent of strong stochastic parsers in the 1990s. Examples of this approach include DOP-based MT and, more recently, synchronous context-free grammars. Syntactic unit is always a combination that has at least two constituents (Figure 3). The basic syntactic units are a word-group, a clause, a sentence, and a text [15]. Their main features are:

-they are hierarchical units – the units of a lower level serve the building material for the units of a higher level;

-as all language units the syntactic units are of two-fold nature;

-they are of communicative and non-communicative nature – word-groups and clauses are of non-communicative nature while sentences and texts are of communicative nature.

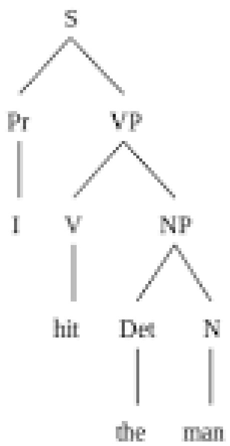


Fig.3. Example of syntactic units

Another example of a statistical machine translator is Google Translate from 2006 to 2016.

Google Translate did not apply grammar rules because its algorithms were based on statistical or pattern analysis rather than traditional rule-based analysis. The system's creator, Franz Josef Och, criticized the effectiveness of rule-based algorithms in favor of statistical approaches. The original versions of Google Translate were based on the statistical machine translation method, or rather, on the research of Och, who won the DARPA competition for high-speed machine translation in 2003. Och was the head of Google's machine translation team.

A solid foundation for developing a usable statistical machine translation system for a new language pair from scratch would consist of a bilingual text corpus (or parallel collection) of over 150-200 million words and two monolingual corpora of over a billion words each. Statistical models from this data are then used to translate between these languages.

To get such a huge amount of linguistic data, Google used documents and transcripts of the United Nations (UN) and the European Parliament. The UN usually publishes documents in all six official UN languages, creating a very large 6-language corpus.

Google representatives participated in internal conferences in Japan where they requested bilingual data from researchers (Figure 4).

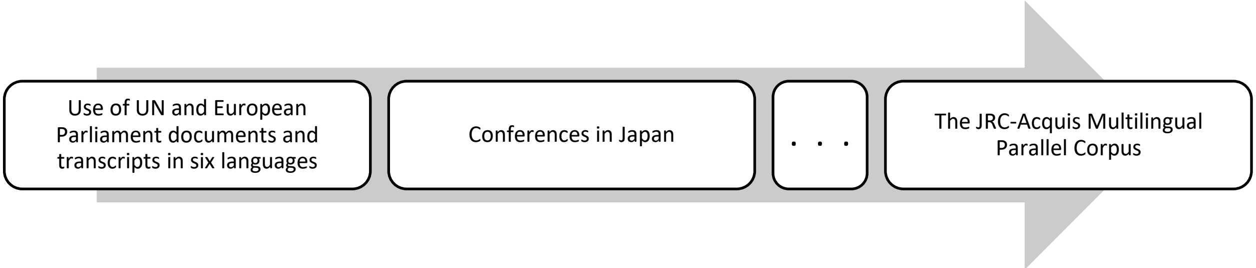


Fig.4. Chronology of the organization of linguistic corpora for the operation of the Google Translate system

When Google Translate generates a translation suggestion, it looks for patterns in hundreds of millions of documents to help choose the best translation. By detecting patterns in documents that have already been translated by translators, Google Translate makes educated guesses (AI) about what the correct translation should be.

Until October 2007, for languages other than Arabic, Chinese, and Russian, Google Translate was based on SYSTRAN, a software engine still used by several other online translation services, such as Babel Fish (now defunct). Since October 2007, Google Translate has used its own technology based on statistical machine translation, and then switched to neural machine translation.

Also, Google Translate did not have the ability to translate directly from certain languages. For the translation of some languages, translations were made into English, which were then translated into the required language. For example, for a translation from Ukrainian, the text was translated into Russian, after which it was translated into English. After these translations, the obtained result was translated into the required language.

In systems that use dictionaries, searching for words is an actual problem, because depending on the selected search algorithm, it will be necessary to change the type of dictionary that will store words. For example, binary search or two pointers cannot be used on an unordered dictionary. We compare the complexity (in Big O notation) of algorithms on different operations with dictionaries (Table 2).

Table 2

Comparison of methods in Big O notation

	Binary search	**Hash table**	**Binary tree**	**Prefix tree**	**2 instructions**	**Sqrt decomposition**	**Normal search**
Creating a dictionary	*NlogN*	*N*M*	*NMlogN*	*N*M*	*NlogN*	*NlogN*	*N*
Combining two dictionaries	*N*	*P*	*NMlogN*	*M*K*	*N*	*N*	*N*
Adding a new word	*N*	*M*	*MlogN*	*M*	*N*	*N*	*1*
Word search	*logN*	*1*	*MlogN*	*M*	*N*	*Sqrt(N)*	*N*
Memory used	*N*M*	*N*M+P*	*N*M*	*M*K*	*N*M*	*(N +sqrt(N)) *M*	*N * M*

Where *N* is the number of words in the dictionary, *M* is the average word length, *P* is the hashing module .

As we can see, each of the algorithms has its advantages and disadvantages. If you need to make an application as quickly as possible, and the running time or occupied memory are not important, then it is best to use a balanced tree, because it is usually already implemented in programming languages. For example, std::map in C++ or dictionary in Python.

For fast search if there are no memory limitations, we can use hashing combined with binary search, which will give a good performance boost.

If you need to find words often, and we can afford to use a complex structure, it is recommended to use the Bohr method, which will allow you to quickly add words and quickly find them in a time close to the length of a word. But when using a large alphabet, the method loses its power.

Main goal and tasks of the research

The main goal of the project is the analysis of word search algorithms in the dictionary for machine translation systems.

To achieve the goal, the following tasks must be solved:

- analysis of shortcomings of the proposed and developed artificial language machine translation system [16];
- improvement of the dictionaries included in the proposed system of machine translation of artificial languages, thanks to the algorithms of searching for words of n-grams and Knuth-Morice-Pratt[17];
- implementation of the possibility of using the prepared dictionary for translation [18];
- analysis of the impact of implemented improvements to word search methods on search accuracy, choice of dictionary structure, and search time.

Experiments

Detailed model of the proposed machine translation system , which is the basis for conducting research and analyzing the impact of data volume on runtime, was proposed in [19].

The system[20] model includes the following modules: a module for generating dictionaries based on input texts, a module for generating dictionaries based on dictionaries of other languages; and a translator program for translating texts with the help of dictionaries(Figure 5).

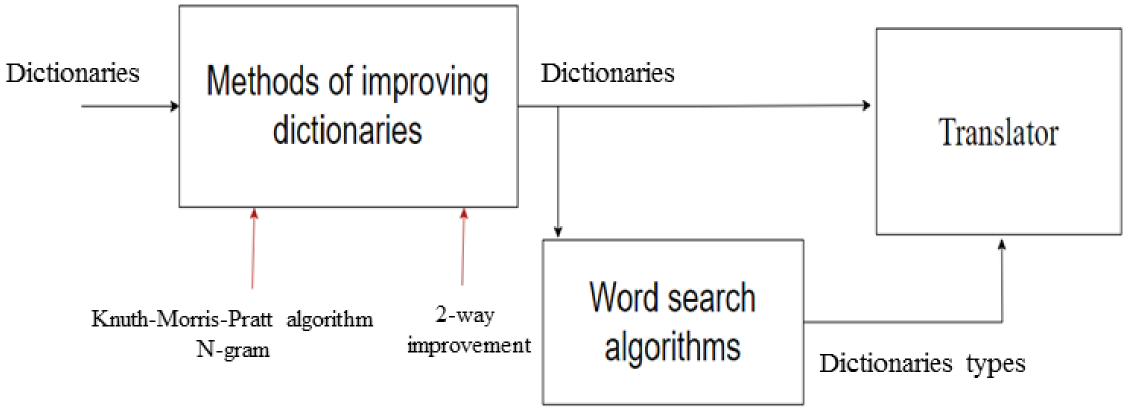


Fig.5. Interaction of components

For a translator to work quickly, you need to be able to quickly find a word in the dictionary. Consider such search methods as binary search, prefix tree, two pointers, balanced tree, root decomposition, and full traversal.

Since our dictionaries are sorted, we can apply the binary search method quite easily.

Binary search is an algorithm for finding a given value in an ordered array, which consists in comparing the middle element of the array with the desired value, and repeating the algorithm for one or the other half, depending on the result of the comparison.

This method will help us find the necessary information quite quickly, but will not provide additional information that can be used later.

After the experiments, the following results were obtained (Table 3). We can see that the time increases proportionally with a large amount of input data, and the algorithm is quite efficient.

Table 3

Results of the running time of the binary search algorithm

Size of input data (number of words)	10^6	3*10^6	10^7	3*10^7	10^8
Operating time (seconds)	0	1	5	11	41

For the two-pointer method, you need to use the following idea. For each pair of words, you need to make a query from the second dictionary. Since the dictionary is already sorted, for a quick search for words, you need to sort the queries and then use the following idea.

Since the words are sorted, if one word in the list is lexicographically larger than the second word in another list, then all the words following it are also larger than the selected word. Therefore, if we put pointers at the beginning of both lists, and move the one that is lexicographically smaller, we will not miss a single pair of words. Since each word occurs only once in the dictionary, and not always in the queries, if the words are equal, the pointer will move in the query list.

After the experiments, the following results were obtained (Table 4). From this experiment, one can see that the time increases proportionally with a large amount of input data, and the algorithm is quite efficient.

Table 4

Results of the running time of the algorithm of two pointers

Size of input data (number of words)	10^6	3*10^6	10^7	3*10^7	10^8
Operating time (seconds)	0	1	5	17	63

However, with a large number of words, the algorithm slightly loses to binary search. This is possibly related to the implementation of the algorithm, because with large data, even a small constant in the complexity can increase the real time of the algorithm many times, although the complexity of both is the same.

For the next experiment, we used the Prefix tree method. The use of the Prefix tree method was described in detail in [19] , so in this work we will only use the obtained results for comparison (Table 5).

Table 5

Results of the running time of the improved prefix tree

Size of input data (number of words)	10^6	3*10^6	10^7	3*10^7	10^8
Operating time (seconds)	0	1	3	10	33

Hashing algorithm is the next researched algorithm. For this algorithm, we will generate a hash function from each line, and distribute all the words by it, after which the obtained results will be searched by binary search. However, for our testing, let's eliminate the second part and just look at the speed of hashing for our strings.

Table 6

Results of the running time of the hashing algorithm

Size of input data (number of words)	10^6	3*10^6	10^7	3*10^7	10^8
Operating time (seconds)	0	0	0	2	6

From this experiment, we can see that hashing takes quite a bit of time, but it cannot be the only method of finding words, since two different strings can have the same hashes, and there will be false positives. This algorithm will allow you to mix the search area, but will increase the required amount of memory, which can be critical with a large amount of input data, so it should be used only when necessary (Table 6).

A balanced tree in the general sense of the word is a type of binary search tree that automatically maintains its height, that is, the number of levels of vertices under the root is minimal.

For this algorithm, we will use a modification of the usual map, where the keys will be our lines.

After the experiments, the following results were obtained (Table 7).

Table 7

Results of the running time of the balanced tree algorithm

Size of input data (number of words)	10^6	3*10^6	10^7	3*10^7	10^8
Operating time (seconds)	2	11	40	153	579

From this experiment, we can see that the time with a large amount of input data is very high compared to other algorithms. Since the algorithm has the same O(N*logN) complexity as the others, this is a rather unexpected result. Since we use a string as a key, their comparison takes quite a lot of time, which leads to such time consumption. However, can we say that the algorithm is not working at all? To do this, let's compare it with those that will have a worse difficulty, and compare it with them.

For the next algorithm, we used SQRT decomposition. Unlike the previous ones, it will have worse complexity. We will use the sqrt decomposition algorithm. To do this, in the sorted array, we will select the words that are in the positions $0, \sqrt{n}, 2\sqrt{n}, .., n$ containing approximately the root of n words, where n is the number of words in the list. Since the entire list has been sorted, the list selected in this way will also be sorted. Then, after comparing the input word with each of the words in the list, we can find the root of the length of the list of comparisons, on which the searched word can be found. Since we chose words with a different root between them, the length of the found subsegment will be equal to the root of n. Thus, for two roots of the length of the list of comparisons, we can find our word. After the experiments, the results were obtained (Table 8).

Table 8

Results of the running time of the sqrt decomposition algorithm

Size of input data (number of words)	10^6	3*10^6	10^7	3*10^7	10^8
Operating time (seconds)	45	225	1464	9516*	60000*

Since with the input data size of 10^7 words, the working time approaches 25 minutes, it was decided not to conduct an experiment for larger data, but to find the approximate working time of the algorithm. As you can see, the last experiment will take more than 16 hours of time, which is certainly not possible within the scope of this research.

It is clear that within the framework of this problem, the first thing you can do is simply to fulfill the condition, and for each of the words iterate over every other word – it is so called full search.. Since it will be quite long, instead of conducting an experiment, we will calculate the expected time of the algorithm. As a basis, we will take the operating time of the root decomposition algorithm, since it is the closest to it, and multiply it by the root of the size of the input data.

This is possible because the real time of the algorithm is equal to the number of commands that the algorithm will execute, divided by the frequency of the processor, or multiplied by the execution time of one command. Since we are calculating for the same processor, when one is divided by another, this value will decrease, and the ratio of the number of commands will be proportional to the complexity. Since $n^2 / n\sqrt{n} = \sqrt{n}$, we can multiply the root decomposition result by the root of the input data. After the calculations, the following results were obtained (Table 9).

Table 9

Results of the running time of the quadratic search algorithm

Size of input data (number of words)	10^6	3*10^6	10^7	3*10^7	10^8
Working hours (hours)	12.5	108	1286	14478	19 years old

From this experiment we can see that in order to process a large amount of data, we need to use at least some algorithms that will improve the usual choice of correspondence for each of the words
To compare the algorithms, we will summarize all the data obtained during the experiments into a common table (Table 10, Figures 6,7).

Table 10

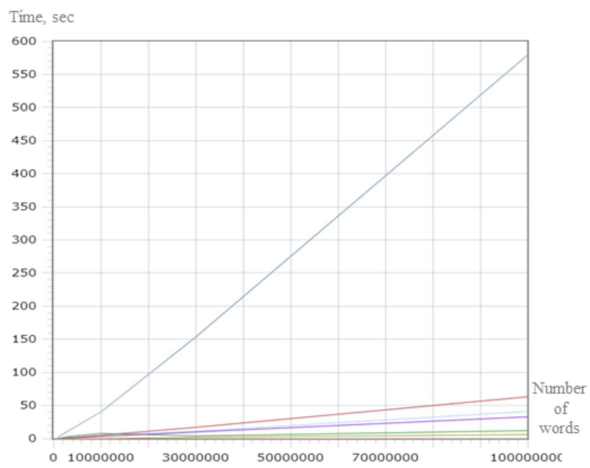
Comparison of algorithms

Size of input data (number of words)	10^6	3*10^6	10^7	3*10^7	10^8
Binary search	0	1	5	11	41
Two pointers	0	1	5	17	63
Prefix tree	1	3	8	4*	12*
Improved Prefix tree	0	1	3	10	33
Hashing	0	0	0	2	6
A balanced tree	2	11	40	153	579
Root decomposition	45	225	1464	9516**	60000**
Full search	12.5 hours **	108 hours**	1286h**	14478 hours **	19 years old **

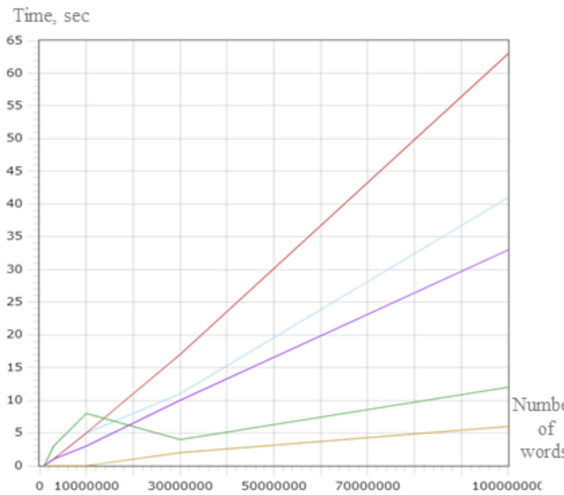
*-The data was executed on a tree built on 10^6 words instead of the required ones.
**-Estimated expected working time.

After all experiments, it is clear that fast algorithms must be used for big data. The best choice would be either an improved boric, which works better with small alphabets, or a binary search, because they are fast and don't have the problem of false positives. The worst among the logarithmic algorithms is the balanced tree, so it is possible to use any other logarithmic algorithm depending on other factors, such as reliability, memory, and dictionary structure.

Graphs on Figure 6a and 6b show the runtime dependency on number of words in the dictionary. On Figure 6a one can see grey line, that corresponds to balanced tree algorithm. This graph was added to make the result obtained more demonstrative. But since it is difficult to analyze other results, Figure 6b shows the same dependencies, but without the balanced tree algorithm.



a)



b)

Fig.6. Comparative graph of the runtime dependency on number of words in the dictionary for logarithmic algorithms: two pointers algorithm – red line; binary search algorithm- blue line; improved prefix tree algorithm – violet line; hashing algorithm- green line; prefix tree algorithm- brown line; balanced tree - grey line.

Conclusions

In the course of the work, an analysis of the machine text translation system for artificial languages and the augmentation methods used in it was carried out.

The method of storing dictionaries was improved using the following methods: storing words in the order in which they are presented; storing words in alphabetical order; balanced tree; hash table; root decomposition; prefix tree.

In addition, the methods of searching for elements in the dictionary were analyzed, such as quadratic search, root decomposition, prefix tree, two pointers, binary search and hashing. As a result, hashing, prefix tree and binary search showed the best result. Each of which has its own drawbacks, and we cannot clearly indicate the best method. The use of these methods made it possible to increase the speed of work on a large amount of data compared to a conventional search from several years to tens of seconds. This made it possible to use the translator on large volumes of text. When switching from the balanced tree algorithm to other logarithmic algorithms, the running time decreased by about 20 times.

The model has quite a high potential for use in cases where there are no conventional translators. With a growing number of languages, this problem may worsen, as the number of translators cannot grow at the same rate.

References

1. K. Manuel, KV Indukuri and PR Krishna, "Analyzing Internet Slang for Sentiment Mining," 2010 Second Vaagdevi International Conference on Information Technology for Real World Problems, 2010, pp. 9-11, doi: 10.1109/VCON.2010.9

2. F. Ren and K. Matsumoto, "Semi-Automatic Creation of Youth Slang Corpus and Its Application to Affective Computing," in IEEE Transactions on Affective Computing, vol. 7, no. 2, pp. 176-189, 1 April-June 2016, doi: 10.1109/TAFFC.2015.2457915

3. Karen S. Jones. Natural language processing: a historical review //Cambridge: Computer Laboratory, University of Cambridge, 2001

4.V. Ryzhkova, "Possibilities of Computer Lexicography in Compiling Highly Specialized Terminological Printed and Electronic Dictionaries (Field of Aviation Engineering)," 2020 Ivannikov Memorial Workshop (IVMEM), 2020, pp. 40-42, doi: 10.1109/IVMEM51402.2020.00013

5. B. Ranaivo-Malançon, S. Saee and JF Wilfred Busu, "Discovering linguistic knowledge by converting printed dictionaries of minority languages into machine readable dictionaries," 2014 International Conference on Asian Language Processing (IALP), 2014, pp. 140-143, doi: 10.1109/IALP.2014.6973522.

6. GR Chumarina. Classification of electronic dictionaries in modern lexicography and lexicologists and features of their use [Electronic resource] Baltic Humanitarian Journal. 2013. No. 4. P.123-126

7. Deng, Yonggang, and William Byrne. "HMM word and phrase alignment for statistical machine translation." IEEE Transactions on Audio, Speech, and Language Processing 16.3 (2008): 494-507.

8. .Hey, Xiaodong. "Using word-dependent transition models in HMM-based word alignment for statistical machine translation." Proceedings of the Second Workshop on Statistical Machine Translation . 2007.

9..Oh, Franz Josef. Statistical machine translation: From single word models to alignment templates . Diss. Aachen, Techn. Hochsch., Diss., 2002, 2003

10. Östling, Robert, and Jörg Tiedemann. "Efficient word alignment with markov chain monte carlo." The Prague Bulletin of Mathematical Linguistics (2016).

11. Moore, Robert C. "A discriminative framework for bilingual word alignment." Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing. 2005.

12.Pitrelli, John F., et al. "The IBM expressive text-to-speech synthesis system for American English." IEEE Transactions on Audio, Speech, and Language Processing 14.4 (2006): 1099-1108.

13. Casacuberta, Francisco, and Enrique Vidal. "GIZA++: Training of statistical translation models." Retrieved October 29 (2007): 2019.

14. Olesia Barkovska , Oleg Mikhal , Daria Pyvovarova , Oleksii Liashenko , Vladyslav Diachenko , Maxim Volk . Local Concurrency in Text Block Search Tasks. // International Journal of Emerging Trends in Engineering Research. - Volume 8. No. 3, March 2020. – P.690-694. DOI: 10.30534/ijeter/2020/13832020

15.Croft, William. "Intonation units and grammatical structure." (1995): 839-882.

16. Kazakov D. Artificial naturalness // Science and life. – 2017. – No. 10. – P. 100–107.

17.D. Anggreani, DPI Putri, AN Handayani and H. Azis, "Knuth Morris Pratt Algorithm in Enrekang-Indonesian Language Translator," 2020 4th International Conference on Vocational Education and Training (ICOVET), 2020, pp. 144-148, doi: 10.1109 / ICOVET50258.2020.9230139.

18..Barkovska O., Pyvovarova D. and Serdechnyi V., Pryskorenyj alghorytm poshuku sliv-obraziv u teksti z adaptyvnoju dekompozycijeju vykhidnykh danykh. [Accelerated word-image search algorithm in text with adaptive decomposition of input data]. Systemy upravlinnja, navighaciji ta zv'jazku 4 (56), 28-34. (in Ukrainian) DOI: https://doi.org/10.26906/SUNZ.2019.4.028

19. Havrashenko A., Barkovska O. ANALYSIS OF TEXT AUGMENTATION ALGORITHMS IN ARTIFICIAL LANGUAGE MACHINE TRANSLATION SYSTEMS //Advanced Information Systems. – 2023. – No. 7. - no. 1. – pp . 47-53. DOI: 10.20998/2522-9052.2023.1.08

20. Olesia Barkovska, Anton Havrashenko, Vladyslav Kholiev, Olena Sevostianova. Automatic text translation system for artificial languages. // Computer systems and information technologies. – 2021. – no. 3. – p . 21-30. DOI: 10.31891/CSIT-2021-5-3

https://doi.org/10.31891/csit-2023-2-3
UDC 681.32

Yaroslav NYKOLAYCHUK
West Ukrainian National University
Volodymyr HRYHA
Vasyl Stefanyk Precarpathian National University
Nataliia VOZNA, Ihor PITUKH
West Ukrainian National University
Lyudmila HRYHA
Nadvirna Vocational College by NTU

HIGH-PERFORMANCE COMPONENTS OF HARDWARE MULTI-BIT SPECIFIC PROCESSORS FOR THE ADDITION AND MULTIPLICATION OF BINARY NUMBERS

The relevance of solving the priority improvement problem for functional and computational components of microelectronics for arithmetic and logic unit (ALU) of modern supercomputers is emphasized. It is shown that in the structures of ALU operations such as: comparison, multiplexing and adding affect the marginal productivity of ALU. It is substantiated that mathematical operations of addition, accumulation of sums, multiplication and division are crucial for ensuring extremely high speed of vector and scalar computers. The purpose of the work is the development of new calculation algorithms and microelectronic structures of multi-bit ALU coprocessors, which are characterized by extremely minimal parameters of speed, hardware and structural complexity. Such high-performance coprocessors are widely used as components of ALU when performing algorithmically complex calculations of statistical, correlation, spectral, cluster and entropy analysis. High-speed co-processors for multiplication and accumulation of digital data with the properties of crypto-protection of telecommunication channels are effectively used in the conditions of military operations and the modern information front, for example, in unmanned aerial vehicles, ground launchers and processors of air defense systems. The main areas for improving system characteristics of microelectronic components of the ALU processors of vector and scalar supercomputers are outlined. Structures of combinational and synchronized single-bit binary adders are systematized due to the characteristics of minimum hardware complexity and maximum speed. The theoretical foundations of double binary arithmetic are outlined. A new structure of a high-performance matrix multiplier based on synchronized adder-accumulators and Booth's algorithm is proposed, which is characterized by increased speed compared to known structures. Grounded perspective directions for further development improvement of the basic characteristics of the investigated class of computing equipment components.

Key words: structures, ALU, microelectronics, matrix multiplier, Booth's algorithm, supercomputers

Ярослав НИКОЛАЙЧУК
Західноукраїнський національний університет
Володимир ГРИГА
Прикарпатський національний університет імені Василя Стефаника
Наталія ВОЗНА, Ігор ПІТУХ
Західноукраїнський національний університет
Людмила ГРИГА
ВСП "Надвірнянський фаховий коледж"

ВИСОКОПРОДУКТИВНІ КОМПОНЕНТИ АПАРАТНИХ БАГАТОРОЗРЯДНИХ СПЕЦПРОЦЕСОРІВ СУМУВАННЯ ТА ПЕРЕМНОЖЕННЯ ДВІЙКОВИХ ЧИСЕЛ

Акцентована актуальність вирішення проблеми пріоритетного удосконалення функціонально-обчислювальних компонентів мікроелектроніки арифметико-логічних пристроїв (АЛП) сучасних суперкомп'ютерів. Показано, що у структурах АЛП операції типу: порівняння, мультиплексування та додавання впливає на граничну продуктивність АЛП. Обгрунтовано, що математичні операції додавання, накопичення сум, перемноження та ділення є визначальними для забезпечення гранично високої швидкодії векторних та скалярних комп'ютерів. Метою роботи є розробка нових алгоритмів обчислень та мікроелектронних структур співпроцесорів багаторозрядних АЛП, які характеризуються гранично мінімаксними параметрами швидкодії, апаратної та структурної складності. Такі високопродуктивні співпроцесори широко застосовуються у якості компонентів АЛП при реалізації алгоритмічно-складних обчислень статистичного, кореляційного, спектрального, кластерного та ентропійного аналізу. Максимально-швидкодіючі співпроцесори перемноження та накопичення цифрових даних з властивостями криптозахисту телекомунікаційних каналів, ефективно застосовуються в умовах військових операцій та сучасного інформаційного фронту. Наприклад, у безпілотних літальних апаратів, наземних пускових установках та процесорах систем протиповітряної оборони. Викладені пріоритетні напрямки покращення системних характеристик мікроелектронних компонентів процесорів АЛП векторних та скалярних суперкомп'ютерів. Систематизовані структури комбінаційних та синхронізованих однорозрядних двійкових суматорів з характеристиками мінімальної апаратної складності та максимальної швидкодії. Викладені теоретичні основи бінарної двійкової арифметики. Запропонована нова структура високопродуктивного матричного перемножувача згідно алгоритму Бута, який характеризується підвищеною швидкодією у порівнянні з відомими структурами. Обгрунтовані перспективні напрямки подальшого розвитку та покращення базових характеристик досліджуваного класу компонентів обчислювальної техніки.

Key words: структура, АЛП, мікроелектроніка, матричний перемножувч, алгоритм Бута, суперкомп'ютер

Introduction

One of the main and relevant directions for improving the ALU coprocessors of supercomputers is achieving the maximum possible speed of calculations of high capacity mono-binary codes (MBCs). Reduction in algorithmic and structural complexity, as well as the heat reduction released by the crystals of the ALU coprocessor chips is the main characteristic of improving the ALU coprocessors.

Such high-performance coprocessors are widely used as components of ALU when performing algorithmically complex calculations of statistical, correlation, spectral, cluster and entropy analysis. High-speed co-processors for multiplication and accumulation of digital data with the properties of crypto-protection of telecommunication channels are effectively used in the conditions of military operations and the modern information front, for example, in unmanned aerial vehicles, ground launchers and processors of air defense systems [1,2].

This is very effective when solving complex mathematical and algorithmic problems in the field of cryptography, holography and pattern recognition by processing RGB images in digital video cameras.

This type of calculation optimization devices can be improved when applying mathematical foundations of binary arithmetic and synchronized adder-accumulators.

Related works

Modern significant progress in the development of multi-core vector and scalar supercomputers is based on the application of achievements in microelectronics and nanotechnology [1-3]. The world's leading companies (Intel, IBM, DEC, Motorola, ARM, SPARC, MIPS, PowerPC) are replicating multi-bit processors for universal and specialized computers. The implementation of logical and computational operations in known supercomputers is usually realized in binary arithmetic of Rademacher's theoretical-numerical basis (TNB). Supercomputers of 64-bit architecture, including EM64T, Turion 64, Xeon, Core2, Corei3, Corei5, Intel (IA-64 (Itanium)), Ultra SPARC (Sun Microsystems) MIPS64 (MIPS) have a significant prospect of application in all branches of industry and military special equipment [4].

Ultra-high-performance supercomputers are used to perform complex engineering and scientific computations and other resource-intensive tasks of military equipment.

The type of such processors includes the ALU of vector and scalar supercomputers developed by Cray, Fujitsu, Hitachi, Nec, DLXV, IBM, HP [1-5].

Large bit-width hardware multipliers for binary codes, which contain n^2 of single-bit binary adders, are widely used in 32-, 64- and 128-bit coprocessors of universal computers (IBM, HP, Cray, Fujitsu, Hitachi) [6-8].

In [9-11], typical structures of ALUs in supercomputers were given, were corresponding ALU structures, which are served by the data bus without access to the accumulator registers of individual cores.

The availability of memory registers in structures of this type of ALUs provides wide possibilities of deeply synchronized and parallel algorithms for multiplication [12], division, exponentiation, etc.

The extremely widespread use of classical binary arithmetic with ripple carry-overs between bits in modern computer systems and superprocessors is a very negative factor for increasing the speed of large bit-width computing devices. For example, when adding and multiplying two n-bit MBCs, signals are delayed in the computing device by n and 2n+(n-1) clock cycles, respectively. That is, when the memory registers of the ALU cores in supercomputers are in the range of 128-2048 bits, the signal delay is, respectively, 256-4096 clock cycles when performing an addition operation in classical multi-bit binary adders (MBAs) [13] with direct data inputs and outputs. Similarly, the speed of matrix multipliers based on such components is 2048+(2048-1)=4095 clock cycles, respectively. It is obvious that such speed is insufficient and practically unacceptable to solve many modern problems, even on the basis of microelectronics of quantum computers.

Systematization of characteristics of single-bit combinational and synchronized binary adders with minimax characteristics

Today, in the field of development of microelectronic structures of single-bit half, full, combinational and synchronized binary adders, the structures proposed by us are characterized by minimal hardware complexity, maximum speed of executing ripple carry-overs and generating logical sum values [14-20].

Fig. 1 shows single-bit components of MBAs based on direct data inputs, direct inputs of ripple carry-overs and direct outputs of sums.

Such a full adder (right) has the following system characteristics:

1. A – hardware complexity: A = 7V (V - gates).
2. Input/output speed parameters: $\tau_1(a_i y_i \to S_i) = 2\upsilon$, $\tau_2\ (a_i y_i \to C_{out}) = 2\upsilon$, $\tau_3\ (C_{in} \to C_{out}) = 2\upsilon$,

υ - microcycles.

Such a full adder (left) has the following system characteristics:

1. A – hardware complexity: A = 6V.
2. Input/output speed parameters: $\tau_1(a_i b_i \to S_i) = 2\upsilon$, $\tau_2\ (C_{in} \to \overline{C_{out}}) = 1\upsilon$, $\tau_3\ (a_i b_i \to \overline{C_{out}}) = 2\upsilon$

.

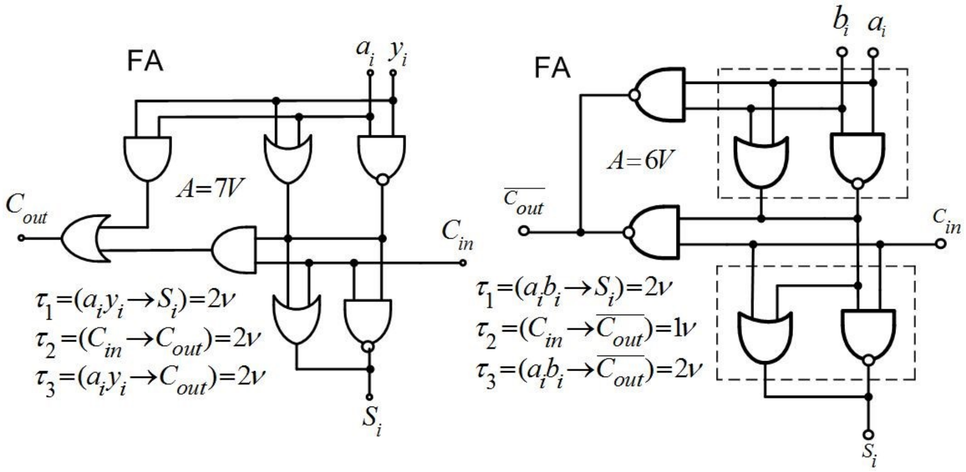


Fig.1. Microelectronic structures of single-bit full adders

Fig. 2a microelectronic shows structure of 6-gate single-bit binary full adder (FA) with paraphase data inputs and direct output of the sum, inverse inputs and outputs of ripple carry-overs and extended functionality of generating the inverse output of the sum ($\overline{S_{Ni}}$) of the first half adder (HA1).

Fig. 2b shows the minimax structure of 8-gate full adder with paraphase data inputs (qubits), inverse sum output ($\overline{S_i}$) and inverse inputs / outputs of ripple carry-overs ($\overline{C_{in}}$, $\overline{C_{out}}$).

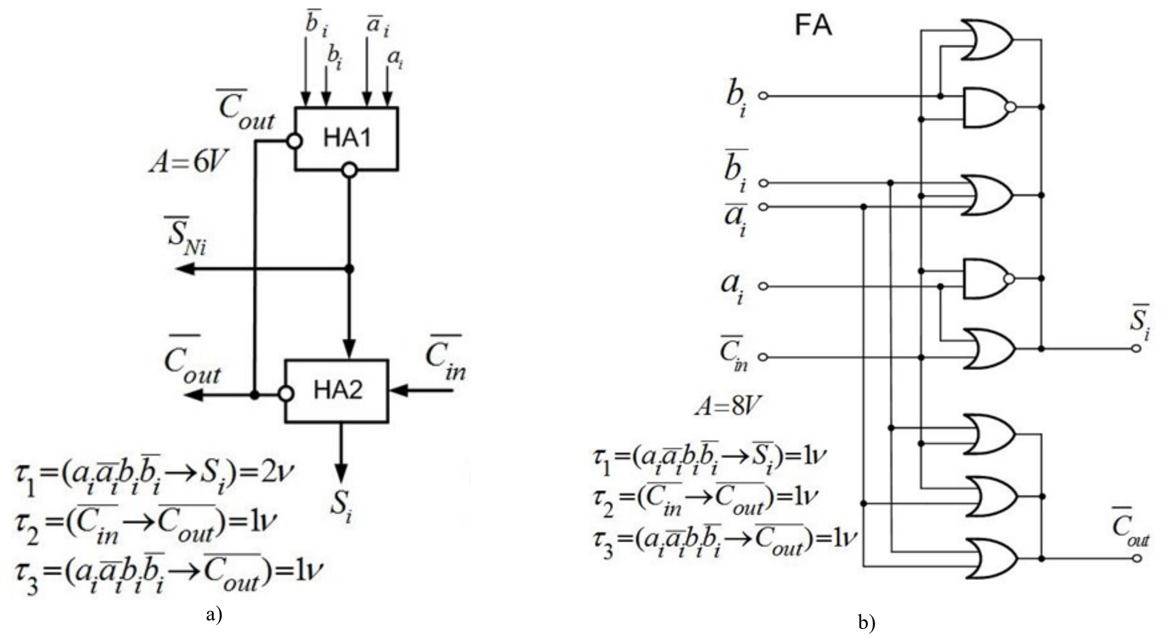


Fig.2. Microelectronic structures of single-bit adders: a) 6-gate structure of single-bit FA; b) 8-gate structure of single-bit FA

Such a FA (fig.2a) has the following system characteristics:

1. A – hardware complexity: A = 6V.
2. Input/output speed parameters: $\tau_1\,(a_ib_i \to S_i) = 2\upsilon$, $\tau_2\,(\overline{C_{in}} \to \overline{C_{out}}) = 1\upsilon$, $\tau_3\,(a_ib_i \to \overline{C_{out}}) = 1\upsilon$

.

This structure of a single-bit adder (Fig.2b), in comparison with the well-known high-speed adder with paraphase inputs and outputs [21-23], which is a component of streaming multi-bit matrix multipliers and contains 20 logic gates, has 2.5 times less hardware complexity and, respectively, characteristics of reduced power consumption and crystal heat release.

Such a FA (fig.2b) has the following system characteristics:

1. A – hardware complexity: A = 8V.
2. Input/output speed parameters: $\tau_1\,(a_ib_i \to S_i) = 1\upsilon$, $\tau_2\,(\overline{C_{in}} \to \overline{C_{out}}) = 1\upsilon$, $\tau_3\,(a_ib_i \to \overline{C_{out}}) = 1\upsilon$

The developed single-bit combinational adders with minimax characteristics compared to the well-known classical structure [13], which contains 13 logic gates, has a signal delay of 6 clock cycles when generating the sum and a ripple carry delay of 2 clock cycles, provide speed increase by 6 times when generating the sum signals and by 2 times when generating the ripple carry signals, as well as the reduction in hardware complexity by 1.6-2.2 times, respectively.

Fig. 3 show the proposed structure of single-bit synchronized full adder-accumulator (SFAA) with memory on D-triggers [16, 18].

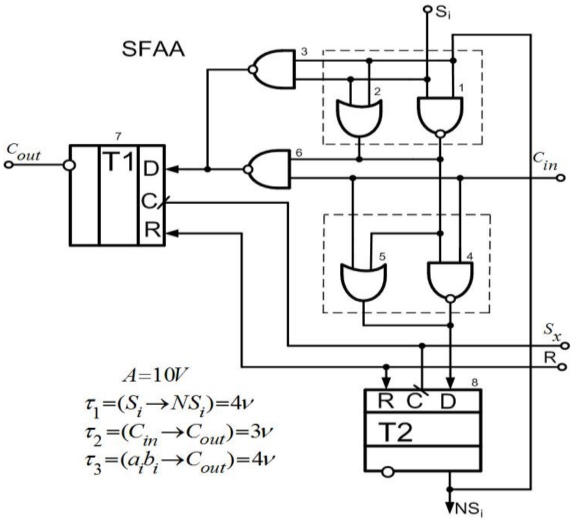


Fig. 3. Microelectronic structures of synchronized full adder-accumulator

Such a SFAA has the following system characteristics:

1. A – hardware complexity: A = 10V.
2. Input/output speed parameters: $\tau_1(S_i \to NS_i) = 4\upsilon$, $\tau_2(C_{in} \to C_{out}) = 3\upsilon$, $\tau_3(a_ib_i \to C_{out}) = 4\upsilon$

.

The developed single-bit synchronized combinational adders and synchronized adder-accumulators with minimax characteristics are the basic components of multi-bit synchronized adder-accumulators that perform computational operations of double binary arithmetic [24].

Theoretical foundations of double binary arithmetic

The basis of the binary arithmetic of the ALU in multi-bit supercomputers is the registration of the double binary code (DBC) for each bit, the sum bits ($\dot{S}_j$) and the ripple carry bits ($\dot{C}_j$) [24].

An example of generating DBC, as a result of adding two mono-binary codes (MBC) (x and y), is shown as the following graph:

$$
\begin{array}{l}
x = (a_{n-1}, \quad \ldots \quad, a_i, \quad \ldots \quad, a_1, \quad a_0 \quad) \\
\underline{+y = (b_{n-1}, \quad \ldots \quad, b_i, \quad \ldots \quad, b_1, \quad b_0 \quad)}\ , \\
\dot{d} = (\dot{C}_n < \dot{S}_{n-1}, \ldots \quad, \dot{C}_{i+1} < \dot{S}_i, \ \ldots \quad \dot{C}_2 < \dot{S}_1, \ \dot{C}_1 < \dot{S}_0)
\end{array}
$$

where $x = \sum_{i=0}^{n-1} a_i \times 2^i$; $y = \sum_{i=0}^{n-1} b_i \times 2^i$; $\dot{d} = \sum_{i=0}^{n-1} S_i \times 2^i + \sum_{i=0}^{n-1} C_{i+1} \times 2^i$.

Thus, each position of a double binary number is presented by two bits ($\dot{C}_j$, $\dot{S}_j$), which correspond to quadrilateral arithmetic according to Table 1.

Table 1

Notation of a DBC position

$\dot{C}_{i+1}$	$\dot{S}_i$	$\dot{d}_i$
0	0	0
0	1	S
1	0	2S
1	1	3S

Generation of a double binary code is shown on the example of adding two 8-bit Fermat and Mersenne numbers, which correspond to the following numbers in the decimal and mono-binary number systems: $255_{(10)} = 11111111_{(2)}$; $129_{(10)} = 10000001_{(2)}$.

Let us write these numbers in the form of a DBC and perform the operation of addition on them:

$$\begin{array}{l} \dot{x} = (0 < 1, \quad ... \quad ,0 < 1, \quad ... \quad ,0 < 1, \quad 0 < 1) \\ \underline{+\,\dot{y} = (0 < 1, \quad ... \quad ,0 < 0, \quad ... \quad ,0 < 0, \quad 0 < 1)} \\ \dot{d} = (1 < 1, \quad ... \quad ,1 < 0, \quad ... \quad ,1 < 0, \quad 1 < 0) \end{array}$$

Such operation of generating a DBC by adding two mono-binary codes (x, y) and generating their sum ($\dot{d}$) is performed using the structure of the n-bit combinational adder, which is shown in Fig. 4.

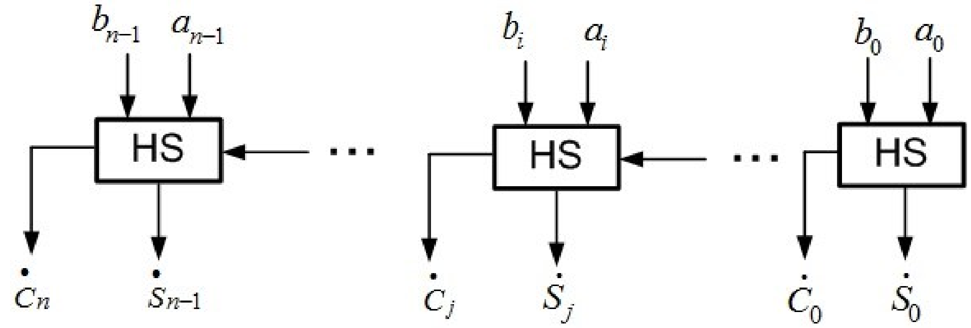


Fig. 4. Structure of n-bit sumator of DBC

An important feature of addition of two multi-bit binary mono-codes, which is implemented according to the structure shown in Fig. 4, is its maximum achievable speed, i.e., 1 clock cycle regardless of the capacity of the input codes.

The theory of double binary arithmetic developed by Professor Ya. Nykolaychuk [24] refers to a new, previously unknown, number system (YaN), the use of which allows us to increase the speed of digital data processing, presented by multi-bit binary codes, by several orders.

The use of DBCs in the ALU structures of supercomputers allows us to increase the speed of calculations and the performance of digital data processing by 1-3 orders.

Synchronized multi-bit adder-accumulators

This type of adders is widely used in processors for statistical, correlation and spectral analysis, as well as in the structures of high-performance synchronized matrix multipliers based on Booth's algorithm [25].

In Fig. 5, the structure of a synchronized multi-bit adder-accumulator (SMAA) [15], which is a component of the device for determining the selective mathematical expectation, is proposed.

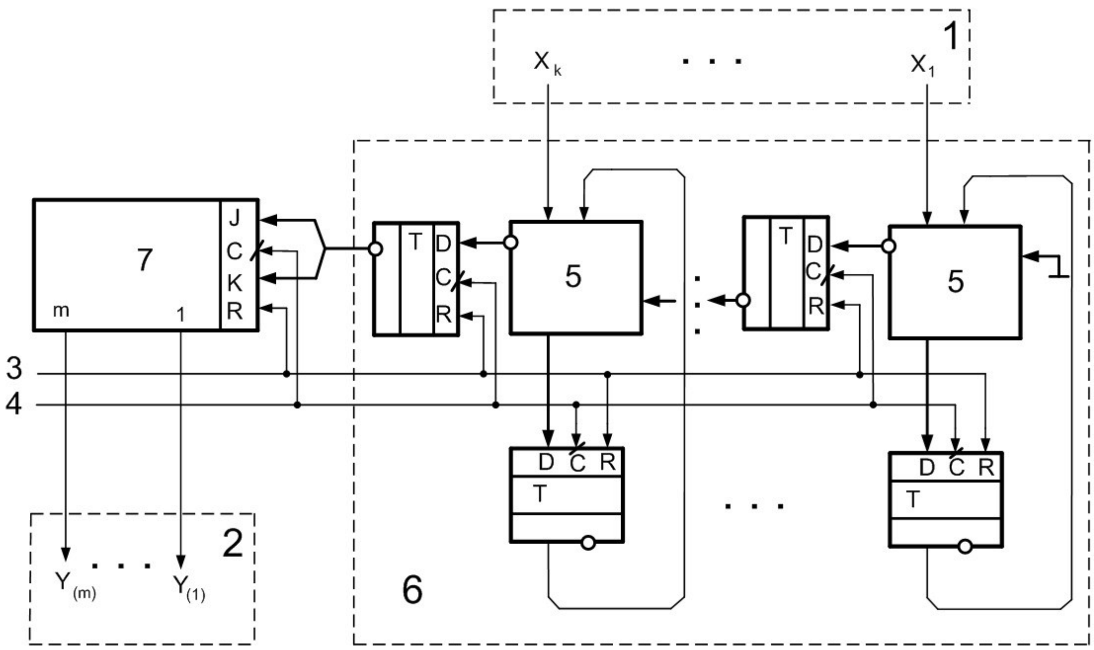


Fig. 5. Structure of the device for determining mathematical expectation

The device consists of the following components: 1 – input n-bit data bus; 2 – output m-bit data bus; 3 – channel for resetting triggers to zero state; 4 – synchronization channel; 5 – single-bit FSA; 6 – synchronized multi-bit adder accumulator; 7 – synchronized binary counter on JK triggers.

The mathematical expectation is calculated according to the expression [12]:

$$M_x = \frac{1}{n} \sum_{1}^{n} x_i \ , \tag{1}$$

where, n – sample size; k, m – bits of input and output codes.

The main advantage of the synchronized adder- accumulator as part of the device for determining selective mathematical expectation is the implementation of the addition operation in 4 clock cycles regardless of the capacity of the input binary codes, which provides an increase in the speed of such processors by 1-2 orders compared to the accumulation of the binary numbers sum in combinational adders with ripple carry-overs [13].

An advantage of such a device for determining the selective mathematical expectation is that the accumulated DBCs are not needed to be decoded, since in the most significant bits a synchronous binary counter is used based on JK-triggers, which with a delay of 2 clock cycles in each microcycle generates the output code of the most significant bits of the adder in the binary code of the Rademacher number system.

Matrix multiplier based on Booth's algorithm

The analyses of the system characteristics of known structures of matrix multipliers based on Brown's, Dadda, Booth's and other algorithms were given in [22, 25-28], where it was shown that the speed of these processors with n-bit input MBCs is not less than 2n+(n-1) clock cycles.

The disadvantage of these known structures in the multiplication cycle is the simultaneous use of all components of the adder matrix, which leads to significant power consumption and heat release of the crystals. The use of multiplier structures based on Booth's algorithm [25] also does not lead to a significant increase in speed, since in known structures, multi-bit binary combinational adders with ripple carry-overs are used in each pair of multiplied bits.

For the first time, the proposed structure of the synchronized matrix multiplier, which contains synchronized multi-bit adder-accumulators and implements Booth's algorithm, is presented in Fig. 6.

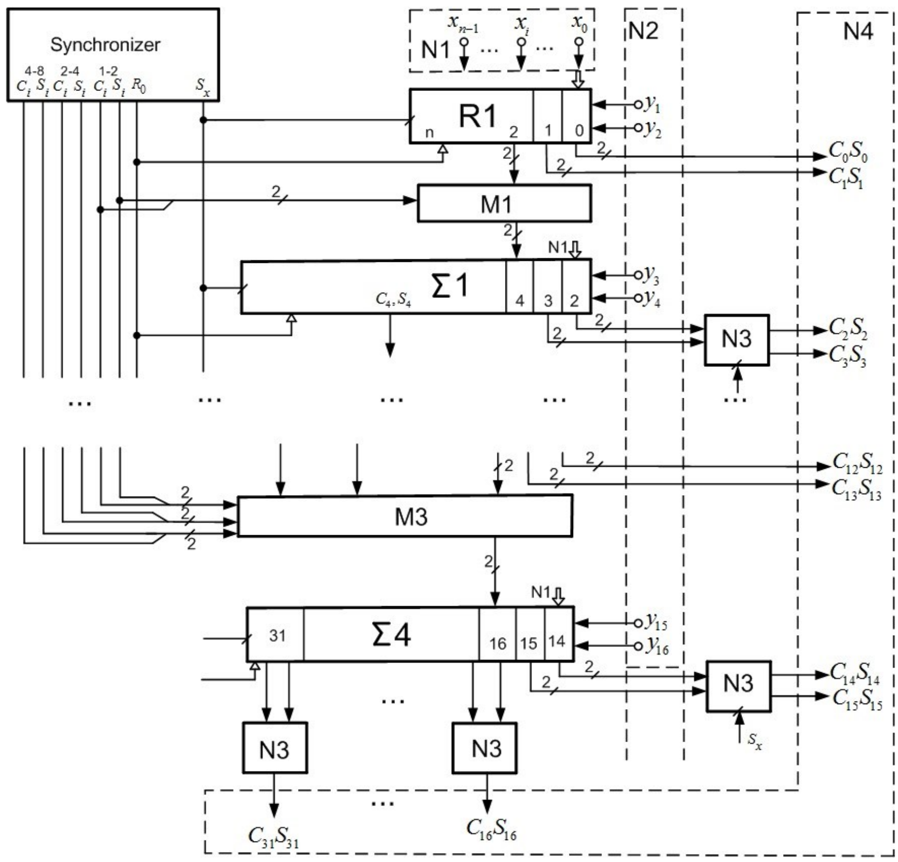


Fig.6. Structural and functional scheme of a synchronized multi-bit matrix multiplier

Such a synchronized multiplier of binary codes consists of the following components: 1 – the first n-bit input data bus of the binary code ($x_{n-1},\dots,x_i,\dots,x_0$); 2 – the second n-bit input data bus of the binary code ($y_n,\dots,y_{i+1},\dots,y_1$); 3 – synchronizer; 4 – output 4n-bit data bus of the binary code; 5 – 4n-bit memory register; 6 – n-bit synchronized adder- accumulators; 7 – synchronized multiplexers (M7.1, M7.2, …, M7.log$_2$n).

The synchronizer has output channels: R_0 – a channel for resetting all device triggers to the 0-th state; S_x – data recording synchronization channel on D-inputs of synchronized adder triggers; C_iS_i – channels for generating permits for performing the addition operation between adders (1-2, 2-4, 4-8).

The device operates according to the following algorithm:

1. The potential '0' is fed to the R-inputs of the D-triggers of all registers and adders to set them to the zero state.

2. With a delay of 3 clock cycles, a synchronizer generates a rising edge on the D-inputs of the trigger registers and synchronized adders, where Booth codes (0,X,2X,3X) are written depending on the logical values (x_i) and (y_iy_j) in each n-th digit of the multiplier.

3. The information received at the outputs of the 2 least significant bits of registers and odd adders in the form of codes $\dot{C}_j, \dot{S}_j$ та $\dot{C}_{j+1}, \dot{S}_{j+1}$ is written into the triggers of the output data bus (4), after which the gates x and y are closed and then they do not participate in the operation of the device.

4. Synchronization signals, fed to each multiplexer lasting 12 clock cycles, add code to the adders - 4 clock cycles $\dot{S}_j$ and 8 clock cycles $\dot{C}_j$.

5. The signals of the synchronizer (4, 8, 12, 16, ...) activate the multiplexers, which initiate the addition of the corresponding DBCs in pairs of adders (1-2,2-4, 4-8).

Thus, the total signal delay in this multiplier structure is calculated according to the following expression due to the bit width n=1024: $T = 3 + 12 \times (\log_2 n - 2) = 99$ (clock cycles).

Compared to the known Brown's multiplier, in which the signal delay is 2n+(2n-1) and when n=1024 it equals 4095 clock cycles, the signal delay in the proposed 1024-bit multiplier based on binary arithmetic is 99 clock cycles, respectively.

That is, the speed of the proposed multiplier is increased by 41 times in comparison with the known one.

The main advantage of the developed structure of the synchronized matrix multiplier, compared to the known ones, is the possibility of significant reduction of power consumption and heat release of crystals by disconnecting the input memory registers and odd adder-accumulators from the power supply that provides a corresponding reduction in the power consumption of the crystal in the multiplication cycle by 42%.

Conclusions

Above mentioned areas of application and the main directions for improvement of high-performance multi-bit matrix multipliers as components of ALU coprocessors in multi-core supercomputers, determine the perspective of their effective applications in solving complex computing problems, which include exponentiation, multiplication, division, square root extraction, etc. in statistical, correlation, spectral, cluster and entropy analysis.

The proposed structure of the synchronized matrix multiplier based on Booth's algorithm increases the speed by 1-2 orders compared to the speed of known matrix multipliers. The obtained results of multiplication in the form of a double binary code allow you to significantly speed up further calculations due to the absence of ripple carry-overs in DBCs.

References

1. A.O. Melnyk. "Cyber-physical systems multilayer platform and research framework. Advances in cyber-physical systems", Lviv Polytechnic National University Publishing, 2016, № 1 (1), p.1-6
2. Edward R. Miller-Jones Modern Supercomputers. The most powerful computer in the world // FastBook Publishing, 2012, – 136 p.
3. MIPS official website. URL: https://www.mips.com (дата звернення 15.06.2023).
4. IBM official website. URL: https://www.ibm.com (дата звернення 15.06.2023).
5. Intel official website. URL: https://www.intel.com (дата звернення 15.06.2023).
6. ARM official website. URL: https://www.arm.com (дата звернення 15.06.2023).
7. HP official website. URL: https://www.hp.com (дата звернення 15.06.2023).
8. Cray official website. URL: https://www.cray.com (дата звернення 15.06.2023).
9. W. Robert and Jr. Heath, "Introduction to Wireless Digital Communication," 1 Ed. Prentice Hall, 2017, p.464.
10. A.O. Melnyk, Computer architecture. Scientific ed., Lutsk: Volyn Regional Printing House, 2008, 470 p.
11. Y. Nyckolaychuk, N. Vozna, A. Davletova, I. Pitukh, O. Zastavnyy, V. Hryha Microelectronics Structures of Arithmetic Logic Unit Components // Advanced Computer Information Technologies. International Conference. ACIT'2021. – Deggendorf, Germany, September 2021. – P. 682-685.
12. Specialized Computer Technologies in Informatics: Monograph edited by Y. M. Nykolaychuk, Ternopil: Beskydy, 2017. 919 p.
13. A. Anand Kumar Fundamentals of Digital Circuits / Prentice-Hall of India Pvt.Ltd, 2007. – 664 p.
14. Patent of Ukraine. No. 115861 Single-bit half-adder Bull. No. 8, 2017
15. Patent of Ukraine. No. 150332 Binary adder-accumulator. Bul. No. 5, 2022
16. Patent of Ukraine. No. 146833 Single-bit synchronized full-adder Bull. No. 12, 2021

17. Patent of Ukraine. No. 150331 Carry-look-ahead adder. Bul. No. 5, 2022
18. Patent of Ukraine. No.147625 Single-bit synchronized half-adder. Bul. No. 21, 2021
19. Nyckolaychuk, V. Hryha, N. Vozna, A. Voronych, A. Segin, P. Humennyi High-performance coprocessors for arithmetic and logic operations of multi-bit cores for vector and scalar supercomputers // Advanced Computer Information Technologies. 12th International Conference. ACIT 2022. – Spišská Kapitula, Slovakia, September 2022. – P. 410-414
20. Patent of Ukraine. No. 150330 Single-bit full-adder. Bul. No. 5, 2022
21. Patent of Ukraine. No. 132520 Matrix multiplier. Bul. No. 4, 2019
22. Patent of Ukraine. No. 150331 Matrix multiplier. Bul. No. 4, 2019
23. Patent of Ukraine. No. 123752 Multipler of multi-bit data streams. Bul. No. 21, 2021
24. Y. Nyckolaychuk, V. Hryha, N.Vozna, I.Pitukh and L. Hryha High-performance multi-bit adder-accumulators as component of the ALU in supercomputers // publ. on CEUR Workshop Proceedings, 2023, P. 649-661.
25. Barun Biswas, Bidyut B. Chaudhuri Generalization of Booth's Algorithm for Efficient Multiplication, Procedia Technology, Volume 10, 2013, P. 304-310.
26. V. Gryga, I. Kogut, V. Holota, R. Kochan, S. Rajba, T. Gancarczyk, U. Iatsykovska Spatial-Temporal Transformation of Matrix and Multilayer Algorithms of Binary Number Multiplications // Proceedings of 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications. IDAACS'2019. – Metz, France, September 18-21, 2019. – P. 691-694.
27. J.W. Lee, C.Y. Meher, P. K. Patra. Concurrent Error Detection in Bit-Serial normal Basis Multiplication Over GF(2m). Using Multiple Parity Prediction Schemes in Large Scale Integration (VLSI) Systems, IEEE Transactions on Very Large Scale integration (VLSI) Systems: 2009-08-25.
28. James Reinders and Jim Jeffers, High Performance Parallelism Pearls Volume One: Multicore and Many-core Programming Approaches, 2014, 600p.

Yaroslav Nykolaychuk **Ярослав Николайчук**	DrS on Engineering, Professor of Specialized Computer System Department, West Ukrainian National University, Ternopil, Ukraine e-mail: ya.nykolaichuk@wunu.edu.ua orcid.org/0000-0002-2393-2332, Scopus Author ID: 24179012300, Researcher ID: H-4325-2017	доктор технічних наук, професор кафедри спеціалізованих комп'ютерних систем, Західноукраїнський національний університет, Тернопіль, Україна
Volodymyr Hryha **Володимир Грига**	PhD, Associate Professor of Computer Engineering and Electronics Department, Vasyl Stefanyk Precarpathian National University, Ivano-Frankivsk, Ukraine e-mail: volodymyr.gryga@pnu.edu.ua orcid.org/0000-0001-5458-525X, Scopus Author ID: 57188576389, Researcher ID: HKE-6775-2023	кандидат технічних наук, доцент кафедри комп'ютерної інженерії та електроніки, Прикарпатський національний університет імені Василя Стефаника, Івано-Франківськ, Україна
Nataliia Vozna **Наталія Вознa**	DrS on Engineering, Professor of Specialized Computer System Department, West Ukrainian National University, Ternopil, Ukraine e-mail: n.vozna@wunu.edu.ua orcid.org/0000-0002-8856-1720, Scopus Author ID: 24178221500, Researcher ID: H-4297-2017	доктор технічних наук, професор кафедри спеціалізованих комп'ютерних систем, Західноукраїнський національний університет, Тернопіль, Україна
Ihor Pitukh **Ігор Пітух**	PhD, Associate Professor of Specialized Computer System Department, West Ukrainian National University, Ternopil, Ukraine e-mail: i.pitukh@wunu.edu.ua orcid.org/0000-0002-3329-4901, Scopus Author ID: 37122611700, Researcher ID: H-5367-2017	кандидат технічних наук, доцент кафедри спеціалізованих комп'ютерних систем, Західноукраїнський національний університет, Тернопіль, Україна
Lyudmila Hryha **Людмила Грига**	Teacher, Teacher of Programming and Informatics Department, Nadvirna Vocational College by NTU, Nadvirna, Ukraine e-mail: hrihaludmila31@gmail.com orcid.org/0000-0002-6260-7559, Scopus Author ID: 57188576389, Researcher ID: ITV-4046-2023	викладач, викладач циклової комісії програмування та інформатики, ВСП "Надвірнянський фаховий коледж", Надвірна, Україна

https://doi.org/10.31891/csit-2023-2-4
UDC 378:005.6

Andriy PEREYMYBIDA
Ivan Franko National University of Lviv
Mariia RASHKEVYCH
Lviv Polytechnic National University

EFFECTIVE PARTNERSHIP WITH BUSINESS AS A MEANS OF IMPROVING THE QUALITY OF EDUCATION IN THE FIELD OF INFORMATION TECHNOLOGIES

The quality of education in the field of information technology depends on the use of modern tools of cooperation between universities and leading companies. Micro-credentials, which are rapidly spreading throughout the world, are among the new innovative tools. On the example of the SoftServe company, the experience of using dual study, non-formal study, professional training of teachers is analyzed. Problems are identified and models of SoftServe cooperation with universities are described, which increase the employability of graduates.

One of the important criteria that determine the quality of education is consideration of labor market requirementsin education standards and educational programs (EP). This criteria is also among the main ones that are considered in the process of accreditation of individualEP. In the system of vocational education and training, the balance between the content of education and the requirements of employers is ensured by the fact that the training of specialists is carried out on the baseof list of professions (professional qualifications), and both the standards of vocational education and individual educational programs are built based on relevant professional standards.

In the field of professional pre-higher and higher education, the situation is more complicated, since the training of students is based on the list of specialties (established on the national level), and the relationship between individual specialties and the corresponding professional qualifications and standards is much more complicated, sometimes even ambiguous. In these conditions, it is critical that institutions of higher education, especially technological ones, cooperate as closely as possible with the relevant enterprises of the region, professional organizations, and individual employers. Such a phenomenon has already become typical in the field of information technologies (IT). One of the factors that directly affects this is the desire of IT companies and, especially, students to find employment as soon as possible, which in turn has a significant impact on the change in the market of educational services: short educational programs, dual education, non-formal and informal education are becoming more and more popular.

Keywords: models of dual study, non-formal and informal education, micro-credentials.

Андрій ПЕРЕЙМИБІДА
Львівський Національний Університет ім. І. Франка
Марія РАШКЕВИЧ
Національний Університет «Львівська політехніка»

ЕФЕКТИВНЕ ПАРТНЕРСТВО З БІЗНЕСОМ ЯК ЗАСІБ ПІДВИЩЕННЯ ЯКОСТІ ОСВІТИ В ГАЛУЗІ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Якість освіти в сфері інформаційних технологій в значній мірі залежить від використання сучасних форм та інструментів співпраці університетів із провідними компаніями. В числі нових інноваційних інструментів зараз виступають мікрокваліфікації, які швидко поширюються в цілому світі. На прикладі компанії SoftServe проаналізовано досвід застосування дуальної форми навчання, неформального навчання, професійного вишколу викладачів. Виділені проблеми та сформульовано моделі співпраці SoftServe із університетами, які підвищують готовність випускників до працевлаштування.

Одним із важливих критеріїв, що визначають якість освіти, є врахування вимог ринку праці в освітніх стандартах та освітніх програмах (ОП). Цей критерій також є одним з основних, які враховуються в процесі акредитації окремих ОП. У системі професійно-технічної освіти баланс між змістом освіти та вимогами роботодавців забезпечується тим, що підготовка фахівців здійснюється на основі переліку професій (професійних кваліфікацій), а стандарти професійної освіти та індивідуальні освітні програми будуються на основі відповідних професійних стандартів.

У сфері фахової передвищої та вищої освіти ситуація складніша, оскільки підготовка студентів здійснюється на основі переліку спеціальностей (встановленого на національному рівні), а зв'язок між окремими спеціальностями та відповідними професійними кваліфікаціями і стандартами набагато складніший, іноді навіть неоднозначний. У цих умовах критично важливо, щоб заклади вищої освіти, особливо технічні, якомога тісніше співпрацювали з відповідними підприємствами регіону, професійними організаціями та окремими роботодавцями. Таке явище вже стало типовим у сфері інформаційних технологій (ІТ). Одним із факторів, який безпосередньо впливає на це, є бажання ІТ-компаній і, особливо, студентів якомога швидше працевлаштуватися, що, в свою чергу, має значний вплив на зміну ринку освітніх послуг: дедалі більшої популярності набувають короткі освітні програми, дуальна освіта, неформальна та інформальна освіта.

Ключові слова: моделі дуального навчання, неформальне та інформальне навчання, мікрокваліфікації.

Introduction

One of the important criteria that determine the quality of education is consideration of labor market requirementsin education standards and educational programs (EP). This criteria is also among the main ones that are considered in the process of accreditation of individualEP. In the system of vocational education and training, the balance between the content of education and the requirements of employers is ensured by the fact that the training of specialists is carried out on the baseof list of professions (professional qualifications), and both the standards of vocational education and individual educational programs are built based on relevant professional standards.

In the field of professional pre-higher and higher education, the situation is more complicated, since the training of students is based on the list of specialties (established on the national level), and the relationship between individual specialties and the corresponding professional qualifications and standards is much more complicated, sometimes even ambiguous. In these conditions, it is critical that institutions of higher education, especially technological ones, cooperate as closely as possible with the relevant enterprises of the region, professional organizations, and individual employers. Such a phenomenon has already become typical in the field of information technologies (IT). One of the factors that directly affects this is the desire of IT companies and, especially, students to find employment as soon as possible, which in turn has a significant impact on the change in the market of educational services: short educational programs, dual education, non-formal and informal education are becoming more and more popular.

The SoftServe company is one of the leaders in the IT sector, has over 10,000 employees, a significant number of whom work in the Western region of Ukraine. The rapid development of the IT market, which was somewhat slowed down, but not stopped, by the Ukrainian-russian war, has as a result a shortage of personnel. Therefore, it is not surprising that SoftServe closely cooperates with more than 60 higher education institutions of the region. Hundreds of students at these universities, starting from the third year of study, work in the company. To say that their level of training and the modernity of the programs satisfy SoftServe would be a significant exaggeration. And it is not surprising that since 2019 the company has significantly intensified a cooperation with higher education institutions, using all possible tools: dual education programs, teacher training, short training courses, joint development of EP and their expert evaluation, scientific laboratories, etc. So, for example, almost a thousand students participate in 20 dual study programs, more than 4,350 teachers have improved their qualifications, 170 programs of individual academic disciplines have been updated, students working at SoftServe have access to the UDEMY educational platform and specialized online courses developed by the company.

The main problems that hinder SoftServe cooperation with educational institutions are:

- inconsistency of the content of EP with the rapidly changing needs of the labor market, they are too theorized, not saturated with practical components, the selective blocks of disciplines they contain are oriented more towards teachers than students;
- insufficient personal experience of teachers in information technologies and tools used by the company in commercial projects development;
- insufficient development of students' soft skills;
- terminological misunderstandings between the EP designers and the company's mentors during the modernization of the EP as a whole and individual modules;
- recognition in EP and individual disciplines of learning outcomes obtained by students in the process of practical work and professional development within the company.

Among the other modern educational innovations, dual study deserves special attention. A dual form of education is an in-depth, integrated form of education for those who are ready to combine study in an educational institution and gaining experience during real work in companies. In October 2019, the Ministry of Education and Science of Ukraine initiated a pilot project in professional pre-higher and higher education institutions to train specialists using a dual form of education. The analytical report on the results of which was published last year [1]. Currently, in the legislation of Ukraine, namely in the regulations of Ukraine "On Education", "On Professional Pre-Higher Education", dual education is legalized, and this creates appropriate conditions for the implementation of this form of education in the activities of higher and professional pre-higher education institutions.

The use of elements of dual education has gained special importance since 2020, when firstly the COVID-19 pandemic and then the russian intervention made the traditional educational process,basedon university classrooms and laboratories, impossible. Institutions of higher education lost a significant part of their capacity to provide quality education and began to look for help from outside, primarily among employers. SoftServe, among other technology oriented companies, immediately began to introduce new opportunities, initially developing 3 dual education programs with leading Lviv universities. As always, when new approaches to the educational process are introduced, the testing of various organizational models has begun, using the appropriate possibilities of legislation.

Today, the SoftServe company uses two main models of dual training:

1. "Classical model" - when practical training, which is provided in the professionally oriented disciplines of the program, is implemented in the company mainly during the implementation of real projects.
2. "Online model" - when students, while working in the company, obtain part of the necessary learning outcomes by individual (under the systematic supervision of mentors) training on specialized platforms (both open access and internal). Recognition of the obtained learning outcomes (as those obtained in non-formal education) is based on the internal Regulations of each higher education institution.

The "classical model" provides students with the opportunity to apply their academic knowledge and skills in real work settings where they can apply theoretical concepts to practical tasks and collaborate with professionals in the relevant field. During the development of real projects in the company, students can gain valuable practical experience, develop communication and interpersonal skills, and the ability to work in a team. The "classical model" of dual study contributes to students' deeper understanding of the practical aspects of their profession, and also helps them determine their professional interests and goals. It also helps align university education with the needs of the labor market, as companies can actively influence the curriculum and produce the specific skills and knowledge required in a particular industry.

In addition, this model facilitates the deepening of partnerships between universities and companies, ensuring the exchange of knowledge, resources, and expertise. Students participating in this model get a unique opportunity to build a network of professional contacts and establish connections with specialists in the relevant field. However, this is a more complicated model to implement because it involves a lot of bureaucratic procedures: such as the concluding of a bilateral or tripartite contract, and an employment contract.

In addition, the introduction of dual study often causes resistance from teachers, since part of the credits of the educational program or even academic courses are officially transferred to the company, which leads to a decrease in the overall educational load, and sometimes to a reduction in staff.

"Online model" uses the possibilities of recognition of learning outcomes obtained in non-formal and informal education introduced in 2022 by the Ministry of Education and Science of Ukraine. Students, working in the company and performing real projects at the same time, have access to a variety of educational resources, which can be both open for general access and internal, depending onthe specific project. These resources may include video lessons, e-courses, interactive tasks, tests. Enrollment of acquired knowledge and skills into university educational programs takes place based on the internal regulations of individual institutions of higher education, which recognize the learning outcomes acquired by students during work at the company and their independent learning on platforms. This model promotes flexibility and individualization of learning, as students can learn at their own pace and choose the resources that best suit their individual project trajectories. It also facilitates interaction between students and mentors, creating a virtual community where knowledge, experience and practical skills can be shared.

"Online model" shows new opportunities for students and companies in improving the quality of education in the field of information technologies by combining academic knowledge with practical work experience in real conditions. Based on the "online model", the SoftServe company is currently co-operating with more than 15 institutions of higher education, 20 EP in the Information technologies subject area have been updated. Students note that with the help of this model it is possible to effectively allocate their time, focus on the work project and at the same time get all the necessary knowledge required by the university curriculum. This model is a great challenge for responsible and conscientious students who are ready to take responsibility and really want to become professionals while studying at the university.

Obviously, in both cases, the work begins with a joint analysis and modernization of educational programs (lasts approximately 6 months), and includes preliminary training of students in soft skills, which is funded by the company during the second year of study. Further, based on the results of the assessment, students are selected for dual education programme, which continues in the 3rd and 4th years of study, on a competitive basis.

We should emphasis that the main factors inhibiting the expansion of the range of programs using elements of dual study and the recognition of the learning outcomes gained in non-formal education are: fears of teachers that reducing the classroom load will lead to a staff reduction, a certain mistrust of the academic community regarding the quality of learning outcomes in non-formal education, significantcompany's expenses for the participation in dual form of education and the lack of a guarantee that the student will continue to work for the company after graduating from university.

As already noted, the SoftServe company pays special attention to improving the qualifications of teachers at partner universities, creating opportunities for them to study new information technologies and tools and promoting professional development. For this purpose a set of certificate programs, specialized webinars, short-term internships, and trainings, etc. are used.

Today, micro-qualifications (micro-credentials) begin to play a crucial role in professional development and life-long learning, the need for their widespread use is emphasized in the European Strategy for the Development of Skills and Abilities [2]. Since the beginning of 2020s several important recommendations concerning the micro-credentials development and implementation have been published by UNESCO [3], European Training Foundation (ETF) [4] and European Center for the Development of Vocational Education (CEDEFOP) [5]. In different forms micro-credentials have been used for a long time, but there is no generally accepted understanding of them, formats of description, assignment methods, approaches to their mutual recognition both at the domestic and international levels. In this regard, in 2021, the European Commission, after carrying out special studies in cooperation with the CEDEFOP, published the Council Recommendation on a European approach to micro-credentials for lifelong learning and employability[6]. These recommendations define micro-credentials as the record of the learning outcomes that a learner has acquired following a small volume of learning; these learning outcomes will be assessed based on transparent and clearly defined criteria. They can be independent or combined with other credentials (educational achievements) and are also supported by quality assurance in accordance with agreed standards in the relevant sector or area of activity. This document recommends to EU members and candidates for membership the introduction ofmicro-credentials to national qualification systems, offers the main principles and approaches to their development and implementation, as well as the formats of their description in the relevant certificates.

Many countries and international professional organizations (for example, the European Consortium of Massive Open Online Courses (European MOOC Consortium) [7], Latvia [8] and others) have published special analytical reports with the aim to promote the use of micro-credentials both in education and professional development.

Since the introduction of micro-credentials into the system of qualifications must consider national characteristics, this problem becomes particularly relevant for Ukraine, as a candidate for EU membership. At various levels, we have already started discussing the expediency and possibilities of using micro-credentialsin education and in the professional development, the first scientific publications and proposals have appeared [9].

The introduction of micro-credentials is particularly effective in the field of information technologies for the establishment of effective cooperation between universities and IT companies, the implementation of digital technologies into the educational process, and the improvement of the effectiveness of dual education.

The above-mentioned European Consortium of Massive Open Online Courses gives the following recommendations on the formatting of micro-credentials:

- the total students workload is 4-6 ECTS credits,
- learning outcomes relate to levels 6-8 of the European Qualifications Framework (EQF) with the possibility of extension to levels 4 and 5,
- assessment takes place in accordance with accepted quality assurance standards,
- reliable identification of the person must be ensured during the assessment,
- have a direct relationship with the needs of the labor market,
- the final certificate of micro- credentials must contain information about the content of the course, learning outcomes, their level in relation to the EQF, the amount of ECTS credits.

Micro-credentials can also play an important role in creating opportunities for both students and workers to gain professional qualifications that are defined by relevant professional standards. By its structure, the professional standard (based on which professional qualifications are assigned) is hierarchical: the qualification covers a number of work functions, for the performance of which it is necessary to possess a certain set of professional competences, eachof which is a combination of relevant learning outcomes. According to the national law in Ukraine, both full and partial qualifications are based on job functions, which greatly reduces their flexibility.In addition, the development and approval of professional standards is a long procedure, as a result of which the newly created standards may lose their relevance. Thesecond one, especially applies to the field of information technologies. At the same time, micro-credentials can be designedon a set of professional competencies needed by employers today, and educational institutions and qualification centers have the right to use them for the professional development of employees and meeting the needs of the labor market.

In general, according to SoftServe opinion, micro-credentials should be developed with the aim of: acquisition of certain technologies by students; specialization by choosing appropriate selective blocks of disciplines within the framework of a broad programme of bachelor's training. In the latter case, a significant role can be played by corporate and occupational standards, the development of which is currently being initiated by SoftServe company specialists.

Conclusions

A close partnership between universities and technology companies is a modern trend in the development of education and is especially effective in the field of information technology, where IT companies conduct specialized training in their academies, offer a wide selection of specialized online courses. The formalization of the received learning outcomes in the form of relevant documents can be carried out both by formal education institutions through the inclusion of training results in their educational programs, and by qualification centers through the mechanisms of recognition of the outcomes of non-formal and informal training, which are accredited at the state level and ensure the quality of final assessment and the qualifications obtained.

Undoubtedly, the organization of specialists training by dual study in the system of higher and professional pre-higher education is a more complex, time-consuming process, as it requires a more careful balancing of the interests of the parties. Therefore, the cooperation of employers, educators, scientists, public and international organizations is crucial for the development and implementation of different models of dual education for students, contributes to increasing of the higher education institution ability to provide education and training of better quality and meeting the current and futurelabor market demands.

References

1. Representation of the Foundation named after Friedrich Ebert in Ukraine (2021). Analytical report on the results of the second year of the pilot project in the institutions of vocational pre-higher and higher education on the training of specialists using the dual form of education. Kyiv.
2. OECD (2019). OECD Skills Strategy 2019: skills to shape a better future. Paris: Organisation for Economic Co-operation Development. Retrieved from: https://www.skillsforemployment.org/KSP/en/Details/index.htm?dn=EDMSP1_246628 .
3. UNESCO (2022). Towards a common definition of micro-credentials. Retrieved from: https://unesdoc.unesco.org/ark:/48223/pf0000381668 .
4. ETF (2022). Micro-credentials are taking off: how important are they for making lifelong learning a reality? Retrieved from: https://www.etf.europa.eu/sites/default/files/2022-05/Micro-credentials%20%281%29-combined_1%20copy_0.pdf .
5. CEDEFOP (2022). Are microcredentials becoming a big deal? Retrieved from: https://www.cedefop.europa.eu/files/9171_en.pdf

6. European Commission (2022). Council Recommendation on a European approach to micro-credentials for lifelong learning and employability. Retrieved from: https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=COM:2021:770:FIN .

7. EMC (n.d.). EMC Common Microcredential Framework. European MOOC Consortium. Available under: https://emc.eadtu.eu/images/EMC_Common_Microcredential_Framework_.pdf

8. Anita Lice. (2021). Micro-Credentials in Latvia. Retrieved from: https://www.nki-latvija.lv/content/files/Mico_credent_LV_draft.pdf .

9. Semigina T., Rashkevyh Yu. Micro-credentials as a trend in the development of modern continuous education. Theory and practice of modern science: III International Scientific and Theoretical Conference. Vol.2. Kraków: European Scientific Platform, 2022. P. 46-50. URL: https://bit.ly/3tVOkDK

Andriy Pereymybida **Андрій Переймибіда**	PhD, Associate Professor of Applied Mathematics Department, Ivan Franko National University of Lviv ORCID ID: 0000-0002-5367-6245 E-mail: aperey@gmail.com	К.ф.-м.н., Доценткафедри прикладної математики, Львівського Національного Університету ім. І. Франка
Mariia Rashkevych **Марія Рашкевич**	PhD, Associate Professor of Iinformation systems and technologies Department, Lviv Polytechnic National University ORCID ID: 0000-0001-5490-1750 E-mail: marikupchak@gmail.com	К.т.н., Доцент кафедри інформаційних систем та технологій, Національного університету «Львівська політехніка»

https://doi.org/10.31891/csit-2023-2-5
UDC 004.93

Oleksii MAKIEIEV, Natalia KRAVETS
Kharkiv National University of Radio Electronics

STUDY OF METHODS OF CREATING SERVICE-ORIENTED SOFTWARE SYSTEMS IN AZURE

The modern development of service-oriented software systems is accompanied by the wide use of cloud technologies, which affect the competitiveness of companies and their systems, which provide opportunities to expand the client base thanks to the coverage of several regions of the city or country.

The advantage of cloud services is availability in any part of the world where there is an Internet connection. Cloud providers provide a large volume of services for various needs: such as hosting, deployment of containers, file storage, databases, etc.

In particular, all the most popular cloud providers offer several options for creating service-oriented software systems, including both standard technologies and proprietary developments. This paper compares the methods of creating service-oriented software systems based on the Azure cloud platform: Azure Container Apps, Azure Kubernete Service, and Azure Red Hat OpenShift. The subject area of technologies for the implementation of service-oriented application architecture is considered, and criteria for the analysis of methods for implementing applications with such an architecture are proposed. A software solution for comparing methods of creating service-oriented applications based on the Azure cloud platform was designed and developed. The developed software system provides an opportunity to rent scooters, bicycles and cars.

The purpose of the study is a comparative analysis of the methods of creating service-oriented software systems based on Azure services, and the subject of the study is a software solution implemented using these methods.

The purpose of this work will be the development of a software system that will provide an opportunity to rent scooters, bicycles and cars. Using this system, we will investigate the deployment of this system on certain services from Azure.

The results of this research on Azure services: Azure Container Apps, Azure Kubernete Service and Azure Red Hat OpenShift can be used when creating a new software system, when expanding an existing software system, when transferring software system components from other platforms to the Azure platform using these services.

Keywords: service-oriented software system, cloud technologies, Azure cloud, Azure Container Applications, Azure Kubernetes Service, Azure Red Hat Open Shift, docker, web-services.

Олексій МАКЄЄВ, Наталія КРАВЕЦЬ
Харківський національний університет радіоелектроніки

ДОСЛІДЖЕННЯ МЕТОДІВ СТВОРЕННЯ СЕРВІСНО-ОРІЄНТОВАНИХ ПРОГРАМНИХ СИСТЕМ У AZURE

Сучасний розвиток сервісно-орієнтованих програмних систем супроводжується широким використанням хмарних технологій, які впливають на конкурентоспроможності компаній та їх систем, що надають можливості в розширенні клієнтської бази завдяки охопленню декількох областей міста чи країни.

Перевагою хмарних сервісів є доступність в будь-який точці світу, де є підключення до Інтернету. Хмарні провайдери надають великий обсяг сервісів для різних потреб: таких як хостинг, розгортання контейнерів, файлове сховище, бази даних тощо.

Зокрема всі найпопулярніші хмарні провайдери пропонують кілька варіантів створення сервісно-орієнтованих програмних систем, включаючи як стандартні технології так і власні розробки. У даній роботі виконане порівняння методів створення сервісно-орієнтованих програмних систем на базі хмарної платформи Azure: Azure Container Apps, Azure Kubernete Service та Azure Red Hat OpenShift. Розглянуто предметну область технологій реалізації сервісно-орієнтованої архітектури застосунків, запропоновано критерії для аналізу методів реалізації застосунків із такою архітектурою. Спроектовано та розроблено програмне рішення для порівняння методів створення сервісно-орієнтованих застосунків на базі хмарної платформи Azure. Розроблена програмна система надає можливість брати в оренду самокати, велосипеди та автомобілі.

Метою дослідження є порівняльний аналіз методів створення сервісно-орієнтованих програмних систем на базі сервісів Azure, а предметом дослідження – програмне рішення, яке реалізоване за допомогою цих методів.

Отримані результати даного дослідження над Azure сервісами: Azure Container Apps, Azure Kubernete Service та Azure Red Hat OpenShift, можна буде використовувати при створені нової програмної системи, при розширенні існуючої програмної системи, при перенесенні компонентів програмної системи з інших платформ на Azure платформу використовуючи дані сервіси.

Ключові слова: сервісно-орієнтована програмна система, хмарні технології, хмара Azure, програми-контейнери Azure, сервіс Kubernetes Azure, Azure Red Hat OpenShift, докер, веб-сервіси.

Introduction

Service-oriented architecture [1] is a software architectural pattern that uses a modular approach to software development based on the use of distributed, loosely coupled replaceable components equipped with standardized interfaces to interact using standardized protocols.

Service-oriented architecture is not tied to a specific technology. It can be implemented using a wide range of technologies, including technologies such as Web services, message-oriented middleware, enterprise service bus, microservices.

This architecture can include the following elements, as can be Application frontend, services, service bus or message broker, data storage. These elements can be loosely connected services that interact through a strictly defined interface with each other.

When implementing a service-oriented architecture, it is necessary to attach to certain principles, namely:

- ensuring compatibility;
- weak interdependence;
- abstraction;
- degree of detailing.

Ensuring contiguity means that any system will be able to run the service regardless of the underlying platform or programming language. For example, business processes can use services written in C#, Java, and Python.

The principle of weak interdependence suggests that services should be weakly connected, which should have as few external dependencies as possible. That way, if you change a service, it won't affect client applications and other services that use that service.

The principle of abstraction means that users do not need to know the logic of the service code or the details of the implementation. For them, services should be like a black box.

The principle of the degree of detail means that the services should have an appropriate size and scope. Developers can use multiple services to create a composite service to execute complex logic.

Service-oriented architecture has a number of advantages, namely:

- reduction of market entry time;
- effective service;
- improved adaptability.

Efficient maintenance makes it easier to create, update, and debug small services than larger blocks of code in monolithic applications.

Improved adaptability makes it possible to modernize your programs effectively and without unnecessary costs.

For the study, service-oriented software for renting vehicles, namely scooters, bicycles, and cars, will be developed to investigate the deployment system on certain services from Azure.

Related Works

At the current time, we have some technologies for implementing service-oriented architecture:

- Microservice architecture[2];
- RESTful Web Services[3];
- GraphQL.

Microservice architecture is a specific implementation of a service-oriented architecture (SOA) that focuses on breaking down a large, monolithic system into smaller, independent services that can be developed, deployed, and scaled independently. Microservices are designed to be loosely coupled and communicate with each other through APIs.

In a service-oriented architecture, microservices can provide a number of benefits, including:

- Greater flexibility: Microservices allow for faster and more frequent deployments, as individual services can be developed and deployed independently of each other.
- Better scalability: Because microservices can be scaled independently, it's easier to handle changes in demand for specific services without impacting the rest of the system.
- Improved fault tolerance: By isolating each service, it's easier to handle failures in one service without impacting the rest of the system.

However, implementing a microservice architecture in a service-oriented architecture can also introduce some challenges. For example:

- Increased complexity: With more services, there is greater complexity in managing service-to-service communication, monitoring, and testing.
- Data consistency: Maintaining consistency across multiple services can be challenging, as each service may have its data store.
- Service discovery: In a microservice architecture, services need to be discoverable by other services, which can be challenging to manage.

Representational State Transfer (REST) is an architectural style that defines a set of constraints for building web services. RESTful web services conform to these constraints and are designed to be simple, lightweight, and scalable. They use HTTP methods such as GET, POST, PUT, and DELETE to perform CRUD (create, read, update, delete) operations on resources. In a service-oriented architecture, RESTful web services can be used to facilitate communication between services by providing a standard way for services to interact with each other. Each service can expose a RESTful API that other services can use to access its functionality.

Some benefits of using RESTful web services in a service-oriented architecture include:

- Scalability: RESTful web services are designed to be scalable, making it easy to handle changes in demand for specific services.

- Interoperability: Because RESTful web services use standard HTTP methods, they can be accessed by a wide variety of clients and platforms.
- Simplicity: RESTful web services are easy to understand and can be implemented using simple, lightweight frameworks.

However, implementing RESTful web services in a service-oriented architecture can also introduce some challenges. For example:

- Data consistency: Maintaining consistency across multiple services can be challenging, as each service may have its own data store.
- Service discovery: Services need to be discoverable by other services, which can be challenging to manage.
- Security: RESTful web services need to be secured against unauthorized access and attacks such as SQL injection.

GraphQL is a query language and runtime for APIs that was developed by Facebook in 2015. It provides a more flexible and efficient way for clients to request data from servers, allowing clients to specify exactly what data they need and reducing the number of round trips to the server. In a service-oriented architecture, GraphQL can be used to facilitate communication between services by providing a single API endpoint that aggregates data from multiple services.

One of the main benefits of using GraphQL in a service-oriented architecture is that it allows for greater decoupling between services. Because each service can expose its own GraphQL schema and resolvers, other services can query that service's data without needing to know the details of its internal implementation. This makes it easier to change or replace individual services without impacting the entire system.

However, implementing GraphQL in a service-oriented architecture can also introduce some challenges. For example, it requires careful consideration of data ownership and access control, as well as potential performance implications due to the increased number of round trips to the server. Additionally, it may require additional development effort to create and maintain the GraphQL schema and resolvers for each service.

The software system will be implemented using RESTful web services because web services are easy to implement using modern frameworks, web services can be used in different systems because they support HTTP methods, and web services can support scalability. Using the RESTful web services the system will be implemented for renting scooters, bikes, and cars. This system will be used in the Azure services to analyze methods for creating a service-oriented software systems.

Methods of creating service-oriented software systems in Azure

In Azure we choice three services for creating container applications:

- AzureContainerApps[4];
- AzureKubernetesService[5];
- AzureRedHatOpenShift[6].

Azure Container Apps is a service provided by Microsoft Azure that allows you to run and manage containerized applications without having to worry about the underlying infrastructure. It simplifies the process of deploying and managing applications in a containerized environment.

Azure Container Apps can be used in a variety of scenarios, as you can see on the Fig. 2, including:

- Running web applications: You can use Azure Container Apps to run web applications built on any technology stack, including Node.js, Python, PHP, .NET, and more.
- Running microservices: Azure Container Apps can be used to run microservices-based applications, allowing you to easily deploy and manage each service as a containerized application.
- Running batch processing jobs: You can use Azure Container Apps to run batch processing jobs, such as data processing or image rendering, in a containerized environment.
- Running AI and machine learning workloads: Azure Container Apps can be used to run machine learning models and other AI workloads in a containerized environment, providing a scalable and reliable platform for your applications.
- Running IoT workloads: Azure Container Apps can be used to run IoT workloads, such as data ingestion and processing, in a containerized environment, allowing you to easily scale your applications as needed.

Applications built on top of Azure Container Apps can scale dynamically based on the following characteristics:

- HTTP traffic;
- event-driven processing;
- CPU or memory load.

Azure Container Apps allows the execution of application code packaged in any container and is independent of the runtime environment or programming model.

Azure Container Apps manages automatic horizontal scaling using a set of declarative scaling rules. When a container application scales, new instances of the container application are created on demand. These instances are known as replicas. When you first create a container application, the scale rule is set to zero.

Azure Kubernetes Service (AKS) is a managed container orchestration service provided by Microsoft Azure. It allows you to deploy, manage, and scale containerized applications using the open-source Kubernetes orchestration engine.

Kubernetes is the de facto open source platform for orchestrating containers, but typically requires a lot of cluster management overhead. AKS helps manage much of the overhead by reducing the complexity of deployment and management tasks. AKS is designed for users and companies who want to build scalable applications using Docker and Kubernetes using the Azure architecture.

You can create an AKS cluster using the Azure Command Line Interface (CLI), the Azure portal, or Azure PowerShell. Users can also create template-based deployment options using Azure Resource Manager templates.

The main benefits of AKS are flexibility, automation, and reduced management costs for administrators and developers.

Some of the key features of AKS include:

- Automatic scaling: AKS can automatically scale your cluster based on the demand for your applications. This allows you to handle sudden increases in traffic without having to manually add more resources.
- High availability: AKS provides built-in high availability features, such as automatic node replacement and node redundancy, which ensure that your applications remain available even in the event of node failures.
- Security: AKS provides a secure environment for your containerized applications, with features such as network security policies, private cluster access, and integration with Azure Active Directory.
- Integration with other Azure services: AKS integrates with other Azure services, such as Azure Container Registry and Azure Monitor, which allows you to easily manage your containerized applications and monitor their performance.

An AKS deployment also spans two resource groups. One group is just a Kubernetes[7] service resource and the other is a node resource group. A node resource group contains all the infrastructure resources associated with the cluster. A service principal or managed identity is required to create and manage other Azure resources.

A Kubernetes module encapsulates a container and how packages are assembled into nodes. Kubernetes node can contain different pods. For example, Front-end, Back-end, undrelated pods, etc.

Azure Red Hat OpenShift (ARO) is a fully managed container platform offered jointly by Microsoft and Red Hat. It provides a Kubernetes-based platform for developing, deploying, and managing containerized applications as you can see the high-level architecture of Azure Red Hat OpenShift.

Running containers in production with Kubernetes requires additional tools and resources. This often involves juggling image registries, storage management, networking solutions, and logging and monitoring tools, all of which must be versioned and tested together.

Building container-based applications requires even greater integration with middleware, frameworks, databases, and CI/CD[8] tools. Azure Red Hat OpenShift brings it all together in a single platform, making it easier for IT teams while giving teams what they need to get things done. In addition, the Microsoft Azure Red Hat OpenShift service allows you to deploy fully managed OpenShift clusters.

Some of the key features of ARO include:

- Automatic scaling: ARO can automatically scale your cluster based on the demand for your applications. This allows you to handle sudden increases in traffic without having to manually add more resources.
- High availability: ARO provides built-in high availability features, such as automatic node replacement and node redundancy, which ensure that your applications remain available even in the event of node failures.
- Enterprise-grade security: ARO provides a secure environment for your containerized applications, with features such as network security policies, private cluster access, and integration with Azure Active Directory.
- Integration with other Azure services: ARO integrates with other Azure services, such as Azure Container Registry and Azure Monitor, which allows you to easily manage your containerized applications and monitor their performance.
- Support for legacy applications: ARO includes support for running legacy applications, such as databases or other stateful workloads, within the platform, making it easier to modernize your applications.

Design and development of the software solution

Designing an architecture for a service-oriented software system is a complex and important process with a complex subject area. Considering the subject area of the transport rental software system, a multi-tier architecture was chosen for the backend of the web application, and Flux was chosen for the client application.

The server and client application will communicate using the REST API, and the authorization mechanism will also be used. A relational database will be used to store data about transport, users and other subject models.

A relational database will be used to store data about transport, users and other object models.

All services that will be developed will be able to create a Docker image, so that they can then be placed in Docker containers, as shown in Fig. 1.

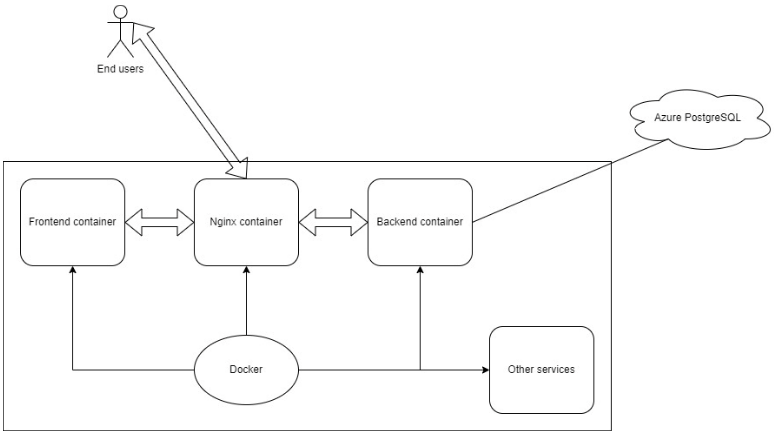


Fig. 1. The structure of Docker containers

The server part and the client part will be in Docker[9] containers, and nginx will handle requests.

Nginx is a web server used as a reverse proxy, load balancer, mail proxy and HTTP cache. Nginx[10] will redirect the request from the end user to the desired container.

Creation of Docker images will be done via Dockerfile.

A Dockerfile is a text document that contains all the commands that a user can call at the command line to build a Docker image. The Dockerfile will include the core files for building and configuring the project.

Building Docker images can be done locally for local testing and through a CI/CD process. If this is done through CI/CD, then in this case, the creation of Docker images should be done as the last step to check the service for correct execution of the main functionality.

To carry out planned studies of Azure services, it is necessary to build a prototype of a software system using design patterns, technologies, and certain architectural styles. This software system will be responsible for renting bikes, scooters, and cars and providing them in Ukrainian cities.

Backend: On the server side, we use the SOA architecture as shown in Fig. 2. The architecture contains 2 services: core service, mail service. These services communicate through the Azure Service Bus using the queue. Also, these services use the same database, Azure PostgreSQL. Each service is responsible for specific business logic. The core service includes the primary logic for renting transports. The mail service is responsible for sending emails to users to notify users, sending verification codes after registration, etc. Communication between services also uses protobuf to standardize sending and receiving data. Protocol Buffers (Protobuf) is a free and open-source cross-platform data format used to serialize structured data. It is useful in developing programs to communicate with each other over a network or for storing data. Also, communication between services and clients uses the API Gateway to hide requests to the services and provide user-friendly API to end users.

To start developing the business logic, it is first necessary to design the use cases of the system for different users using UML.

During the development of the system, a UML diagram was created, this diagram represents the roles of the users of the system, as well as information about the functionality available to each role, thanks to which you can get an idea of what the end user can get.

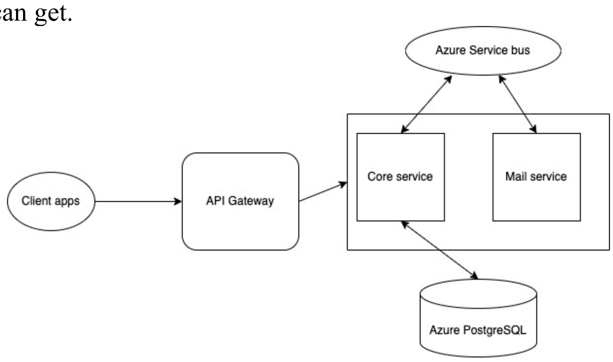


Fig.2. The backend architecture

Fig.3shows a capability diagram for a vehicle rental software system.

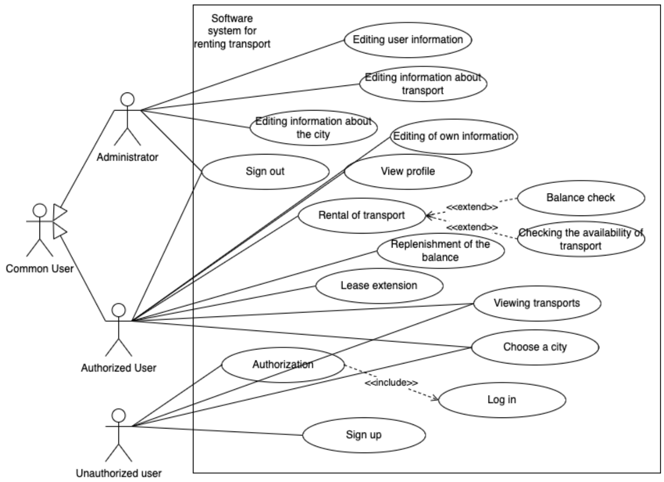


Fig. 3. UML for the software system

All system users are divided into two main roles:
- authorized user;
- unauthorized user.

This software system includes 2 roles of an authorized user:
- system administrator;
- authorized user.

Each role has its own restrictions on the use of functionality. A user who has the role of an unauthorized user can authorize in the system, that is, register or log in to the system. You can also view transport and select a city for further transport selection.

A user with the role of an authorized user can view transport, view a city, rent a vehicle, view his profile, edit his data and top up his balance.

A user with the role of an administrator manages user data, transport and cities.

We will use PostgreSQL[11] for data storage. The PostgreSQL database is compatible with several major programming languages and protocols, including C, C++, Go, Perl, Python, Java, .Net, Ruby, ODBC, and Tcl. This means that users will be able to work in the language they know best without the risk of system conflicts. To work with this database, we will use Azure Database for PostgreSQL server.

The services will be built on a multi-tier architecture to divide the logic. One of the biggest advantages of a multi-tier architecture is that new functionality can be easily added to the system. Making changes to one of the system levels will not affect the modification of the system components in any way if the interaction goes through interfaces and isolation of the model from other components.

In the component diagram, as shown in Fig. 4, you can see parts for each service and how services communicate with each other.

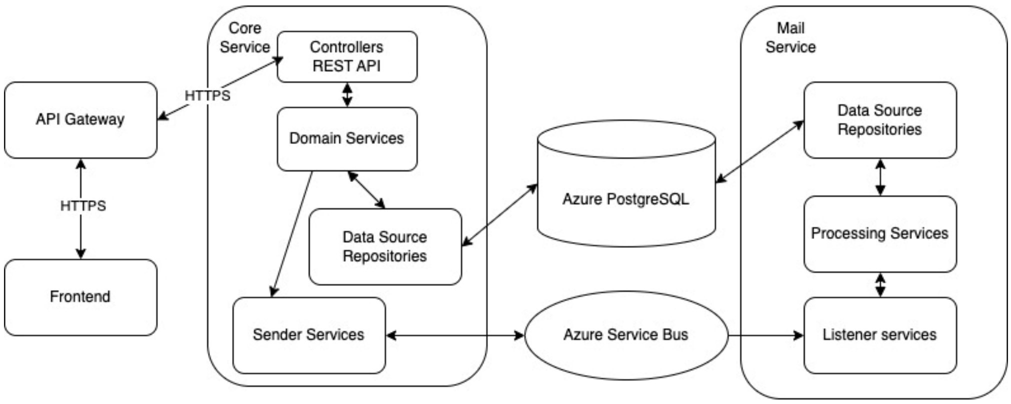


Fig.4. Component diagram

The core service contains four layers: controllers, domain services, data source, and sender services. Domain services can communicate with repositories and sender services. The sender services send messages to the Azure Service Bus, and others receive these messages. In this case, the Mail service contains the listener services to receive notifications from the Azure Service Bus. After getting messages, it starts processing them through the processing services responsible for providing a specific logic for sending mail to users. The data source layer includes repositories and communicates with Azure PostgreSQL.

The mail service contains three layers: listener services, processing services and data source. The listener services receive messages from the Azure Service Bus and start processing them using the processing services. The processing services define a pattern for sending emails. It can be password confirmation, suggestions of the day/week, new updates.

For implementing core and mail services we choice the Java and the Spring Framework. This framework provides rapid development for writing the server part of the system. To make the communication of levels as simple as possible, the principle of injection through the Spring Framework, namely the Spring Boot and Spring Core modules, will be used. For services to work on different devices using the required versions of the libraries, the Gradle automatic assembly system was chosen. Also, thanks to the Gradle configuration, it is possible to break the structure of the server application into separate modules that will be responsible for separate levels of the architecture.

Frontend: To implement the client part, the TypeScript programming language and the React library were chosen. TypeScript will provide us with data typing, which will facilitate the scaling of this software system, and thanks to the React library, we will be able to create components that will be used in the implementation of the Flux architecture. The Flux architecture includes components such as Dispatcher, Store, React Views, and Action Creators.

The Flux architecture imposes restrictions on the flow of data, in particular, excluding the possibility of updating the state of components themselves. This approach makes the flow of data predictable and makes it easier to trace the causes of possible errors in the software.

Analysis of methos for creating a service-oriented software system in Azure using a software solution

After implementing the software system, we are going to start compare Azure services using the scales:
- cost and pricing;
- features and functionality;
- speed of deployment;
- support container registries;
- monitoring and logging support;

Cost and pricing

For calculating we used the official pricing calculator - https://azure.microsoft.com/en-us/pricing/calculator/ and calculated prices for 1 month. After researching we collected the main characteristics of each service and used the results to create table 1 with the cost and pricing for Azure services.

Table 1

Cost and pricing services

Service name	**Azure Container Apps**	**Azure Kubernetes Services**	**Azure Red Hat OpenShift**
vCPU	4	4	4
Memory	16 GB	16 GB	16 GB
Requests per month	60 million	-	-
Nodes	-	4	4
Temporary storage	-	150 GB	-
Master nodes	-	-	8 vCPU, 32 GB RAM
Price per month	771.04$	744.60$	2545.87$

Features and functionality

After researching we determined some features and functionality for Azure Container Apps, Azure Kubernetes Service and Azure Red Hat OpenShift and represented them in table 2.

Table 2

Features and functionality

	Service names		
Azure Container Apps	1.Docker image support 2. Auto-scaling 3.Integration with various Azure services 4.Network Configuration 5. Security protection	6. Monitoring and logging 7. Support for multiple programming languages 8. Scaling services 9. Deployment speed 10. Versioning	11.Support for different types of containers 12. Cluster Management 13. Backup and Restore 14.Support for different operating systems
Azure Kubernetes Services	1. Optimized Kubernetes Management 2. Auto-scaling 3. Docker image support 4. Integration with AzureMonitor 5. Integration with Azure Active Directory 6. Security protection 7. Integration with Azure DevOps 8. Support for multiple programming languages	9. Scaling services 10. Deployment speed 11. Versioning 12. Support for different types of containers 13. Cluster Management, 14. Backup and Restore 15. Automatic recovery 16. Remote access 17. Integration with Azure Policy	18. Flexibility 19. Cross-platform support20. Integration with Azure Arc 21. Integration with Azure Policy for Kubernetes 22. Network management 23. Integration with Azure Private Link 24. Support for different types of accounts
Azure Red Hat OpenShift	1. Kubernetes-based container orchestration	15. Automated testing and quality assurance	28. Kubernetes Operators for automated application management

	2. Red Hat Enterprise Linux operating system 3. Integration with Azure services 4. Enterprise-grade security 5. High availability and disaster recovery options 6. Scalability and elasticity 7. Monitoring and logging 8. Resource utilization tracking and optimization 9. Automated container builds and deployments 10. Multi-cluster management 11. Application templates and deployment patterns 12. Developer tools and SDKs 13. Integrated development environment integration 14. Application lifecycle management	16. Continuous integration and deliverypipeline 17. Git integration and version control 18. Application scaling and load balancing 19. Networking and service mesh capabilities 20. Role-based access control 21. Compliance and audit logging 22. Integration with external identity providers 23. Customizable policies and quotas 24. Resource tagging and management 25. Secure image registry and distribution 26. Integration with third-party registries 27. Application portability across hybrid cloud environments	29. Full-stack observability with Prometheus and Grafana 30. Distributed tracing with Jaeger 31. Logging with Elasticsearch, Fluentd, and Kibana 32. Integration with Azure DevOps for end-to-end software development 33. Container-native storage 34. Data persistence options 35. Serverless computing with Azure Functions 36. Artificial intelligence and machine learning services 37. Edge computing and IoT integration 38. Compatibility with Red Hat OpenShift ecosystem and marketplace 39. Flexible pricing and billing options 40. Enterprise-level support and service level agreements

Support container registries

After getting results we can represent table 3 for showing the number of container registries that Azure services support.

Table 3

Support container registries

Service name	Azure Container Apps	Azure Kubernetes Services	Azure Red Hat OpenShift
Docker Hub	+	+	+
Azure Container Registry	+	+	+
GitHub Container Registry	+	+	+
Amazon Elastic Container Registry	+	+	+
Google Cloud Registry	+	+	+
Harbor Registry	+	+	+
JFrog Container Registry	+	+	+
Quay.io	+	+	+
Red Hat Quay	+	+	+
IBM Cloud Container Registry	+	+	+
GitLab Container Registry	+	+	+
Artifactory		+	+
Quay Enterprise		+	+
VMware Harbor Registry		+	+
Oracle Cloud Infrastructure Registry			+
Azure Stack Hub Container Registry			+

Monitoring and logging support

After researching we determined monitoring and logging features for Azure Container Apps, Azure Kubernetes Service and Azure Red Hat OpenShift. Monitoring and logging features help to monitor and diagnose the state of the app to improve performance. Azure Monitor for monitoring and analyzing metrics, logs, and traces of distributed applications. Azure Log Analytics to collect, analyze and visualize logs from various sources in Azure, including container logs. Azure Application Insights for monitoring and analyzing the performance of applications running in containers. Kubernetes Dashboard for visualizing the status of a Kubernetes cluster and its components. Azure Log Analytics to collect, analyze and visualize logs from various sources in Azure, including container and Kubernetes cluster logs. Prometheus for collecting, monitoring, and analyzing Kubernetes and container metrics. Grafana for visualizing monitoring data from Prometheus and other sources. Kibana for visualizing logs collected using Elasticsearch and Logstash. Elasticsearch for storing and indexing logs from various sources. Fluent to collect and forward logs from containers to Elasticsearch or other storage. Jaeger for tracing distributed applications and identifying problems in the interaction between application components. OpenShift Console for visualizing the state of the cluster and its components. Istio for managing network traffic between application components and protecting against security threats between microservices. Other descriptions of these technologies you can read in the previous paragraphs.

After getting results for monitoring and logging we can represent table 4 for showing the number of monitoring and logging functionality that Azure services support.

Table 4

Monitoring and logging support

Servicename	Azure Container Apps	Azure Kubernetes Services	Azure Red Hat OpenShift
Azure Monitor	+	+	+
Azure Log Analytics	+	+	+
Azure Application Insights	+		
Kubernetes Dashboard		+	
Prometheus		+	+
Grafana		+	+
Kibana		+	+
Elasticsearch		+	+
Fluentd		+	+
Jaeger		+	+
OpenShift Console			+
Istio.OpenShift Console			+

Speed of deployment

For comparing speed of deployment we pushed our system to GitHub repositories to deploy them on the different services. For deploying we will use pipelines to deploy new images to Azure Container Apps, Azure Kubernetes Service and Azure Red Hat OpenShift. We will run 10 times pipelines for each services. In the case for Azure Red Hat OpenShift, we will use Argo CD to configure GitOps processors, one of which is the image deployment. We got average results for each service: Azure Container Apps is 80 sec, Azure Kubernetes Service is 207 sec and Azure Red Hat OpenShift is 220 sec.

Analysing results of deployments, we can create diagram to visualize the results as shown in Fig. 5.

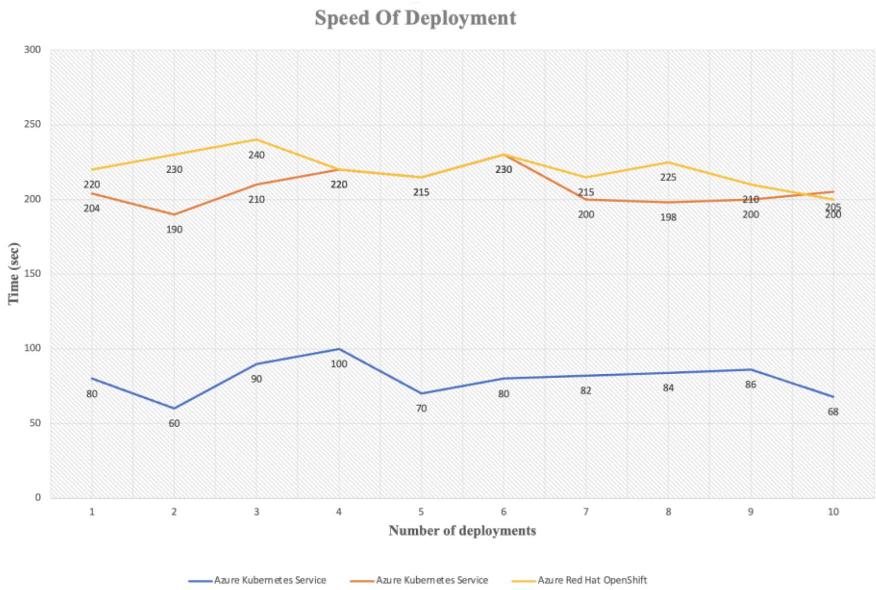


Fig. 5. Speed of deployments

After analyzing methods of creation, we can create a table with results as shown in table 5.

Table 5

Results of analyzing methods

Services/Scale	Cost and pricing ($)	Features and functionality (amount)	Speed of deployment (sec)	Support container registries (amount)	Monitoring and logging support (amount)
Azure Container Apps	771.04	14	80	11	3
Azure Kubernetes Service	744.60	24	204	14	9
Azure Red Hat OpenShift	2545.87	40	220	16	10

Conclusions

The article explored the study of methods for creating service-oriented software systems in Azure. Based on the results of the experiment, the following conclusions can be drawn. In terms of cost and pricing, Azure Container Apps is the most cost-effective option, while Azure Red Hat OpenShift is the most expensive. However, the cost difference is largely due to the advanced features and capabilities offered by Azure Red Hat OpenShift, which may be necessary for more complex applications or large-scale deployments.

In terms of features and functionality, Azure Red Hat OpenShift offers the most advanced set of features, including large-scale cluster support, advanced security support, and integrated CI/CD tooling. Azure Kubernetes Service offers a comprehensive set of features and functionality, while Azure Container Apps provides a simpler, more streamlined option for containerized application deployment.

In terms of support for container registries, all three services offer support for a wide range of container registries, with Azure Container Apps and Azure Kubernetes Service offering similar options and Azure Red Hat OpenShift offering a slightly wider range of options.

In terms of monitoring and logging support, Azure Kubernetes Service offers the widest range of tools, while Azure Red Hat OpenShift offers advanced security features and integrations.

In terms of speed of deployment, all three services offer fast and efficient deployment of containerized applications, with Azure Red Hat OpenShift offering the most advanced features for pipeline automation and integrated CI/CD tooling.

Ultimately, the best choice for a particular project will depend on the specific needs and requirements of the application being developed, as well as factors such as budget, development team experience, and existing infrastructure. After receiving and analyzing the results, we can see such prospects as choosing an Azure service for a certain budget and business needs for building a new system.

References

1. Jain K., Aggarwal P. A Systematic Review of Service-Oriented Architecture: Benefits, Challenges and Road Ahead // International Journal of Computer Applications, 2018. Vol. 182. No. 24. 31-40 c.
2. Sivaramakrishnan, K., & Gokulakrishnan, R. A survey on microservices architecture: Challenges and benefits. In: 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), 2019. 369-373 c.
3. Xu, J., Wang, Q., & Liu, D. Design and Implementation of a RESTful Web Service for Intelligent Agriculture. In: 2019 12th International Symposium on Computational Intelligence and Design (ISCID), 2019. 67-70 c.
4. Azure Container Apps. URL: https://azure.microsoft.com/en-us/products/container-apps.(дата звернення: 10.02.2023)
5. Azure Kubernetes Service. URL: https://azure.microsoft.com/en-us/products/kubernetes-service.(дата звернення: 10.02.2023)
6. Azure Red Hat OpenShift. URL: https://azure.microsoft.com/en-us/products/openshift.(дата звернення: 10.02.2023)
7. Лукса М. Kubernetes в дії / пер. з англ. А. Мартиненко. К.: Видавництво Маннінг, 2018. 372 с.
8. CI/CD for containers / URL: https://learn.microsoft.com/en-us/azure/architecture/solution-ideas/articles/cicd-for-containers. (дата звернення: 15.02.2023)
9. Liu, J., Li, L., Ma, J., & Wang, J. Docker-Based Microservice Architecture of Financial Logistics Management System. In: Proceedings of the 2nd International Conference on Modern Educational Technology and Management Science. Atlantis Press, 2021. 299-304 c.
10. Тоттен Р. Основи Nginx: крок за кроком до володіння базовими можливостями Nginx в реальних додатках / пер. з англ. К. Батигіна. - М.: Видавництво "Пакт", 2019.194 с.
11. Thomas, S. M., & Riggs, S. PostgreSQL 12 High Availability Cookbook: Over 100 recipes to design a highly available server with the advanced features of PostgreSQL. Packt Publishing, 2020. 462 c.

Oleksii Makieiev **Олексій Макєєв**	Bachelor, Kharkiv National University of Radio Electronics, e-mail: oleksii.makieiev@nure.ua https://orcid.org/0009-0002-7909-4793	бакалавр, Харківський національний університет радіоелектроніки
Natalia Kravets **Наталя Кравець**	Ph.D, Associate Professor of Department of Software Engineering, Kharkiv National University of Radio Electronics, e-mail: natalia.kravets@nure.ua https://orcid.org/0000-0002-6753-3333, ResearcherID: B-7312-2019	кандидат технічних наук, доцент кафедри програмної інженерії, Харківський національний університет радіоелектроніки

https://doi.org/10.31891/csit-2023-2-6
UDC 004.91

Liudmyla NECHVOLODA, Kateryna KRYKUNENKO, Katerina PARAMONOVA
Donbas State Engineering Academy

APPLICATION OF A MATHEMATICAL MODEL FOR THE GENERATION OF SEPARATE ELEMENTS OF A STEGANOGRAPHIC SYSTEM IN A HIGHER EDUCATION INSTITUTION

Information protection is extremely important not only in the commercial, but also in the state sphere. The Law of Ukraine "On the National Security of Ukraine" among the threats to the national interests and security of Ukraine in the information sphere indicates: computer terrorism and crime; disclosure of secret or confidential information that is the property of the state or is aimed at ensuring the needs and national interests of society and the state; manipulation of public consciousness, in particular, by spreading false information. There is a need to protect various information systems, in particular local networks of state and commercial institutions, from the threat of information leakage and copyright infringement.

The article presents the method of generating separate elements of the steganographic system based on the combination of cryptography and steganography methods, which makes it possible to increase the level of information protection and develop more effective new non-traditional methods of ensuring information security in the global network. Considering the constant development and improvement of computer cryptography and steganography methods, the study of this particular area of steganoanalysis is the most relevant.

It is proposed to apply the method of replacing the least significant bits (LSB method), because it, in combination with the RSA cryptographic algorithm, allows to ensure a high level of information security and the speed of embedding and extraction of a large amount of information.

The practical value lies in the ability to quickly generate steganographic containers and ensure reliable encryption and decryption of hidden information in them. At the stage of experimental research, the proposed method of replacing the younger bits was compared with other methods that could be used to generate individual elements of the steganographic system. According to research results, the LSB method has clearly confirmed its effectiveness. The results of the experiment showed the high stability and quality of the received data encryption and decryption method when sending it through an open communication channel (e-mail).

Keywords: steganographic system, cryptography, encryption, decryption, steganoanalysis, fractals, RSA, LSB

Людмила НЕЧВОЛОДА, Катерина КРИКУНЕНКО, Катерина ПАРАМОНОВА
Донбаська державна машинобудівна академія

ЗАСТОСУВАННЯ МАТЕМАТИЧНОЇ МОДЕЛІ ДЛЯ ГЕНЕРАЦІЇ ОКРЕМИХ ЕЛЕМЕНТІВ СТЕГАНОГРАФІЧНОЇ СИСТЕМИ У ЗАКЛАДІ ВИЩОЇ ОСВІТИ

Захист інформації є надзвичайно важливим не тільки в комерційній, але й у державній сфері. Закон України "Про національну безпеку України" серед загроз національним інтересам і безпеці України в інформаційній сфері зазначає: комп'ютерний тероризм та злочинність; розголошення таємної або конфіденційної інформації, яка є власністю держави або спрямована на забезпечення потреб та національних інтересів суспільства і держави; маніпулювання суспільною свідомістю, зокрема, шляхом поширення неправдивої інформації. Існує необхідність захисту різних інформаційних систем, зокрема, локальних мереж державних і комерційних установ, від загрози витоку інформації та порушення авторських прав.

У статті представлено методику генерації окремих елементів стеганографічної системи на базі об'єднанні методів криптографії та стеганографії, що дає змогу підвищити рівень захисту інформації та розробити більш ефективні нові нетрадиційні методи забезпечення інформаційної безпеки у глобальний мережі. Зважаючи на невпинний розвиток та вдосконалення методів комп'ютерної криптографії та стеганографії дослідження саме цього напрямку стеганоаналізу є найбільш актуальним.

Запропоновано застосувати метод заміни молодших біт (LSB- метод), оскільки він, у комбінації з криптографічним алгоритмом RSA, дає змогу забезпечини високий рівень інформаційної безпеки та швидкість вбудовування і вилучення великого об'єму інформації.

Практична цінність полягає в можливості швидкої генерації стеганографічних контейнерів та забезпеченні надійного шифрування і дешифрування прихованої інформації в них. На етапі експериментальних досліджень запропонований метод заміни молодших біт був порівняний з іншими методами, що могли б використовуватись для генерації окремих елементів стеганографічної системи. За результатами досліджень метод LSB підтвердив наочно свою ефективність. Результати експерименту показали високу стійкість та якість отриманої методики шифрування та дешифрування даних при пересиланні їх відкритим каналом зв'язку (електронною поштою).

Ключові слова: стеганографічна система, криптографія, шифрування, дешифрування, стеганоаналіз, фрактали, RSA, LSB

Introduction

Information protection is extremely important not only in the commercial, but also in the state sphere. The Law of Ukraine "On the National Security of Ukraine" [1] among the threats to the national interests and security of Ukraine in the information sphere indicates: computer terrorism and crime; disclosure of secret or confidential information that is the property of the state or is aimed at ensuring the needs and national interests of society and the state; manipulation of public consciousness, in particular, by spreading false information. There is a need to protect various information systems, in particular local networks of state and commercial institutions, from the threat of information leakage and copyright infringement.

The methods of criminals are diverse, therefore the information security service must conduct a constant search for information leakage channels, monitor the goals and actions of criminals [2].

Two methods of software protection are traditionally used to preserve the confidentiality of information when it is sent over open communication channels: cryptographic and steganographic protection methods.

The essence of cryptographic protection is that information is encrypted by a certain algorithm into an unreadable format. In turn, steganographic protection is the concealment of the very fact of the existence of information by embedding it in digital objects (containers), which causes some distortion of these objects. The most common types of such containers are text, images, audio data, video sequences.

Today, methods of probability theory and mathematical statistics, theory of fast orthogonal transformations, theory of approximation, theory of coding, theory of complexity, theory of errors, digital processing of signals and images, etc. are widely used as tools for the development of this field.

Both cryptographic and steganographic approaches have their advantages and disadvantages. A promising direction of software information protection is the combination of cryptography and steganography methods. Considering the continuous development and improvement of computer cryptography and steganography methods, the study of this direction of steganoanalysis is the most relevant [3].

Quantitative assessment of the resistance of a steganographic system to external influences is a rather difficult task, which is usually implemented by methods of system analysis, mathematical modeling, or experimental research.

A reliable steganosystem solves two main tasks: hiding the very fact of the message's existence (the first level of protection); preventing unauthorized access to information by choosing an appropriate method of hiding information (second level of protection). The existence of a third level of protection is possible - preliminary cryptographic protection of the message (encryption).

Related works

A significant part of research in the field of steganography is devoted to methods of hiding confidential messages and digital watermarks in still images. Currently, there are a large number of methods of hiding information in graphic files.

A classic example of replacement methods in the spatial domain is the method of replacing the least significant bits (LSB method), which is based on the fact that the least significant bits of graphic, audio and video formats carry little information and their change practically does not affect the quality of the transmitted image or sound. This makes it possible to use them to encode confidential information [4].

In work [5], the main advantage of this method is the simplicity of implementation, the high speed of message embedding and extraction, and the possibility of secret transmission of a large amount of information. However, due to the introduction of additional information, the statistical characteristics of the container file are distorted, and the hidden message is in some cases easy to detect using statistical attacks, such as entropy estimation and correlation coefficients. To reduce compromising features, correction of statistical characteristics is required.

In the scientific publication [6] Taranchuk A.A. the methods operating in the frequency domain are considered, and the data is hidden in the coefficients of the frequency representation of the container. For this, transformations are most often used, which are used in modern lossy compression algorithms (discrete cosine transformation in the JPEG standard and wavelet transformation in JPEG2000). Information can be hidden both in the initial image and simultaneously with the compression of the container image. It is important that stegosystems, which take into account the features of the compression algorithm, are insensitive to further compression of the container.

The essence of broadband methods is to expand the frequency band of the signal to a spectrum width much greater than is necessary for the transmission of real information. There are two ways to expand the range: the method of direct spectrum expansion, using a pseudo-random sequence, and the method of frequency hopping. At the same time, useful information is distributed over the entire range, so when a signal is lost in some frequency bands, there is enough information in other bands to restore it. The principle of operation of broadband methods is related to the tasks solved by stegosystems: to try to "dissolve" a secret message in a container and make it impossible to detect it [7]. Also, in recent years, methods based on image processing have begun to be used for signal detection. In [8], a method for estimating the parameters of transmission from the HFRC based on the spectrogram is proposed. The application of image processing methods to the obtained spectrogram allows you to suppress noise and highlight the necessary parameters.

Statistical methods hide information by changing some statistical properties of the image. In [9], the idea of the Patchwork algorithm is considered, which is based on the assumption that the pixel values are independent and equally distributed. At the same time, a secret key is generated to initialize the generator of pseudo-random numbers, which indicate the place in the image where the watermark bits are entered. This method provides high resistance to digital processing operations, and the difficulty of detecting hidden data without a corresponding secret key.

Thus, the research results show that the reliability of replacement methods in the spatial domain depends on the level of frequency distortions of the container. At the same time, they provide high speed and a significant amount of embedded data, so it is advisable to use them when transmitting hidden messages. Methods operating in the frequency domain are more resistant to distortions and digital processing operations, but can hide a smaller amount of data. The presence of a secret key in broadband and statistical methods using pseudo-random coding increases their reliability.

Proposed mathematical model for the generation of separate elements of a steganographic system in a higher education institution

One of the tasks of information security of every educational institution is to ensure reliable transmission of information between its units. In particular, it can be used to protect students' personal data.

Reliability of data transmission through an open communication channel can be ensured using a steganographic system.

The stegosystem should have the following components:

– message;
– container;
– steganographic channel;
– public and private keys.

The method of secure data transfer through an open communication channel involves the implementation of eight stages.

1. Formation of the student base.
2. Selection of students from the established base.
3. Generation of logins and passwords.
4. Construction of "Plasma" stochastic fractals.
5. Data encryption with the RSA algorithm.
6. Hiding an encrypted message in the lower bits of fractal images using the LSB method.
7. Emailing images to users/students.
8. Decoding and extracting the login and password from the "Plasma" fractal.

For example, generated data: login and password are sent to users of the information system of a higher education institution as an e-mail message. To simplify the work, it is advisable to use email distribution of messages with the help of a previously formed database of students.

The random stochastic fractal "Plasma" can be used as a stegocontainer in this case. The Diamond Square algorithm is most suitable for constructing the "Plasma" stochastic fractal. First, the four corners of the square are assigned random values. After the boundaries of the square are set, it is divided into four equal squares, in each of which the value of one of the angles is known. The value of the height of the central point is the sum of the averaging of the heights of all four corner points and a random value (noise). In practice, the noise should not be completely random, but a function that depends on the distance between neighboring points, because the smaller the distance, the smaller the average value of the noise should be [10].

To ensure greater reliability, along with hiding data, it is advisable to apply data encryption, for example, using the RSA method. The RSA algorithm consists of 4 stages: key generation, encryption, decryption, and key distribution. The security of the RSA algorithm is built on the principle of the complexity of the factorization of integers. The algorithm uses two keys - public and private, together the open and corresponding private keys form key pairs. The public key does not need to be kept secret, it is used to encrypt data. If the message was encrypted with a public key, it can only be decrypted with the corresponding private key.

In order for the Sender to be able to send his secret messages, he transmits his public key (n,e) to the Receiver through a secure, but not necessarily secret, route. The private key d is never distributed. In order to generate key pairs, the following actions are performed:

1. Selecting from two large prime numbers p and q are approximately 512 bits long each.
2. Calculation of their product $n = pq$.
3. Calculation of the Euler function $\varphi(n)=(p-1)(q-1)$.
4. Choosing an integer e such that $1<e< \varphi(n)$ and e is reciprocally prime to $\varphi(n)$.
5. Finding a number d such that $d\equiv 1(\text{mod } \varphi(n))$ using the extended Euclid algorithm.

The number n is called the modulus, and the numbers e and d are called open and closed exponents. Number pairs (n,e) are the public part of the key, and (n,d) are the secret part. The numbers p i q after the generation of the key pair can be destroyed, but must not be revealed in any case.

Algorithm for finding prime numbers:

1. N is an odd number. Finding s and t satisfying the equation: $N - 1 = 2s\ t$.
2. Random selection of the number $a, 1 < a < N$.
3. If N is divisible by a, go to point 6.
4. If the condition $at = 1$ (mod N) is fulfilled, go to point 2.
5. If $k, 0 <= k < s$ such that $a_2k\ t = -1$ (mod N) is found, go to point 2.
6. The number N is composite: choosing another odd number N, going to point 1.

It is important that the number s cannot be greater than the number of bits in the number. The numbers s and t are found using a binary shift of the number $N – 1$, until the lower digit becomes 1. As a result, s is the number of shifts, t is the number $N – 1$ after the shift.

Suppose that the Sender would like to send a message M to the Receiver. First, it transforms M into an integer m such that $0 \leq m < n$ using a consistent reversible protocol known as a complement scheme. It then calculates the ciphertext c using the public key of the recipient e, using the equation:

$$c = m^e \ (mod\ n). \tag{1}$$

This is done quite quickly, even for 500-bit numbers, using modular exponentiation. The Sender then forwards c to the Receiver.

To decipher the message of the Sender m, the Receiver needs to calculate the following equality [11]:

$$m = c^d \ (mod\ n). \tag{2}$$

After encrypting the message, it needs to be hidden in an image. In our case, the encrypted message is hidden in the lower bits of the fractal image (LSB method). The essence of the method of replacing the least significant bit (Least Significant Bits - LSB) is to hide information by changing the last bits of the image, which encode the color, to the bits of the hidden message.

In the BMP format, the image is stored as a matrix of color shade values for each point of the image.

Diamond Square Algorithm:

1. Setting the height at the corner points A, B, C, D (Fig. 1 (a)).
2. Calculation of values at the central point of the square (Fig. 1 (b)):

$$E = \frac{A+B+C+D}{4} + rand_1. \tag{3}$$

3. Calculation of values at the midpoints of the sides (Fig. 1 (c)):

$$F = \frac{A+B}{2} + rand_2. \tag{4}$$

$$G = \frac{A+C}{2} + rand_3. \tag{5}$$

$$H = \frac{B+D}{2} + rand_4. \tag{6}$$

$$I = \frac{C+D}{2} + rand_5. \tag{7}$$

4. Smaller squares AFEG, BFEH, CGEI, DHEI are considered. For them, the steps of the algorithm are repeated until the squares become the required size [12].

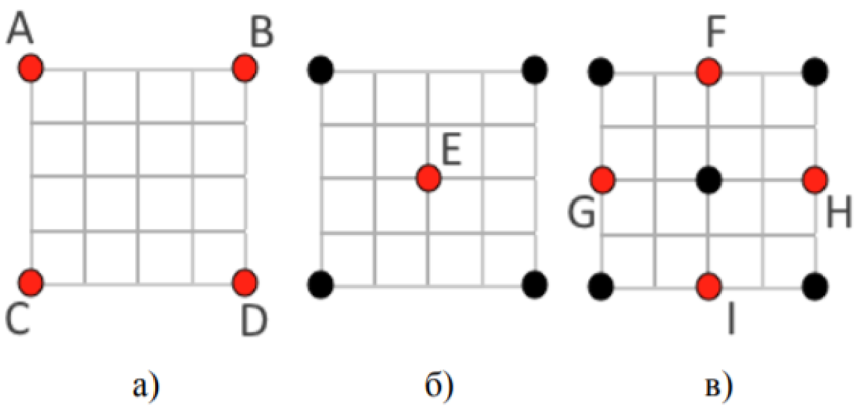


Fig. 1. Diamond Square Algorithm:
a – height in corner points; b – value at the central point;
c is the value at the middle of the sides

The use of the method of replacing the least significant bits (LSB method) in combination with the RSA cryptographic algorithm makes it possible to ensure a high level of information security and the speed of embedding and extraction of a large amount of information.

Experiments

Let's consider the program implementation of the above method using the example of pre-processing of student data for further authorization in the information system of a higher educational institution. The software module for generating individual elements of the steganographic system was developed in Embarcadero RAD Studio 10.2. The program has a user interface in Ukrainian and is divided into three parts: the student database, hiding and data extraction. On the "Data Hiding" tab, the full cycle of actions to obtain a ready-made image with encrypted data for the "Sender" role is performed. Students are selected from the database and will be sent data for entering the personal account [13,14].

The next step in data hiding is to generate the Plasma stochastic fractals using the Generate Fractals button. After displaying the fractals on the form to generate logins and passwords, you need to click the "Generate data" button. Data is invisible to the sender, protected by stars (Figure 2).

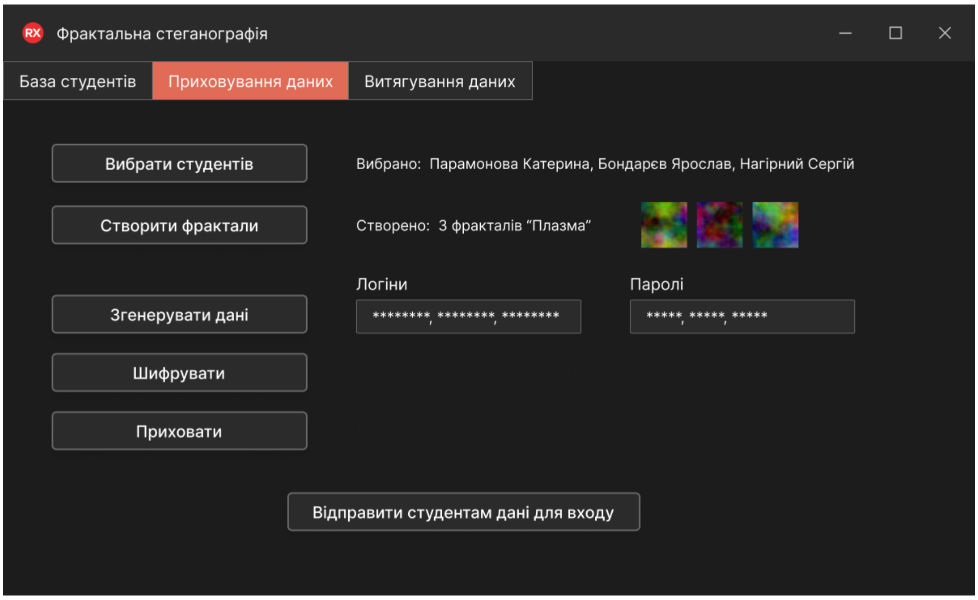


Fig. 2. Generation of logins/passwords and stochastic fractal "Plasma"

To receive ready-made images with encrypted data (data hiding), logins and passwords will be encrypted using the RSA method. To do this, click on the "Encrypt" button. After this action, private keys appear. The private key is important, it is transmitted to the respective student through a secure communication channel. Then data is hidden in the last bits of previously generated fractals using the LSB method, rasterization and image storage (Figure 3).

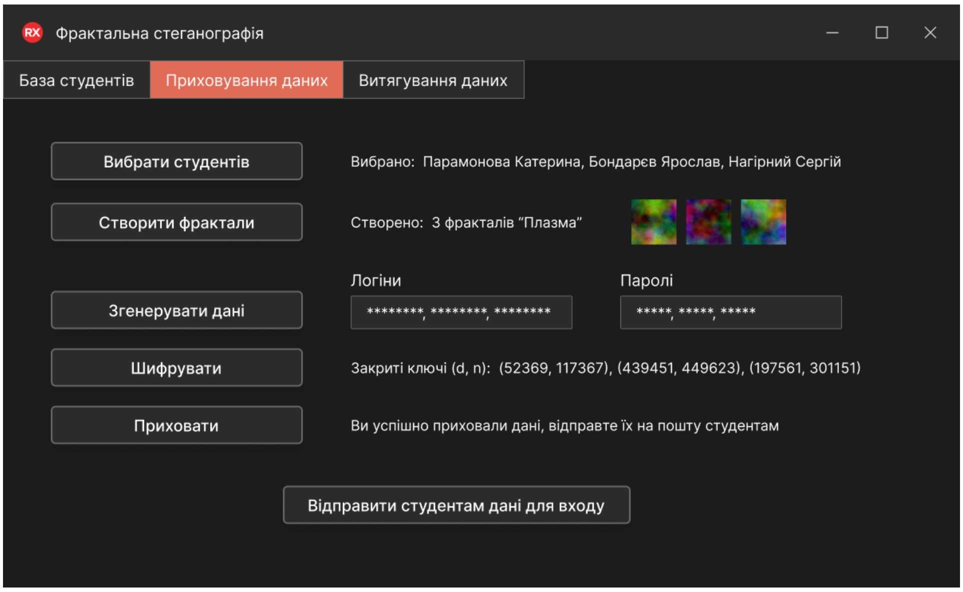


Fig. 3. Hiding data in fractals

To send saved images to selected students by e-mail, click on the "Send login data to students" button. This completes the action cycle for the "Sender" role.

On the "Data extraction" tab, the complete cycle of actions to obtain the login and password for the "Recipient" role is performed. A corresponding message will be sent to the e-mail with an attached image (Figure 4). The message should have: the subject "Fractal", the text of the message: "Login data for "Surname and First name of the student", as well as the attached picture of the stochastic fractal "Plasma".

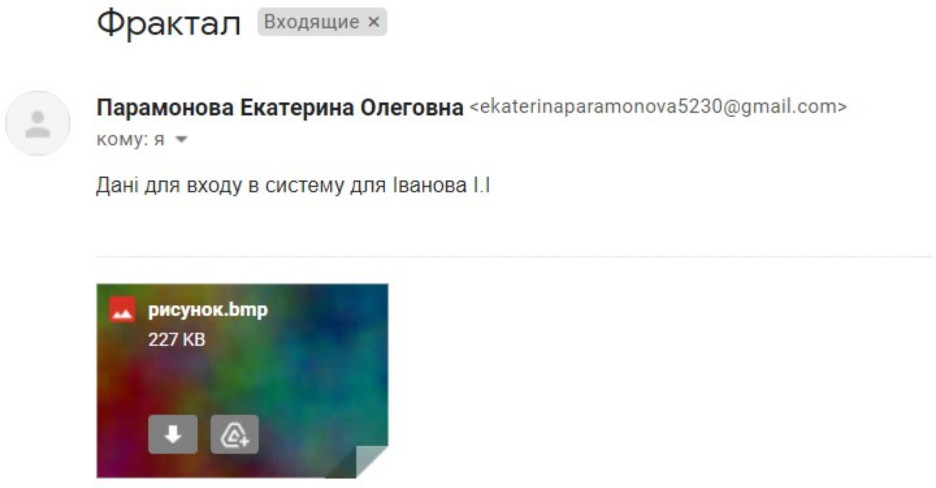


Fig. 4. The message that came to the recipient's e-mail

The next step in extracting data is to enter a private key that is transmitted over a secure communication channel and decrypt the data using that key. After the actions described above, the required initial data will appear in the "Login" and "Password" fields (Fig. 5).

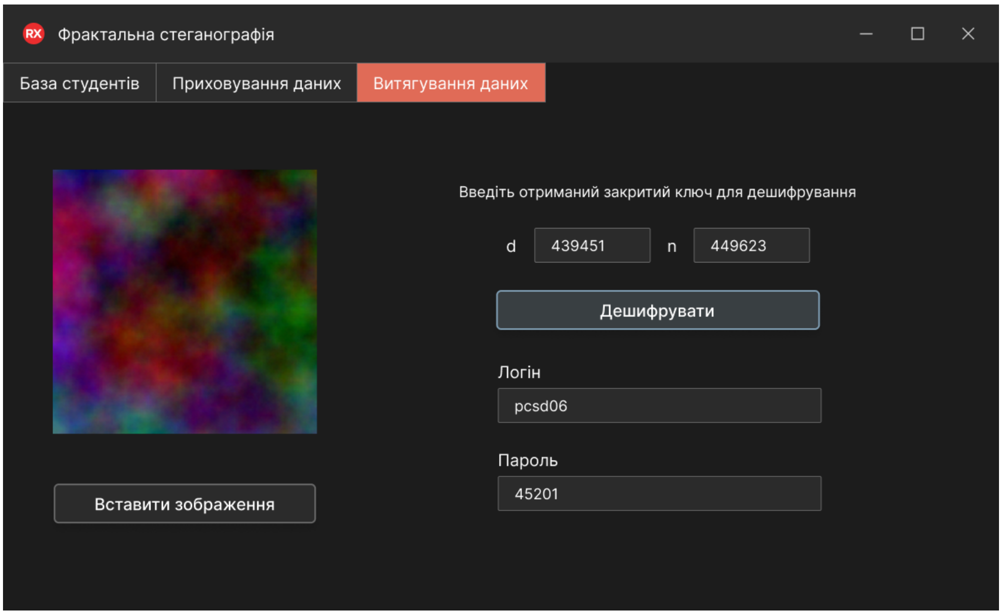


Fig. 5. Data decryption

The given example demonstrates the option of using a steganographic system to protect students' personal data.

Conclusions

In the process of analyzing the quality of the proposed model, it was found that to create a system for generating individual elements of the steganographic system, it is advisable to use the method of replacing the least significant bits (the LSB method), since it, in combination with the RSA cryptographic algorithm, makes it possible to ensure a high level of information security and embedding speed and extracting a large amount of information. In particular, it will make it possible to transmit data over an open channel more securely with minimal risks of errors occurring when transmitting and extracting data from encrypted messages.

The capabilities of the presented algorithm can be used by dean's offices, personnel and practice departments, and other units of the higher education institution to transmit information containing personal data of students. This is especially relevant in the conditions of information warfare.

References

1. Pro natsionalnu bezpeku Ukrainy: zakon Ukrainy vid 2018r № 31, st.241 URL:https://www.mil.gov.ua/diyalnist/reformi-ta-planuvannya-u-sferi-oboroni/zakon-ukraini-pro-naczionalnu-bezpeku-ukraini.html
2. Sorokivska O. A., Hevko V.L. Informatsiina bezpeka pidpryiemstva: novi zahrozy ta perspektyvy [Tekst]: Visn. Khmelnyts. nats. un-tu. Ser.: Ekon. nauky. 2010. № 2. T. 2.
3. Tarnavskyi Yu.A. Tekhnolohii zakhystu informatsii [Elektronnyi resurs] : pidruchnyk dlia stud. spetsialnosti 122 «Kompiuterni nauky», spetsializatsii «Informatsiini tekhnolohii monitorynhu dovkillia», «Heometrychne modeliuvannia v informatsiinykh systemakh». KPI im. Ihoria Sikorskoho. Elektronni tekstovi dani (1 fail: 2,04 Mbait). Kyiv : KPI im. Ihoria Sikorskoho, 2018. 162 s.
4. Khoroshko V. A., Chekatov A.A. Metody y zasoby zakhystu informatsii. K.: Yunior, 2003. 504 s.
5. Kuznetsov O. O., Yevseiev S.P., Korol O.H. Stehanohrafiia : navchalnyi posibnyk. Kh. : Vyd. KhNEU, 2011. 232 s. (ukr. mov.)
6. Taranchuk A. A., Halper L.H. Stehanohrafichnyi metod prykhovuvannia danykh v oblasti chastotnykh peretvoren zobrazhen. Visnyk Khmelnytskoho natsionalnoho Universytetu. Khmelnytskyi, 2009. № 2 «Tekhnichni nauky» C.197-201.
7. Iudin O. K., Konakhovych H.F., Korchenko O.H. Zakhyst informatsii v merezhakh peredachi danykh: Pidruchnyk K.: Vydavnytstvo TOV NVP «INTERSERVIS»,2009. 352 s.
8. Chevva L., Sagar G. V. R. FH Signal Interception Based on the Time-Frequency Spectrogram by Image Enhancement Techniques // International Journal of Engineering Research and Applications, 2012. Vol. 2, Issue 2. R. 687−692.
9. O.B. Polusyn. Metody zabezpechennia zakhystu multymediinykh danykh nanesenniam tsyfrovoho vodianoho znaku Naukovo-tekhnichna konferentsiia profesorsko-vykladatskoho skladu, naukovykh pratsivnykiv i aspirantiv (u rezhymi onlain) (07 liutoho – 11 liutoho 2022 r.). Tezy dopovidei, Lviv. Ukr. akad. drukarstva, 2022. 193 s.
10. Fraktaly [Elektronnyi resurs]. – URL: http://www.kpi.kharkov.ua/archive/microcad/2011/%95%D0%A0%D0%95%D0%94%D0%9E%D0%92%D0%98%D0%A9.pdf.
11. Metody stehanohrafii [Elektronnyi resurs]. – URL: http://masters.donntu.org/2013/fknt/ippolitov/library/article2.htm.
12. Alhorytm Diamond Square [Elektronnyi resurs]. – URL: https://habr.com/ru/post/111538/.
13. Shevchenko N.Iu., Paramonova K.O. Fraktalni konteinery v stehanohrafii. Informatyka, upravlinnia ta shtuchnyi intelekt. Tezy deviatoi mizhnarodnoi naukovo-tekhnichnoi konferentsii. Kharkiv: NTU "KhPI", 2022. 160 s. ukrainskoiu, anhliiskoiu movamy. S. 141.
14. Shevchenko N.Iu., Paramonova K.O. Heneratsiia stehokonteineriv dlia pidvyshchennia informatsiinoi bezpeky kanaliv peredachi danykh. «TAK»: telekomunikatsii, avtomatyka, kompiuterno-intehrovani tekhnolohii: zb. dopovidei Vseukr. nauk.-prakt. konf. molodykh vchenykh, 1-2 hrudnia 2021 r. DVNZ «DonNTU; vidp. red. M.V. Stupak. Pokrovsk: DVNZ «DonNTU», 2021. S. 4–6.

Liudmyla Nechvoloda **Людмила Нечволода**	PhD, Associate Professor of the Department of Intelligent Systems of Decision Making, Donbas State Engineering Academy, Ternopil, Ukraine e-mail: lylyne4v@gmail.com https://orcid.org/0000-0002-7584-6735 ResearcherID: HPB-8625-2023	кандидат технічних наук, доцент кафедри інтелектуальних систем прийняття рішень, Донбаська державна машинобудівна академія, Тернопіль, Україна.
Kateryna Krykunenko **Катерина Крикуненко**	Assistant of the Department of Intelligent Systems of Decision Making, Donbas State Engineering Academy, Ternopil, Ukraine e-mail: ladybabenko87@ukr.net https://orcid.org/0000-0003-1530-216X	асистент кафедри інтелектуальних систем прийняття рішень, Донбаська державна машинобудівна академія, Тернопіль, Україна.
Katerina Paramonova **Катерина Парамонова**	Master of specialty «Information systems and technologies», Donbas State Engineering Academy, Ternopil, Ukraine e-mail: ekaterinaparamonova5230@gmail.com	магістр спеціальності «Інформаційні системи та технології», Донбаська державна машинобудівна академія, Тернопіль, Україна.

https://doi.org/10.31891/csit-2023-2-7
UDC 004.89

Volodymyr SHYMANSKYI, Nikoletta KAZONNIKOVA
Lviv Polytechnic National University

VOICE ASSISTANT FOR FINDING ITEMS IN CLOTHING STORE

The paper presents the use of machine learning in audio file processing, the use of neural networks to recognize voice commands, and the development of a voice assistant for finding products in a clothing store.

The purpose of the work is to develop a voice assistant for finding items in a clothing store.

As a result, a dataset containing 30 categories and 3095 audio recordings was created, a neural network model was trained using the collected data, and an accuracy of 96.02% was achieved; WER: 0.0398; CER: 0.0087. The model was integrated with the search system into the voice assistant API, which allows you to record from a microphone, convert audio to text, break the text into keywords, and search the database using the obtained keywords. The speech recognition system has shown stable and high accuracy when recording from a microphone. This provides users with reliable and accurate recognition results even when using simple microphones. Search allows you to find results by keywords and names of items, and there is a function to recommend similar items if nothing was found at the user's request. The system's flexibility allows it to understand language contexts and does not depend on the order of words in a phrase, as the search is performed by keywords separately. The research analyzes the literature and compares existing approaches to voice command recognition. The methods of audio signal processing are described. The problem of searching for items in a clothing store was analyzed, and the current state of the market and demand for the technology were investigated. The practical value is that the developed voice assistant is specialized and optimized for searching for goods in a clothing store, allows solving the task of finding goods according to specified criteria, and simplifies the search task for different groups of people.

Keywords: machine learning, neural networks, voice recognition, Ukrainian language recognition, voice assistant, product search.

Володимир ШИМАНСЬКИЙ, Ніколетта КАЗЬОННІКОВА
Національний університет «Львівська Політехніка»

ГОЛОСОВИЙ ПОМІЧНИК ДЛЯ ПОШУКУ ТОВАРІВ У МАГАЗИНІ ОДЯГУ

У даній статті досліджено застосування машинного навчання в обробці аудіо файлів, використання нейронних мереж для розпізнавання голосових команд, створенню голосового помічника для пошуку товарів в магазині одягу, створенню набору даних для тренуванння нейронної мережі..

Метою роботи є розробка голосового помічника для пошуку товарів у магазині одягу.

У результаті було створено набір даних, що містить 30 категорій та 3095 аудіозаписів, натреновано модель нейронної мережі з використанням зібраних даних та досягнуто точності 96.02%; WER: 0.0398; CER: 0.0087. Було інтегровано модель разом із системою пошуку в API голосового помічника, який дозволяє зробити запис з мікрофону, конвертувати аудіо в текст, розбити текст на ключові слова та провести пошук по базі даних з використанням отриманих ключових слів. Система розпізнавання мови показала стабільну та високу точність під час запису з мікрофону. Це забезпечує користувачам надійність та точність результатів розпізнавання навіть при використанні простих мікрофонів. Пошук дозволяє знаходити результати за ключовими словами та назвою речей, є функціонал рекомендації схожих речей, у випадку, якщо нічого не було знайдено за запитом користувача. Гнучкість системи дозволяє розуміти мовні контексти та не залежить від порядку слів у фразі, оскільки пошук відбувається по ключовим словам окремо. В роботі було проаналізовано літературу та проведено порівняння існуючих підходів розпізнавання голосових команд. Описано методи обробки аудіо сигналів. Було розглянуто проблему пошуку товарів у магазині одягу та досліджено сучасний стан ринку та попит на технологію.

Практична цінність полягає у тому, що розроблений голосовий помічник є спеціалізований та оптимізований для пошуку товарів у магазині одягу, дозволяє вирішувати задачі пошуку товару за заданими критеріями, спрощує завдання пошуку для різних груп людей.

Ключові слова: машинне навчання, нейронні мережі, розпізнавання голосу, розпізнавання української мови, голосовий помічник, пошук товарів.

Introduction

A voice assistant can be useful for many people with different levels of fashion expertise and different physical abilities. For example, people with disabilities may have difficulty finding the desired items, and for people with dyslexia, a voice assistant will make the search process more comfortable and efficient. Even for average people, searching with a voice assistant can save a lot of time, as there is no need to go around the entire store. In addition, such a voice assistant can partially replace a real consultant, selecting the necessary things for the buyer. Thus, a voice assistant can be useful for different categories of people with different needs. It will help make the process of searching for goods more efficient and comfortable.

The aim of the work is to develop a voice assistant for finding items in a clothing store.

Research objectives:

- research of the subject area;
- analysis of existing approaches and algorithms;
- architecture design and algorithm development;
- development of a system for using the algorithm;
- testing the developed voice assistant and demonstrating the results of its work;

The object of research is the process of searching for goods and voice recognition.

The subject of research is machine learning algorithms and neural networks for processing and recognizing voice commands.

Related works

Developing a voice control system requires an in-depth analysis of existing research and methods for processing voice commands. During the research process, a significant amount of relevant literature was found, which makes it possible to conduct a detailed analysis and use best practices in the development of a voice control system.

For example, an article by Edward L., who investigated methods of detecting voice fraud using machine learning [1]. The final neural network model had an error of 10.8. Vincenzo D. used deep learning to recognize covid-19 through breathing and coughing by feeding the network with audio recordings without pre-processing [2]. Muhammad E. used the k-means algorithm to recognize audio signals [3]. Han G. used deep learning to classify music genres [4]. The model is trained on 5 different genres and recognizes them with an accuracy of 90.32%. Richard M. used convolutional neural networks to classify cough sounds using features such as mel spectrograms and mel-frequency coefficients and achieved an accuracy of 82.96% [5]. Satish K. also conducted research in the field of medicine and used deep learning to monitor the pulse status of patients and achieved 85% accuracy [6]. Elham N. used an ensemble method to detect Parkinson's disease based on patient's voice metrics and achieved 90.6% accuracy [7]. George S. studied the recognition of whole sentences using existing trained models for speech recognition [8]. Eleni T. used processed spectrograms as input to a convolutional neural network and achieved an accuracy of 91.25% [9]. Elizabeth V. conducted a comparative study of audio signal classification using a convolutional neural network and a generative adversarial network [10].

It is also worth noting the work of A.G. Krivokhata, O.V. Kudin, and A.O. Lisnyak, which describes the processing of audio signals using ensemble learning and neural network classifiers [11]. Neural network ensembles are actually effective because it is possible to choose a different number of algorithms, undemanding to resources, and it is possible not to adjust to the quality of the audio recording by choosing a specific algorithm, but to choose the results with the highest accuracy each time.

Presenting main material

The task is to use neural networks to create a voice assistant. There are also subtasks that consist of using algorithms for searching and processing voice commands for the neural network. To create voice assistant we need to define general flow of functionality and to determine implementation steps (Fig. 1).

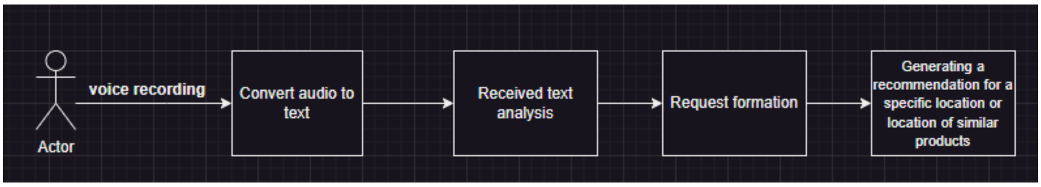


Fig.1 Voice assistant functionality flow

Firstly, we need voice recognition algorithm which can take audio as input and convert it to text. We need to identify a set of words that can be used in voice commands, collect audio data, and train a neural network model on it. Next step is to create a system that can process received text, highlight keywords and form a search query. After that we can use the query to look for certain items in database with search algorithm. Next step is the recommendation system to provide suggestions to the user if nothing was found at their request.

The development of a neural network model can be divided into two stages. The first is, of course, audio pre-processing, such as encoding, splitting audio into frames, or creating mathematical models such as spectrograms, chromagrams, etc. Many of these methods were described in their work by Palanisamy K., Shingania D., Yao A., and explained in which cases it is better to prepare the data in which way [12]. Next, you need to choose the classification method itself, such as the support vector method, k-means, neural networks, or deep learning.

When determining the assistant's functionality, you need to consider whether the store is electronic or physical, whether it is designed to search for specific products, whether it can provide up-to-date information about product availability, discounts, and promotions. In our case, the assistant will be able to include commands of the following type:

- "Where can I find product X?"
- "How much does product X cost?"
- "Find product X in size Y"

One approach to finding products can be to analyze and classify them according to their description and characteristics. With the help of machine learning, you can train algorithms to recognize certain features of products and use this information for search. Such characteristics can be name, color, brand, article, price, category (age, gender, etc.). In addition to classification algorithms, you can use search algorithms such as binary search, search trees. You need algorithms that are fast enough and able to work with large amounts of data.

To train the neural network, a lot of data had to be collected, including audio recordings. First, a set of recognition words was identified that could be used to search for clothes: various command words, such as "find", "show", etc.; words for product names, such as "shirt", "skirt", etc., and various colors. The dataset also contains a variety of complete sentences to train the model to distinguish between cases. A large number of entries of the same word will help to better distinguish certain characteristics of words, which will significantly improve the classification accuracy. In addition, the dataset contains data with different background sounds and mispronunciations. The architecture of the dataset is as follows (Fig. 2)

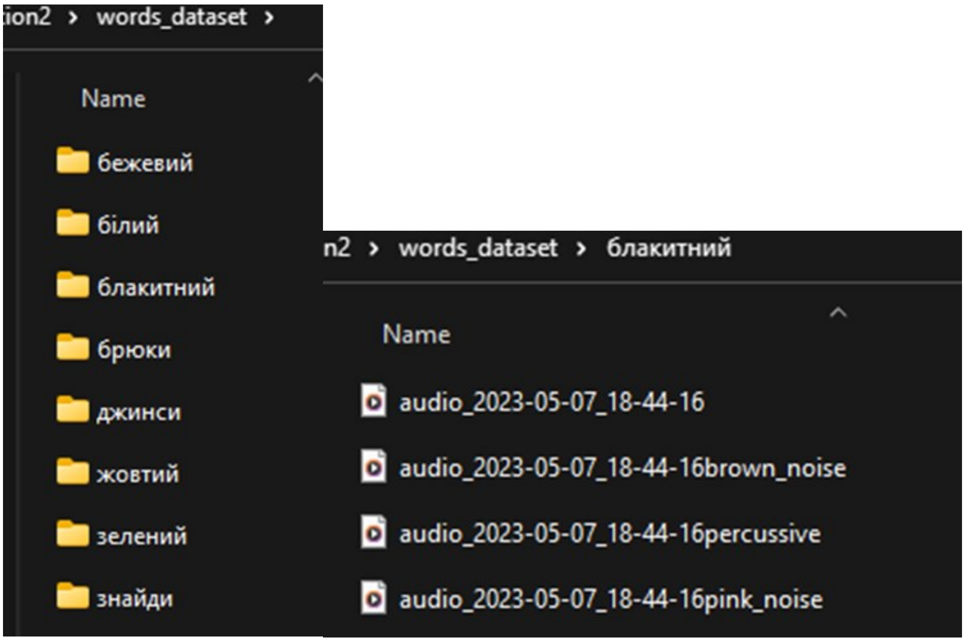


Fig.2. Dataset architecture

As you can see, the data is organized into categories, each category has the name of the recorded command. The files themselves are saved in the ".wav" format, with the date they were recorded and the sequence number in the name. Also, to increase the amount of data, some effect or noise was applied to each recording, which is also indicated in the recording name. The recorded sentences are stored in a separate folder and contain the text of the audio recording in the title.

In total, the dataset contains 30 different categories and a total of 3095 audio files. Each category contains an average of 90-100 records, and the sentence folder contains the most files - 287.

To train on this data, we first need to normalize all the data, i.e., reduce it to a single volume, frequency, and convert it to a single channel (mono) format.

In addition to evaluating the accuracy of speech recognition, it is also important to consider aspects of the development environment and technical requirements associated with the model.

The development environment includes aspects such as programming languages, libraries, and infrastructure required to deploy and use the model. To ensure optimal use of the model, additional technical components may also be required, such as building an API interface for easy interaction.

The functionality of training, improving and testing the model, audio data preprocessing will be developed in Python programming language version 3.8 using such libraries as NeMo, Numpy, Pandas, Torch, Librosa and others. For optimal development, it is recommended to use an integrated development environment (IDE) such as PyCharm or Visual Studio Code, which provide an extended set of tools.

Given the power of the hardware, training will take place on the CPU, although for faster training, such models should be trained on the GPU.

Python will also be used to develop the voice assistant system. The Django framework will be used to develop the API, and HTML, CSS, and JavaScript will be used for the user interface.

It was decided to use a pretrained ASR (Automatic Speech Recognition) model (theodotus/stt_uk_squeezeformer_ctc_ml) provided by Nvidia in its NeMo library. ASR models, also known as STT (Speech To Text) models, are systems that are designed to automatically convert speech to text. These models are used in a variety of applications, including voice command matching, transcription of audio and video files, speech translation, and much more.

ASR models are usually built on the basis of deep neural networks, in particular recurrent neural networks (RNNs), such as LSTM (Long Short-Term Memory) or GRU (Gated Recurrent Unit), or transformers, such as in the Transformer model. The ASR model process can be divided into several stages:

- Audio pre-processing: The audio signal is first subjected to pre-processing, which includes functions such as resampling (changing the sample rate), noise filtering, volume normalization, and possibly other operations to improve the sound quality and reduce the impact of noise.
- Feature extraction: The audio signal is fed to the input of the ASR model, where a feature extraction process is performed. Typically, this process involves converting the audio signal into a spectrogram or other vector representation that can be fed to the neural network.
- Acoustic model: This stage uses a neural network called an acoustic model. It takes as input the feature vectors obtained in the previous step and tries to predict the probability of each input vector belonging to different phonemes or phonetic units. This model is trained on a large amount of labeled audio data with text transcriptions.
- Lexical model: After the acoustic model, the lexical model is used to select the correct word sequence based on the received phonetic sequences. The lexical model can include a dictionary or other resources to help determine the most likely word sequence for a given phonetic sequence.
- Language model: The next step is the language model, which is used to select the most likely text based on the word sequence derived from the lexical model. The language model takes into account the linguistic properties of the language and the context to produce a more natural and understandable textual result.
- Decoding: The last step is decoding, where the most likely text is selected based on the output probabilities of the language model. This may include post-processing, such as error correction or contextual adaptation.

ASR models are trained using large datasets that contain audio recordings and corresponding text transcriptions. Training requires powerful computing resources and a long time. However, with the development of deep learning technologies, ASR models are becoming more accurate and able to work with a variety of speech types and accents.

The processors used to process this model can be dependent on the implementation and hardware used. Typically, GPUs (Graphics Processing Units) or specialized processors for accelerating computation, such as TPUs (Tensor Processing Units), are used to train and execute ASR models. The choice of processor depends on the availability of resources and the size of the task to be performed.

The theodotus/stt_uk_squeezeformer_ctc_ml model used for Ukrainian speech recognition is based on the Squeezeformer architecture and uses the CTC (Connectionist Temporal Classification) method. Figure 3 shows the architecture of the Squeezeformer model [13].

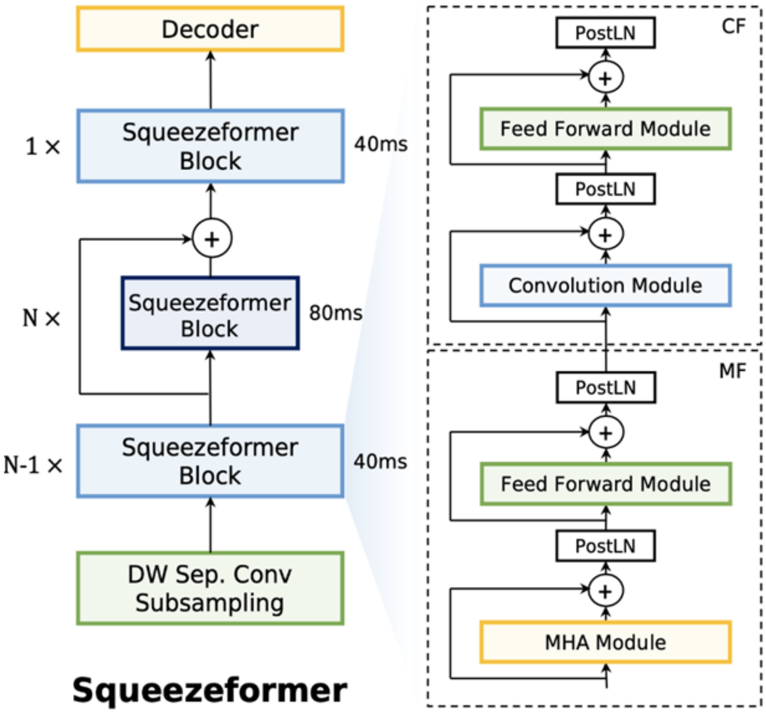


Fig. 3. Squeezeformer model architecture

Squeezeformer is an architecture that combines Squeeze-and-Excitation (SE) and Transformer to model channel importance in incoming audio and sequential dependencies in speech. It uses SE to compress channels and generate weights, and Transformer to model contextual dependencies in speech.

For training and decoding, the EncDecCTCModel class is used, which contains all the necessary functionality for setting up input data, training, and testing. Next described methods that are used in training process.

AudioToMelSpectrogramPreprocessor is a method that converts audio files of the ".wav" format into mel spectrograms. Also, at this stage, input data is processed before feature extraction, for example, resampling or noise filtering

SqueezeformerEncoder is used to implement the encoder. The encoder is used to convert the input features into a compact representation that is used for further speech recognition. These compressed features are then passed to the next layer.

The ConvASRDecoder processes the output features. Next, it determines the probabilities of characters and generates their sequences corresponding to the text of the input audio recording.

CTCLoss implements the CTC loss function to calculate the loss between the recognized character sequences and the actual labels. This loss is used to train the ASR model by reducing the discrepancy between the recognized and true character sequences.

SpectrogramAugmentation allows us to improve the model training by increasing the number of spectrograms.

WERBPE is used to calculate the accuracy of the model.

After the data has been processed and divided into training, validation, and test sets, you need to configure the neural network parameters. The main parameters are:

- Audio frequency(16000kHz)
- Path to the data set
- batch_size – a parameter that determines the number of data samples that are simultaneously submitted to the model before updating its weights(16)
- Number of processors on which the network will be trained(0)

The training will take place for 30 epochs, but functionality will be added to stop the training earlier if there is no loss reduction for three epochs. Since the model takes quite a long time to train, the ability to save the progress at each epoch and save the final model to a ".nemo" file has also been added.

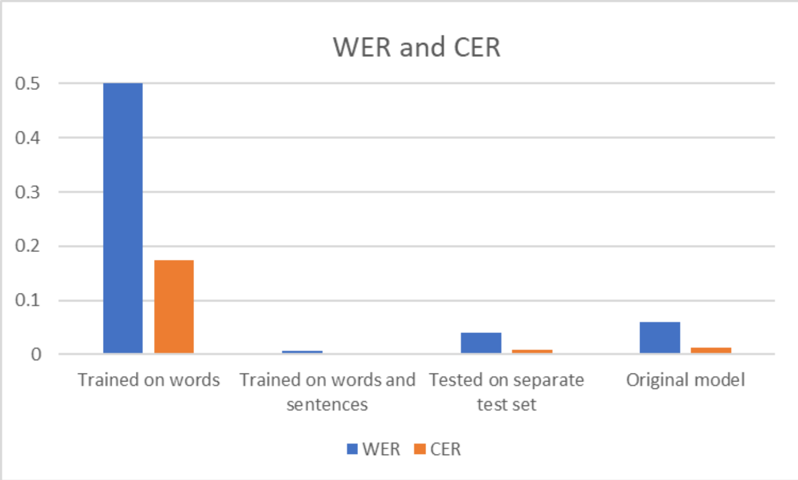
The TensorBoard library was used to determine the training accuracy and validate the model. Also, the model was tested on my and test data to determine the accuracy of the model.

WER (Word Error Rate) and CER (Character Error Rate) are metrics used to evaluate the accuracy of automatic speech recognition (ASR) and optical character recognition (OCR) systems, respectively. Both metrics measure the level of errors in the reproduction of text. WER is defined as the ratio of the total number of insertions, deletions, and substitutions made by the system to the total number of words in the input text. It measures the difference between the recognized text and the correct text at the word level. It is usually expressed as a percentage or as a decimal number. The CER, on the other hand, is defined as the ratio of the total number of insertions, deletions, and replacements made by the system to the total number of characters in the input text. It evaluates the recognition accuracy at the level of individual characters. Like WER, CER is usually expressed as a percentage or as a decimal number. If the WER or CER is close to zero, it means that the recognition system has achieved high accuracy. Values close to 100% indicate low recognition accuracy. WER and CER are useful metrics for comparing different speech recognition systems or evaluating the quality of ASR and OCR models. They allow you to evaluate the efficiency of the system in terms of accuracy and draw conclusions about the quality of its performance.

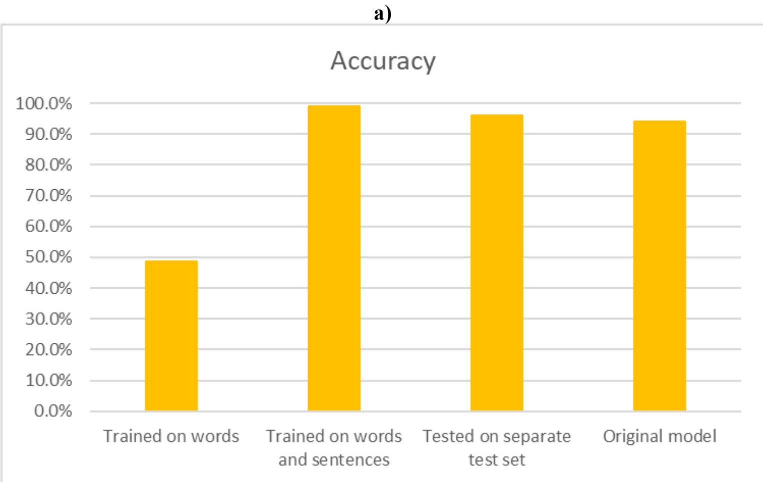
Initially, the model was trained on a set of words only, without full sentences. The training stopped at 7 epochs and had the following accuracy when tested on sentences: WER: 0.5121; CER: 0.1732, which meant that errors were found in almost 50% of the words. The problem was that the model could not recognize word cases. It was then decided to train the model on sentences as well.

This combination of data performed well during training, and in the next attempt the accuracy was like this: WER: 0.0069; CER: 0.0012, which meant that the number of errors was less than one percent. However, the same data was used in testing as in training. A separate test set with sentence recordings was collected for additional testing and the following accuracy was obtained: WER: 0.0398; CER: 0.0087. This test set covers the entire word set.

In general, when testing the model, it was found that the command words are recognized best, and there are not many errors in the names of clothes. Most often, the network makes mistakes on colors when they are not in the nominative case. The final accuracy of the neural network is 96.02%. In the graphs in Figure 4, you can see the comparison of model accuracies in different experiments and the comparison with the original model (the accuracy of the original model was taken from the documentation [13]).



a)



b)

Fig.4 Comparison a) values of WER and CER; b) values of accuracy

As we can see, the trained model on words and sentences shows better accuracy than the original model. Figure 5 shows the loss and wer plots for training and validation.

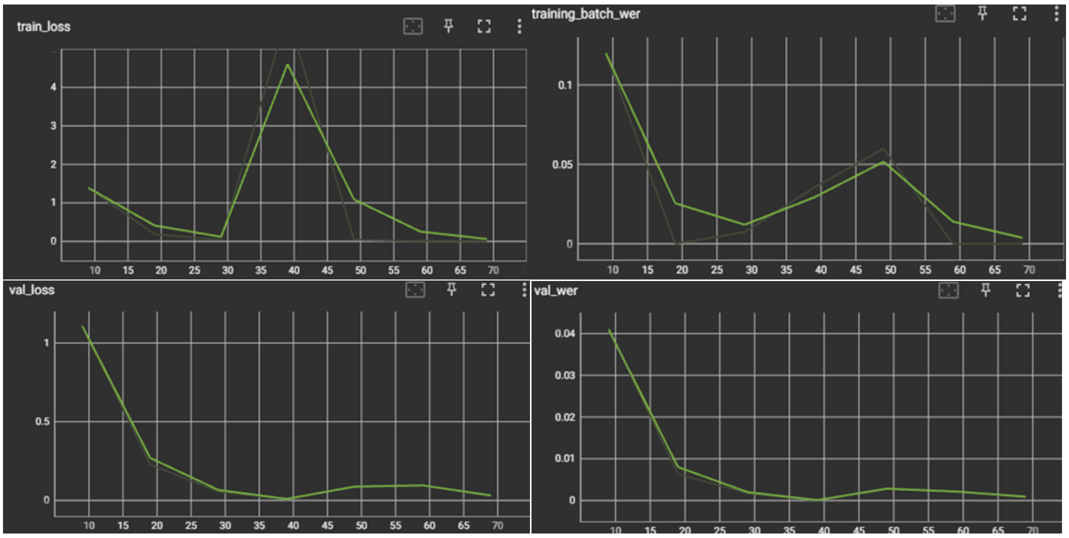


Fig.5 Accuracy and loss graphs for training and validation

We can see that basically the functions are stable, which means that the network is trained well. There is a slight increase in losses in the middle of the training, which may be due to the small size of the dataset, or a small amount of low-quality data that increased the losses. However, we can see that the losses and validation accuracy were steadily approaching zero.

As mentioned above, the voice assistant will be based on the REST API developed on the Django framework. In order to create a voice assistant, the following functionality was created:

- A user interface was created to record audio from the microphone and display the shelf number on the screen;
- The recognize_speech function that receives the recording from the microphone, normalizes it, feeds it to the neural network for transcription, and returns the text;
- The search_keywords function that accepts the text, splits it into words, discards the command words, searches the database, and returns the shelf number.

To pass a record from the interface to the recognize_speech method, the startRecognition JavaScript function was created, which, when you click the "Start" button, sends a request to the API, which in turn launches this method. To use our trained model, the NeMo library will be used to deploy and save the model from a file in the ".nemo" format. Next, the recording from the microphone should be normalized in the same way as for training, i.e., reduced to one channel and set to 16000. Now the audio can be fed to the neural network to be converted to text and passed to the search_keywords method.Figure 6 shows how the voice assistant works.

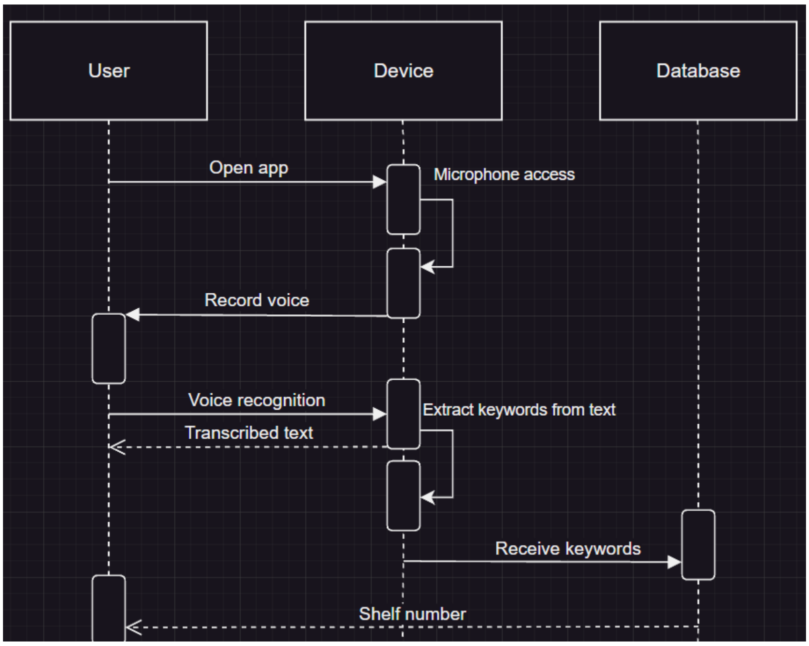


Fig.6. Sequence diagram of voice assistant

Next, the text is split into words, and all the command words that are not included in the search are defined in a separate file and will be removed from the text. Next, you need to take into account the difference in cases in the words in the voice command and in the database. For example, if you ask your assistant for a blue skirt in the accusative case, and the database contains a blue skirt in the nominative case, nothing will be found without taking this into consideration. Therefore, the UkStemmer library was used, which is a tool for processing the Ukrainian language and has the ability to highlight the stem of a word. The stem of a word is its basic form, which represents the essence of the word without endings and other morphological changes. The search algorithm itself is designed in such a way that it will find items for all keywords at once, but if it doesn't find one, it will continue to search by the name of the item. For example, if you wanted a red T-shirt and there is no such thing available, the assistant will offer you other T-shirts. The search results, i.e. all possible shelf numbers, are then returned to the main page. To test the algorithm, we need to create a test dataset with clothes to imitate database. Since we don't care about the authenticity of the products we will use we will use, this dataset can be generated. This dataset will contain such data as product names, color, size, location (shelf number). The items should have a certain dependency in location, for example, the same item of different sizes should be on the same shelf, and items of the same type can be on neighboring shelves. An example of how the assistant works can be seen in Figure 7.

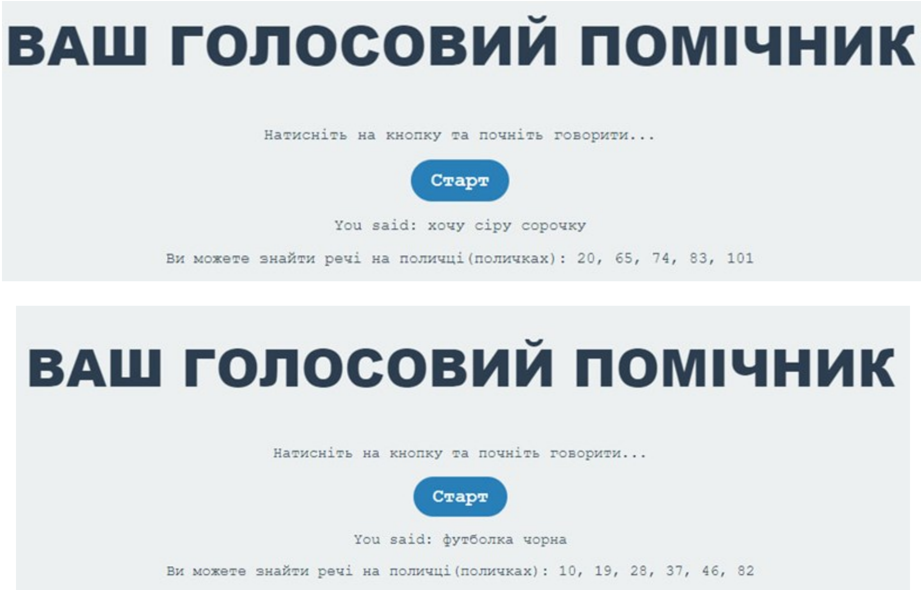


Fig.7. Example of how voice assistant works

As we can see, the speech recognition system demonstrates stable and high accuracy during microphone recording. This is important because it provides users with reliable and accurate recognition results even when using simple microphones. The search also works well, as it is performed on keywords individually, so the order of the words does not matter, which means that the system has flexibility and understanding of speech contexts. Users will find it convenient to use keywords to quickly find specific information.

The interface is designed with ease of use in mind, including for users with no previous experience with technology. The interface has intuitive controls that make it easy to use the program. At the top of the interface is the Start button, which is used to start recording audio. The color of the button changes to red when pressed, indicating to the user that the recording process has started. When the recording is complete, the button returns to blue, indicating that the recording is over. The recorded text is also displayed on the screen, allowing the user to check the correctness of the recognition. This can be useful for identifying possible errors or inaccuracies in the recognized text.

Conclusions

The task of the thesis was set, and the steps to accomplish it. The task was to offer our own solution to the problem and develop a system for its implementation. The methods of audio signal processing were analyzed, the literature was analyzed, and the existing approaches to voice command recognition were compared. The current state of the market and the demand for the technology were researched. The next step was to create a dataset and prepare it for training. The dataset contains 30 different categories and 3095 audio files. The number of records was increased by augmentation. The neural network model was trained and tested. The model's accuracy reached 96.02%. When tested on a separate data set, the WER was 3.98% and the CER was 0.87%. The neural network was integrated with the search system in the API. The speech recognition system showed stable and high accuracy during microphone recording. The search works well and allows you to find results by keywords and names of things. The system's flexibility allows it to understand speech contexts and does not depend on the order of words in a phrase. In general, the results indicate a successful implementation of a voice assistant with high speech recognition accuracy and efficient search.

References

1. Balamurali B. T., Lin K. E., Lui S. et al. Toward robust audio spoofing detection: A detailed comparison of traditional and learned features. IEEE Access. Vol. 7, 2019. P. 84229–84241. DOI:10.1109/ACCESS.2019.2923806.
2. Dentamaro V., Giglio P., Impedovo D. et al. AUCO ResNet: an end-to-end network for Covid-19 pre-screening from cough and breath. Pattern Recognition. Vol. 127, 2022. DOI:10.1016/j.patcog.2022.108656.
3. Esmaeilpour M., Cardinal P., Koerich A. L. A Robust Approach for Securing Audio Classification against Adversarial Attacks. IEEE Transactions on Information Forensics and Security. Vol. 15, 2020. P. 2147–2159. DOI:10.1109/TIFS.2019.2956591.
4. Hasib K. M., Tanzim A., Shin J. et al. BMNet-5: A Novel Approach of Neural Network to Classify the Genre of Bengali Music based on Audio Features. IEEE Access. 2022. P. 1–1. DOI:10.1109/ACCESS.2022.3213818.
5. Huang Y.-P., Mushi R. Classification of Cough Sounds Using Spectrogram Methods and a Parallel-Stream One-Dimensional Deep Convolutional Neural Network. IEEE Access. Vol. 10, 2022. P. 97089–97100. DOI:10.1109/ACCESS.2022.3205591.
6. Kantipudi M. V. V. P., Kumar S. A computationally efficient learning model to classify audio signal attributes. International Journal of Electrical and Computer Engineering. Vol. 12, Issue 5. P. 4926–4934. DOI:10.11591/ijece.v12i5.pp4926-4934.
7. Sheibani R., Nikookar E., Alavi S. An ensemble method for diagnosis of Parkinson’s disease based on voice measurements. Journal of Medical Signals and Sensors. Vol. 9, Issue 4. P. 221–226. DOI:10.4103/jmss.JMSS_57_18.
8. Sterpu G., Harte N. Taris: An online speech recognition framework with sequence to sequence neural networks for both audio-only and audio-visual speech. Computer Speech and Language. Vol. 74, 2022. DOI:10.1016/j.csl.2022.101349.

9. Tsalera E., Papadakis A., Samarakou M. Comparison of pre-trained cnns for audio classification using transfer learning. Journal of Sensor and Actuator Networks. Vol. 10, Issue 4. DOI:10.3390/jsan10040072.
10. Vargas E., Hopgood J. R., Brown K. et al. On Improved Training of CNN for Acoustic Source Localisation. IEEE/ACM Transactions on Audio Speech and Language Processing. Vol. 29, 2021. P. 720–732. DOI:10.1109/TASLP.2021.3049337.
11. Кривохата А. Г. Огляд методів машинного навчання для класифікації акустичних даних / А. Г. Кривохата, О. В. Кудін, А. О. Лісняк // Вісник Херсонського національного технічного університету, Херсон, 2018. 331 с.
12. Palanisamy, Kamalesh; Singhania, Dipika; Yao, Angela. Rethinking CNN models for audio classification. National Institute of Technology, Tiruchirappalli, India. 2020.
13. Nvidia NeMo documentation [Electronic resource]. Retrieved from: https://docs.nvidia.com/deeplearning/nemo/user-guide/docs/en/main/starthere/intro.html

Volodymyr Shymanskyi **Володимир Шиманський**	Ph.D., Associated Professor at the Department of Artificial Intelligence, Lviv Polytechnic National University, Lviv, Ukraine e-mail: volodymyr.m.shymanskyi@lpnu.ua https://orcid.org/0000-0002-7100-3263	Доцент кафедри системи штучного інтелекту Національного університету “Львівська політехніка”
Nikoletta Kazonnikova **Ніколетта Казьоннікова**	Student at the Department of Artificial Intelligence, Lviv Polytechnic National University, Lviv, Ukraine e-mail: nikoletta.kazonnikova.knm.2019@lpnu.ua https://orcid.org/0000-0003-3899-5894	Студентка кафедри системи штучного інтелекту Національного університету “Львівська політехніка”

https://doi.org/10.31891/csit-2023-2-8
UDC 004.93

Khrystyna LIPIANINA-HONCHARENKO
West Ukrainian National University
Sophie SCHAUER, Juergen SIECK
HTW University of Applied Sciences Berlin
Anatoliy SACHENKO, Ivan KIT
West Ukrainian National University

CONCEPT OF INFORMATION SYSTEM FOR CULTURAL HERITAGE SITES RENOVATION USING AUGMENTED REALITY

Cultural heritage is key to identity and development. Many valuable objects are affected by time, natural elements and financial lack. Innovative technologies are crucial for their preservation. The paper develops an information system based on augmented reality (AR) for the restoration of cultural heritage. This extends AURA's approach to preserving musical spaces by applying AR to cultural objects. New techniques improve AURA, allowing accurate restoration of objects affected by time. 3D modeling and machine learning allow to create virtual replicas with precision down to the smallest detail. Augmented reality and machine learning open new perspectives for the preservation of cultural values. The paper proposes an innovative approach of using AR for cultural heritage restoration. The authors offer unique solutions for accurate modeling of 3D models of objects. The purpose of the paper is to develop an information system for the restoration of cultural heritage through AR. This will increase the possibilities of preservation and research of values. Using AR and 3D modeling can improve the restoration of objects and provide access for researchers and the public. In future research, the proposed approaches and methods will be implemented to expand the functionality of the information system. This will include developing interactive interfaces for interacting with virtual models of cultural heritage, analyzing data for a deeper understanding of restoration processes and trend detection, as well as integrating cutting-edge information technologies, such as virtual reality and natural language recognition systems, using artificial intelligence. The primary goal is to improve the processes of preserving and studying cultural heritage through the use of modern information technologies.

Keywords: cultural heritage, restoration, innovative technologies, augmented reality (AR), machine learning, 3D modeling, information technologies.

Христина ЛІП'ЯНІНА-ГОНЧАРЕНКО
Західноукраїнський національний університет
Софі ШАУЕР, Юрген ЗІК
Університет прикладних наук Берліна
Анатолій САЧЕНКО, Іван КІТ
Західноукраїнський національний університет

КОНЦЕПЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ РЕСТАВРАЦІЇ ОБ'ЄКТІВ КУЛЬТУРНОЇ СПАДЩИНИ З ВИКОРИСТАННЯМ ДОПОВНЕНОЇ РЕАЛЬНОСТІ

Культурна спадщина є ключовою для ідентичності та розвитку. Багато цінних об'єктів загрожують час, природні стихії і фінансова відсутність. Інноваційні технології є вирішальними для їх збереження. Стаття розробляє інформаційну систему, засновану на доповненій реальності (AR), для реставрації культурної спадщини. Це розширює підхід AURA, спрямований на збереження музичних просторів, застосовуючи AR для культурних об'єктів. Нові методи покращують AURA, дозволяючи точну реставрацію об'єктів, що зазнають впливу часу. 3D-моделювання та машинне навчання дозволяють створювати віртуальні копії з точністю до деталей. Доповнена реальність та машинне навчання відкривають нові перспективи для збереження культурних цінностей. Стаття пропонує інноваційний підхід використання AR для реставрації культурної спадщини. Автори пропонують унікальні рішення для точного моделювання 3D-моделей об'єктів. Мета статті - розробити інформаційну систему для реставрації культурної спадщини через AR. Це дозволить збільшити можливості збереження та дослідження цінностей. З використанням AR та 3D-моделювання можна покращити реставрацію об'єктів та забезпечити доступ для дослідників і публіки. У подальших дослідженнях запропоновані підходи та методи будуть реалізовані для розширення функціональності інформаційної системи. Це включатиме розробку інтерактивних інтерфейсів для взаємодії з віртуальними моделями культурної спадщини, аналіз даних для глибшого розуміння процесів реставрації та виявлення тенденцій, а також інтеграцію передових інформаційних технологій, таких як віртуальна реальність та системи розпізнавання природної мови, з використанням штучного інтелекту. Основною метою є вдосконалення процесів збереження та вивчення культурної спадщини шляхом використання сучасних інформаційних технологій.

Ключові слова: Культурна спадщина, реставрація, інноваційні технології, доповнена реальність (AR), машинне навчання, 3D-моделювання, інформаційні технології.

Introduction

Cultural heritage is the foundation that unites people with the past and defines their identity. It plays a crucial role in strengthening social cohesion and promoting cultural development. However, many cultural heritage sites gradually deteriorate due to the effects of time, natural disasters, and insufficient funding for their restoration and conservation. In this context, new technologies, such as Augmented Reality (AR) and Machine Learning, become increasingly important, offering innovative solutions for preserving and restoring cultural heritage. Therefore, this article is aimed at developing such an innovative information technology based on AR for the renovation of cultural heritage sites.

The proposed methods in the concept will enhance the achievements of the AURA project, providing impeccable renovation and reproduction of cultural heritage sites that may be lost or destroyed over time. Advanced technologies, such as 3D modeling and Machine Learning, enable the creation of accurate virtual replicas of real objects while preserving their original shape, texture, color, and materials.

Moreover, the application of Augmented Reality in combination with object recognition and Machine Learning algorithms opens up countless new possibilities for the restoration, preservation, and visualization of cultural treasures for present and future generations.

The rest of the paper is structured in the following way. In Section Related work the analysis of recent related works in the domain is performed. In Section Architecture of Information System, the key methods are identified and three fundamental functions of the information system are formed. In Section Case Study the experimental part is described. A Section Conclusions summarizes the received results.

Related work

In this domain, numerous research works have already been conducted, which combine two key themes: the use of Augmented Reality (AR) and Virtual Reality (VR) technologies in the context of architecture and cultural heritage. Several relevant studies analyze the potential of these technologies, including their impact on understanding architectural space [2] and their wide range of applications in the field of cultural heritage [3, 6]. Even specific examples of VR/AR applications that enhance interaction with cultural heritage objects are presented [4, 5].

AR-oriented research includes the development of an AR taxonomy for art and cultural heritage [7], the presentation of an AR-based visualization system for cultural heritage objects [8], the use of photogrammetry and AR for reproducing cultural objects [9], the creation of an AR system to enhance users' knowledge of cultural heritage [10], and methods of multispectral 3D recording and documentation for the development of mobile applications dedicated to cultural heritage [11].

The concept of an information system for the restoration of cultural heritage objects through augmented reality is an extension of the ideas of the AURA project [1], which already uses technologies for the analysis and preservation of musical and cultural spaces. The AURA project focuses on creating acoustic models of music venues and researching their impact on sound, while the new concept uses augmented reality for the reproduction and preservation of significant cultural objects.

This research presents an innovative approach to using augmented reality in the context of cultural heritage restoration compared to existing solutions, including the development of specialized methods and technologies. The authors propose unique solutions that involve precise modeling, identification, and creation of 3D models of cultural monuments.

The purpose of this article is to develop the concept of an information system for the renovation of cultural heritage objects using augmented reality, as discussed below.

Architecture of Information System

The authors have conducted a series of previous research studies on the application of advanced information technologies, including Augmented and Virtual Reality, Machine Learning, and 3D modeling, in various domains [12, 13]. Taking this into account, as well as the expected results of cultural heritage object renovation, the following key methods are identified:

- method of Enhancing 3D Models of Cultural Heritage Objects: This method involves precise and detailed reproduction of the shape, color, texture, and materials of cultural objects in the 3D models;
- method of Recognizing Specific Cultural Heritage Objects based on Machine Learning algorithms, which will recognize and classify objects in the real world;
- method of Generating Specific Cultural Heritage Objects, which involves creating 3D models of virtual replicas of cultural objects with accurate representation of original colors, textures, and other details;
- method of Determining the Correct Placement of Objects, which will utilize orientation sensors and object recognition to determine where the virtual object should be placed in the real world.

The analysis of these methods enables to form the three fundamental functions of the information system: data collection, virtual restoration, and interactivity. Based on the content of these functions, the architecture of the information system for the renovation of cultural heritage objects can be synthesized (Figure 1).

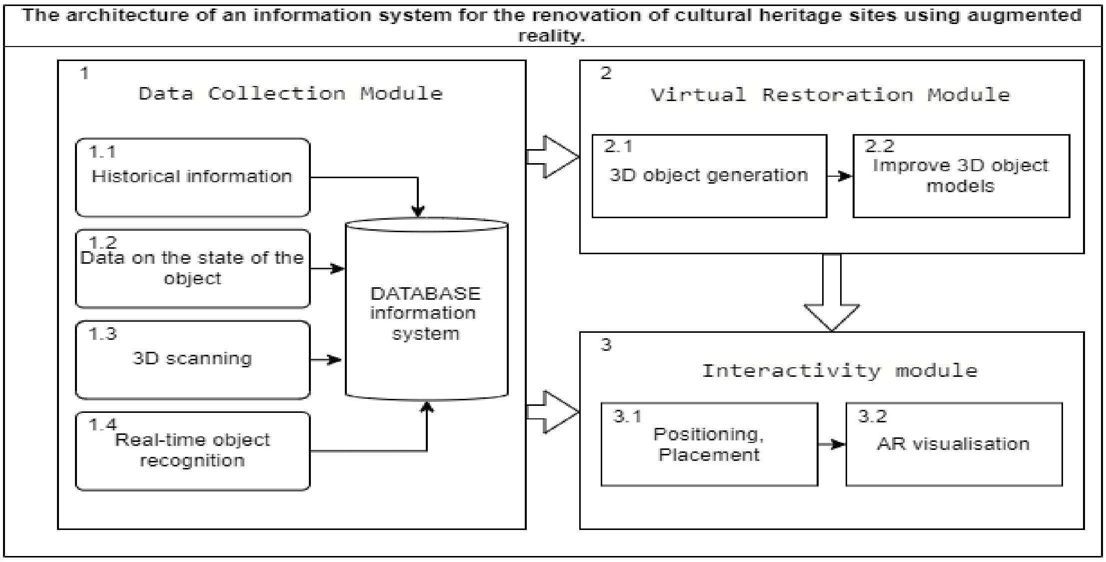


Fig. 1. Architecture of the Information System for Cultural Heritage Objects Renovation

Let's take a closer look at the components of the architecture. The Data Collection Module (Block 1) ensures the gathering of all necessary information about cultural heritage objects and includes the following sub-blocks:

Historical Information (Block 1.1): Collects historical data about the object, including its origin, usage history, significance, and other important details.

Object Condition Data (Block 1.2): Gathers information about the current condition of the object, including any damages, changes, or other issues that may affect the restoration process.

3D Scanning (Block 1.3): Utilizes 3D scanning technologies to create detailed 3D models of the object, which can be used for virtual restoration.

Real-time Object Recognition (Block 1.4): Employs machine learning algorithms for real-time recognition and classification of objects in the real world. All these blocks contribute to the Information System Database.

The Virtual Restoration Module (Block 2) is responsible for the virtual restoration of cultural heritage objects. It includes the following sub-blocks:

Object 3D Generation (Block 2.1): Utilizes 3D modeling technologies to create virtual replicas of cultural heritage objects.

Enhanced 3D Object Models (Block 2.2): Utilizes various techniques and algorithms to enhance 3D models, ensuring precise reproduction of the object's shape, color, texture, and materials.

The Interactivity Module (Block 3) focuses on user interaction with virtual cultural heritage objects and consists of the following sub-blocks:

Positioning and Placement (Block 3.1): Uses orientation sensors and object recognition to determine where the virtual object should be placed in the real world.

AR Visualization (Block 3.2): Employs augmented reality technologies to visualize virtual objects in the real world, allowing users to interact with them in real-time.

This concept has the potential to significantly improve the restoration and preservation processes of cultural heritage objects. Augmented reality can enable a deeper and more effective engagement of researchers, restorers, and even the general public with the process of renovation and preservation of cultural heritage.

These modules play crucial roles in the renovation and restoration of cultural heritage objects. They work together to ensure precise reproduction of objects, effective recognition and classification, and interactive engagement with the objects.

Users of the system could include scholars, restorers, and the general public who can use these technologies to explore, restore, and appreciate cultural heritage objects in new ways.

Compared to existing solutions, this research represents a conceptually novel approach to using augmented reality for cultural heritage renovation, including the development of specific methods and technologies. Unique solutions are proposed, including precise reproduction, recognition, and generation of 3D models of cultural heritage objects.

This will improve the understanding and interaction with cultural heritage objects, ensuring their restoration and preservation for future generations.

Case Study

Case 1: "Virtual Restoration of Cultural Heritage Using 3D Modeling"

The restorer opens the application on their computer and enters their login credentials to access the system. The system recognizes the restorer and provides them with full access to all functionalities.

The restorer selects an object for virtual restoration from the system's database. The system loads the 3D model of the object and all available information about its condition, history, and other important details (Figure 2).

The restorer utilizes the system's tools to model the restoration process, including repairing damages, restoring lost details, etc. Once the restoration is complete, the system saves the enhanced 3D model of the object for future use and analysis.

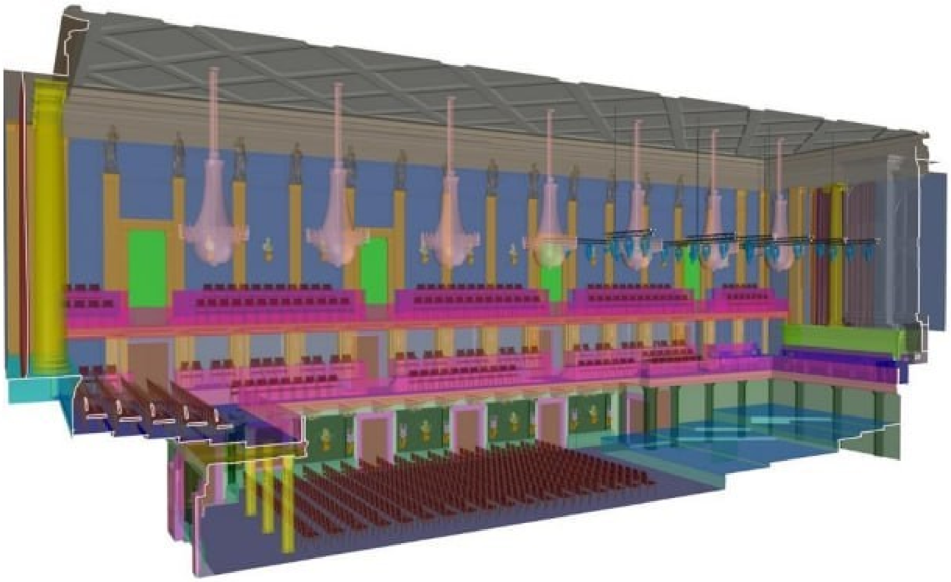


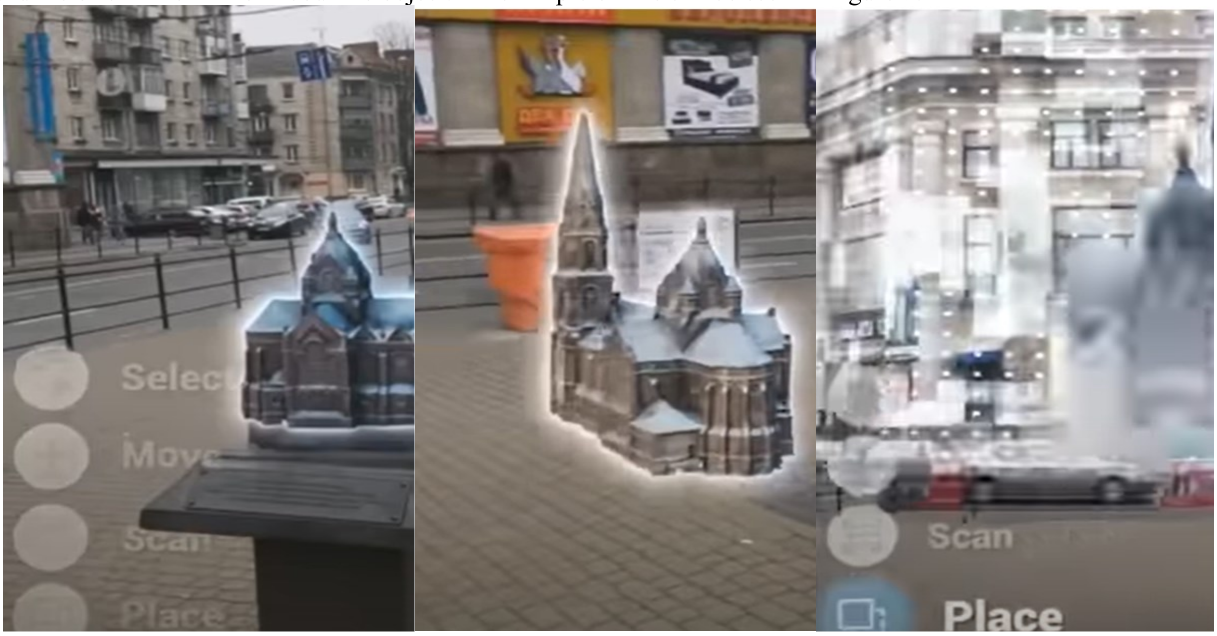
Fig. 2. Example of Case 1 Usage

Case 2: "Augmented Reality for Virtual Tour of Cultural Heritage"

The user opens the application on their mobile device. The app requests permission from the user to access GPS and the camera. The user grants the necessary permissions.

The system determines the user's location and presents a list of nearby cultural heritage objects available for virtual exploration. The user selects one of the objects.

The system loads the 3D model of the chosen object and its historical information. It then displays this model in augmented reality through the mobile device's camera. The user can rotate, zoom in, and zoom out the model and read historical information about the object. An example of this can be seen in Figure 3.



a) b) c)

Fig. 3. Example of Case 2 Usage

Conclusions

The implementation of this information system can open new opportunities for preservation, study, and presentation of cultural heritage. By using advanced technologies such as augmented reality and 3D modeling, the processes of restoration and conservation of cultural heritage objects can be significantly improved, providing researchers, restorers, and the general public with more opportunities for exploration and evaluation.

Using the proposed system, ordinary users have the ability to virtually visit and explore cultural heritage objects, while restorers can plan and model the restoration processes. All of this has the potential to enhance the understanding and preservation of cultural heritage for future generations.

In future research, the proposed approaches and methods will be implemented to expand the functionality of the information system. This will include developing interactive interfaces for interacting with virtual models of cultural heritage, analyzing data for a deeper understanding of restoration processes and trend detection, as well as integrating cutting-edge information technologies, such as virtual reality and natural language recognition systems, using artificial intelligence. The primary goal is to improve the processes of preserving and studying cultural heritage through the use of modern information technologies.

References

1. AURA – Auralisation of acoustic heritage sites using Augmented and Virtual Reality. URL: https://aura-project.eu/
2. Russo M. AR in the Architecture Domain: State of the Art. *Applied Sciences*. 2021. T.11. №15. C.6800. https://doi.org/10.3390/app11156800
3. Bekele M. K., et al. A survey of augmented, virtual, and mixed reality for cultural heritage. *Journal on Computing and Cultural Heritage*. 2018. T.11. №2. C.1–36. https://doi.org/10.1145/3145534.
4. Jung K., Nguyen V. T., Piscarac D., and Yoo S.-C. Meet the virtual Jeju Dol Harubang – The mixed VR/AR application for cultural immersion in Korea's main heritage *ISPRS International Journal of Geo-Information*. 2020. T.9. №6. C.367. https://doi.org/10.3390/ijgi9060367
5. Boboc R. G., Duguleană M., Voinea G.-D., Postelnicu C.-C., Popovici D.-M., and Carrozzino M. Mobile augmented reality for cultural heritage: Following the footsteps of Ovid among different locations in Europe. *Sustainability*. 2019. T.11. №4. C.1167. https://doi.org/10.3390/su11041167
6. Aliprantis J., Caridakis G. A survey of augmented reality applications in cultural heritage. *International Journal of Computational Methods in Heritage Science*. 2019. T.3. №2. C.118-147. https://doi.org/10.4018/ijcmhs.2019070107
7. Čopič Pucihar, K., Kljun, M. ART for Art: Augmented Reality taxonomy for art and cultural heritage. In: Geroimenko V. (eds) *Augmented Reality Art. Springer Series on Cultural Computing*. Springer, Cham. 2018. https://doi.org/10.1007/978-3-319-69932-5_3
8. Han J.-G. et al. Cultural heritage sites visualization system based on outdoor augmented reality. *AASRI Procedia*. 2013. T.4. C.64–71. https://doi.org/10.1016/j.aasri.2013.10.011
9. Portalés C., Lerma J. L., Pérez C. Photogrammetry and augmented reality for cultural heritage applications. *The Photogrammetric Record*. 2009. T.24. №128. C.316-331. https://doi.org/10.1111/j.1477-9730.2009.00549.x
10. Rinaldi A. M., Russo C., Tommasino C. An approach based on linked open data and augmented reality for cultural heritage content-based information retrieval. In: Gervasi O., Murgante B., Hendrix E.M.T., Taniar D., Apduhan B.O. (eds) *Computational Science and Its Applications – ICCSA 2022. ICCSA 2022. Lecture Notes in Computer Science*, vol. 13376. Springer, Cham. 2022. https://doi.org/10.1007/978-3-031-10450-3_8
11. Cabrelles M., Blanco-Pons S., Carrión-Ruiz, B., Lerma, J. L. From multispectral 3D recording and documentation to development of mobile apps for dissemination of cultural heritage. In: Levy T., Jones, I. (eds) *Cyber-Archaeology and Grand Narratives. One World Archaeology*. Springer, Cham. 2018. https://doi.org/10.1007/978-3-319-65693-9_5
12. Lipianina-Honcharenko Kh., Savchyshyn R., Sachenko A., Chaban A., Kit I., Lendiuk T. Concept of the intelligent guide with AR support. *International Journal of Computing*. 2022. T.21. №2. C.271-277. https://doi.org/10.47839/ijc.21.2.2596
13. Schauer S., Sieck J., Lipianina-Honcharenko Kh., Sachenko A. and Kit I. Use of digital auralised 3D models of cultural heritage sites for long-term preservation. *Proceedings of the 2023 IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2023*, 2023. (In print)

Khrystyna Lipianina-Honcharenko **Христина Ліп'яніна-Гончаренко**	PhD, Associate Professor at Department for Information-Computing Systems and Control, West Ukrainian National University, Ternopil, Ukraine e-mail: xrustya.com@gmail.com https://orcid.org/0000-0002-2441-6292, Scopus Author ID: 58070482200, ResearcherID: C-7399-2018	кандидат технічних наук, доцент кафедри інформаційно-обчислювальних систем і управління, Західноукраїнський національний університет, Тернопіль, Україна
Sophie Schauer **Софі Шауер**	HTW University of Applied Sciences Berlin, Germany e-mail: sophie.schauer@htw-berlin.de	Університет прикладних наук Берліна
Juergen Sieck **Юрген Зік**	PhD, Professor at HTW University of Applied Sciences Berlin e-mail: juergen.sieck@htw-berlin.de Scopus Author ID: 24722952700	Університет прикладних наук Берліна
Anatoliy Sachenko **Анатолій Саченко**	DSc, Professor at Department for Information-Computing Systems and Control, West Ukrainian National University, Ternopil, Ukraine e-mail: as@wunu.edu.ua https://orcid.org/0000-0002-0907-3682, Scopus Author ID: 35518445600, ResearcherID: I-4908-2017	доктор технічних наук, професор кафедри інформаційно-обчислювальних систем і управління, Західноукраїнський національний університет, Тернопіль, Україна
Ivan Kit **Іван Кіт**	lecturer at Department for Information-Computing Systems and Control, West Ukrainian National University, Ternopil, Ukraine e-mail: kitivan400@gmail.com https://orcid.org/0000-0002-4526-0020, Scopus Author ID: 57208674228	викладач кафедри інформаційно-обчислювальних систем і управління, Західноукраїнський національний університет, Тернопіль, Україна

https://doi.org/10.31891/csit-2023-2-9
UDC 004.89

Markiyan PYTS, Ivanna DRONYUK
Lviv Polytechnic National University

THEORETICAL BACKGROUND FOR CREATING REAL WORLD DATA LAKE ARCHITECTURE

Data Lakes are the methods for storing and managing large quantities of unstructured data. Modern enterprises and small businesses, regardless of their size, can use this data to derive valuable insights about their business, such as process improvements or product usage. Although this approach to extracting insights is powerful, only some studies describe the actual implementation architectures of data lakes and warehouses.

There are a lot of high-level studies of data lakes, their use cases, and approaches to data analysis. Still, we did not discover a practical guide on how to appropriately unite the available tools to set up a complete data lake capable also of consuming live event stream data, e.g. ingesting data about a user visiting a product website or interacting with some product feature or consuming IoT device event.

The main goal of this article is to describe the architecture using AWS to create a robust, cheap-to-maintain, and scalable solution for Data Lake and Data Warehouse. It can be used by small Software as a Service (SAAS) companies, big enterprises, or individual researchers to build the base of such solutions with a clear guideline of all moving pieces and an understanding of how they are connected.

The article provides a broad overview of setting up a data lake on AWS (Amazon Web Services). It covers setting up an Application Programming Interface (API) to consume data, store data, visualize data, and the ability to create data lakes across multiple AWS accounts quickly with a single Command-line Interface (CLI) command.

This is useful for creating a scalable data lake or data warehouse setup that doesn't require much manual work. We describe how such design can be done using infrastructure as a code approach to achieve this and propose AWS architecture for solving the task of compelling data storage. The article provides a diagram of the proposed architecture accompanied by a high-level description and theoretical background.

Keywords: business intelligence, data science, infrastructure as a code, data warehouse, data lake, Data lake architecture, Amazon Web Services.

Маркіян ПИЦ, Іванна ДРОНЮК
Національний Університет «Львівська Політехніка»

ТЕОРЕТИЧНА ОСНОВА ДЛЯ СТВОРЕННЯ АРХІТЕКТУРИ REAL WORLD DATA LAKE

Data Lake — це методи зберігання та керування великою кількістю неструктурованих даних. Сучасні середні та малі підприємства, незалежно від їх розміру, можуть використовувати ці дані, щоб отримати важливу інформацію про свій бізнес, наприклад, удосконалення процесів або використання продукту.

Існує багато досліджень високого рівня щодо озер даних, їх використання та підходів до аналізу даних. Проте ми не знайшли практичного посібника про те, як належним чином об'єднати наявні інструменти для створення повноцінного озера даних, здатного також споживати дані потоку подій у реальному часі, наприклад, поглинати дані про відвідування користувачем веб-сайту продукту або взаємодію з певною функцією продукту, або споживати події з пристроїв Інтернету речей.

Основна мета цієї статті - описати архітектуру з використанням AWS для створення надійного, дешевого в обслуговуванні та масштабованого рішення для озера даних та сховища даних. Вона може бути використана невеликими компаніями, що надають програмне забезпечення як послугу (SAAS), великими підприємствами або окремими дослідниками для побудови бази таких рішень з чітким описом всіх рухомих частин і розумінням того, як вони пов'язані між собою.

У статті представлено широкий огляд налаштування Data Lake на AWS (веб-сервіси Amazon). Він охоплює налаштування інтерфейсу прикладного програмування (API) для споживання даних, зберігання даних, візуалізації даних і можливості швидкого створення Data Lake для кількох облікових записів AWS за допомогою однієї команди інтерфейсу командного рядка (CLI).

Запропонований підхід корисний для створення масштабованого Data Lake або налаштування сховища даних, яке не вимагає багато ручної роботи. Для досягнення цього використано підхід до інфраструктури як коду. Запропонована AWS архітектура для вирішення задачі ефективності зберігання даних. Стаття демонструє діаграму запропонованої архітектури та її високорівневий опис з теоритичним підгрунтям.

Ключові слова: бізнес-аналітика, наука про дані, інфраструктура як код, сховище даних, Data Lake архітектура, Data Lake, веб-сервіси Amazon.

Introduction

Data lakes help enterprises store and manage large amounts of structured and unstructured data in a cost-effective, secure, and easily accessible way. It is used as a source of data for Business intelligence (BI) [1][10] tools.

Business Intelligence is a crucial process for organizations [12] that involves transforming raw data into meaningful information for decision-making. BI[10] helps companies identify potential opportunities, and threats, and make better decisions.

The setup of a data lake is usually a long and complicated process that requires a lot of expertise in the area and appropriate tools to assemble an initial solution to start ingesting data from various sources. In addition to this, there are several other important considerations, such as security, cost efficiency [1], access to modern data transformation [16], and visualization tools [5]. Ensuring that a data lake's IT infrastructure can be managed easily is also crucial to prevent a single manual configuration error from causing the entire system to break down.

Companies and individuals who want to start implementing a data lake require a clear guide on the topic, to know which services to use, what each service in the data lake is doing, and how they are connected.

There are a lot of high-level studies [2] of data lakes, their use cases, and approaches to data analysis. Still, we did not discover a practical guide on how to appropriately unite the available tools to set up a complete data lake capable also of consuming live event stream data, e.g. ingesting data about a user visiting a product website or interacting with some product feature or consuming IoT [13] device event.

The main goal of this article is to describe the architecture using AWS [6] to create a robust, cheap-to-maintain, and scalable solution for Data Lake and Data Warehouse. It can be used by small Software as a Service (SAAS) companies, big enterprises, or individual researchers to build the base of such solutions with a clear guideline of all moving pieces and an understanding of how they are connected.

Theoretical background for creating architecture

Regarding storing data, there are two popular options: a Data Lake [15] or a Data Warehouse [2]. While both serve the same purpose of storing data, there are some critical differences between them. A data lake is a repository that allows for storing raw, unstructured data in its native format. In contrast, a data warehouse is designed to store structured data that has already been processed and organized.

Data lakes are ideal for storing large volumes of data that may be unstructured or require further processing. This makes them particularly useful for data scientists and analysts who need to analyze data in its raw form. In contrast, data warehouses are ideal for storing structured data ready for analysis. They are well-suited for business intelligence and reporting applications that require fast access to data.

With AWS services [8] available, choosing between a data lake or a warehouse is unnecessary. We can use tools such as AWS Glue Studio or AWS Glue DataBrew to prepare the data for further analysis. Learning Data Science [3,7] basics are also beneficial to get the basic knowledge needed to use data transformation [16] and visualization tools.

Having the difference defined, we should also understand that data from the Data Lake can be post-processed and used in the data warehouse. These approaches usually coexist in modern setups as different stages of data flow.

Data Lake setup needs many different functionalities and tools glued to assemble the working solution. We recommend AWS [8] for data lake infrastructure based on the [11] course and the following factors:

- Many modern SAAS Businesses and companies are using AWS for infrastructure already because AWS is the biggest cloud provider in the world at the time of this article’s creation. So it will be convenient to use familiar services for businesses that use AWS already.
- AWS exists in the cloud. Therefore data that is being stored can be shared with personnel all over the world.
- AWS follows deny access by default principle which helps ensure that only authorized people will access the data.

AWS Lambda is a highly scalable, event-driven computing service that allows developers to run code without provisioning or managing servers. It supports various programming languages, including Node.js, Python, and Java, and can be used to build a wide range of applications, including web applications, mobile backends, and IoT [13] applications. Various events and API Gateway ones can trigger Lambda functions. Lambda is an essential building block for many data lake architectures, as it allows for data processing in real time. Access if this data format is what we expect to consume and use AWS Software Development Kit (SDK) to push data farther to the Kinesis Firehose delivery stream, which will push it into the Data Lake.

AWS Kinesis Firehose is an essential building block for many data lake architectures, as it allows for data processing in real time. Other AWS services, such as data transformation and visualization, can consume this data stream for further processing. It can continuously collect and load data from various sources, such as IoT [13] devices, clickstreams, social media feeds, and logs. Kinesis Firehose takes care of all the underlying infrastructure and scaling and focuses on analyzing data. Kinesis Firehose also provides built-in support for data delivery to Amazon S3, which makes it easy to store and cost-effectively manage large volumes of data. Overall, AWS Kinesis Firehose is a powerful and flexible tool for collecting, processing and delivering real-time data streams to other AWS services.

Amazon Simple Storage Service (S3) is a highly scalable and reliable object storage service offered by AWS. It enables developers and businesses to store and retrieve data anywhere on the web. One of the critical benefits of S3 is its simplicity. S3 provides a simple web services interface for storing and retrieving data from any location on the internet.

The S3 provides several storage classes for different data access patterns and cost requirements. The most commonly used storage classes are Standard, Standard Infrequent Access (Standard-IA), and Glacier. The Standard storage class is suitable for frequently accessed data and provides low latency and high throughput performance. The Standard storage class is suitable for infrequently accessed data but offers low latency and high throughput performance. The Glacier storage class is designed for long-term data archiving and offers the lowest storage costs but higher retrieval times.

Overall, Amazon S3 is a highly scalable, reliable, and secure object storage service that provides simple and cost-effective storage for all data types. With its numerous features and integration with other AWS services, S3 has become an essential tool for many businesses and developers looking to store and manage their data in the cloud. The interesting fact is that even though S3 is file storage, augmented with AWS Athena Capability, It can store data in a database-like format called Parquet. This is a much cheaper storage option than traditional databases, yet Athena can query this data with SQL like a usual database.

AWS Lake Formation [11] is a tool that simplifies the setup and management of data lakes on AWS. It provides various features, such as data ingestion, data cataloging, and access control, which can help organizations get up and running with their data lake quickly and easily. One of the key benefits of AWS Lake Formation is its ease of use. We use AWS Lake Formation to automate everyday tasks, such as data ingestion, cataloging, and access control.

In summary, AWS Lake Formation is a robust and comprehensive service that helps organizations build, secure, and manage data lakes quickly and easily. Its advanced security features, ease of use, and integration with other AWS services make it a popular choice for organizations looking to store and analyze large amounts of data in the cloud.

Features for creating a proposed Architecture

In the proposed architecture, Lake Formation monitors AWS Glue databases and tables that our infrastructure as a code solution will define and deploy, which can be referenced in our Kinesis Firehouse delivery stream described above.

AWS Glue is a fully managed extract, transform, and load (ETL) service offered by Amazon Web Services (AWS). AWS Glue makes it easy to automate the process of discovering, cataloging, and cleaning data, which saves time and resources.

One of the key features of AWS Glue is its ability to automatically discover and catalog data from various sources, such as databases, file systems, and streaming services. AWS Glue crawls data sources, extracts metadata, and creates a centralized metadata repository called the AWS Glue Data Catalog. This makes it easy to search, discover, and query data across multiple sources.

AWS Glue supports various data formats, including CSV, JSON, Parquet, and ORC. It also provides various connectivity options, such as JDBC, ODBC, and AWS services like Amazon S3 and Redshift. This makes it easy to load data from various sources and transform it into the desired format for analysis.

In the proposed architecture, we use the AWS glue database and table (defined in our infrastructure as a code solution) to define the data schema of events which is expected to come into the data lake. The kinesis delivery stream, which consumes events from the outside world, is associated with the appropriate glue table. It validates data that comes with the event against glue table schema and if valid, automatically puts this data into the S3 bucket as a Parquet file which AWS Athena can read.

AWS Athena is a powerful and fully managed serverless query service offered by Amazon Web Services (AWS). It allows querying data in Amazon S3 using standard SQL syntax without the need for complex data processing infrastructure. With AWS Athena, we analyze data stored in Amazon S3 and extract insights quickly and easily.

One of the key benefits of AWS Athena is its serverless architecture. There are no servers to manage or provision capacity. AWS Athena automatically scales queries to match the data's size and complexity, making it ideal for ad-hoc queries and exploratory analysis. AWS Athena provides a simple and intuitive interface for querying data.

Amazon QuickSight is a cloud-based business intelligence [1] service that provides easy-to-use and interactive dashboards and visualizations [5]. It allows users to access and analyze data from various sources, including AWS data services, databases, and third-party applications. With Amazon QuickSight, businesses can gain valuable insights into their data and make informed decisions faster. One of the key benefits of Amazon QuickSight is its ease of use. It provides a simple and intuitive user interface that allows users to create dashboards and visualizations without the need for extensive technical knowledge. Users can quickly and easily connect to their data sources, choose from various visualization options, and create interactive dashboards with drag-and-drop functionality.

Amazon QuickSight supports various data sources, including Amazon S3 via Athena. It provides a range of visualization options, including charts, graphs, and maps, which allows users to create compelling and interactive dashboards. Users can customize their dashboards with branding and layout options, as well as add filters and parameters to enable interactive data exploration. Amazon QuickSight also provides AI-powered features such as auto-narratives that provide contextual explanations to data in the dashboard, anomaly detection, and forecasting.

Infrastructure as code (IaC) [4] is a practice of managing and provisioning technology infrastructure through machine-readable definition files, rather than manually configuring hardware devices and software systems. In IaC, infrastructure is defined using code, which can be version-controlled, tested, and deployed in the same way as software code. The code is written in a declarative format, which describes the desired state of the infrastructure. IaC tools, such as CloudFormation, interpret these definitions and automatically provision and manage the infrastructure accordingly. IaC offers several benefits over traditional infrastructure management approaches.

There are such stages:

- Firstly, it allows for creating of reproducible infrastructure, which can be easily shared and reused across different teams and projects.
- Secondly, it enables the automation of infrastructure provisioning and management, reducing the need for manual intervention and ensuring consistency and reliability.
- Thirdly, it provides a clear audit trail of changes to the infrastructure, which can be tracked and audited easily.

AWS CloudFormation is a powerful and fully managed service to create and manage resources in a repeatable and automated way. One of the key benefits of AWS CloudFormation is its ability to automate infrastructure provisioning and management. By defining the desired state of infrastructure in a template, we can create and manage resources in a repeatable and consistent way. AWS CloudFormation automatically handles the complexity of resource dependencies and orchestrates the deployment of resources in the correct order.

For the proposed architecture, we used AWS CloudFormation to provide all infrastructure required for our data lake. Without it, it would be very hard to maintain all the infrastructure pieces manually in the AWS management console. It would be much more complicated if we needed to deploy similar data lakes in different AWS accounts.

Another significant benefit is that the CloudFormation code can be stored on Git code version control [14]. This provides the advantage of knowing how our infrastructure looked in the past and how it looks now and grants the ability to modify and augment it in the future. Also, we can roll back to the latest working copy if some wrong modification got introduced by mistake.

CloudFormation is an excellent tool on its own, but it's rarely used as is. There are popular tools and frameworks which allow working with it at a higher level, increasing the speed of implementation. One such tool is Serverless Framework [9]. With CloudFormation, we could define infrastructure in YAML format once and use Serverless Framework CLI to deploy or destroy the infrastructure if it's not needed anymore.

The Serverless Framework [9] is an open-source framework that enables developers to build, deploy, and manage serverless applications easily. It abstracts the underlying infrastructure details and allows developers to focus on writing code, reducing the time and complexity required to develop and deploy serverless applications.

The Proposed Architecture Overview

In this section, we present the proposed architecture of a data lake. Fig. 1 represents the architecture that is powering the solution. It fits both Data Lakes and Data Warehouses.

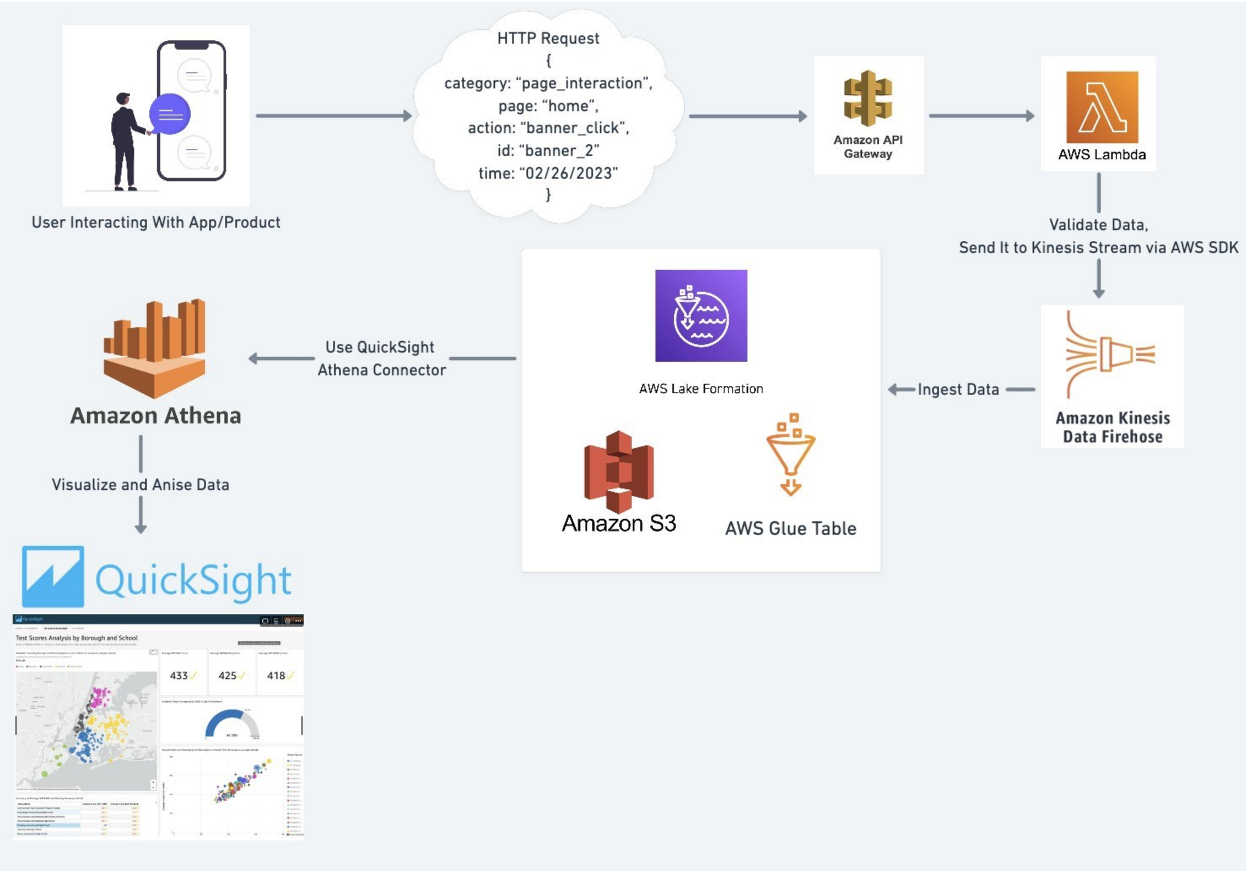


Fig. 1. Data Lake Architecture

Fig. 1 describes a situation when we want more governance over the data to be ingested and stored in the system, which is closer to the Data Warehouse scenario, though can also be used as Data Lake later, since data is in S3 while being originally formatted to serve the user interactions analytics purposes (e.g. tracking user clicks on product webpage), can be post-processed later using many data processing tools. AWS provides to actually transform this data and use it in different sorts of analysis and use cases.

Conclusion

We proposed to create an AWS Architecture that allows storing large amounts of data in the data lake with the optional ability to filter what data format is being stored in the data lake and what data is discarded by the system, to achieve this, we have AWS lambda function exposed to the outside world via AWS API Gateway that acts like a filtering gateway between the data lake and events data that is being ingested into it from external systems and apps. This way, the solution is flexible and can be used to create a Data Warehouse if this approach is preferred.

The lambda function sends data to the Kinesis Firehose delivery stream, which can handle close to the infinite scale of data being sent to it, so we don't have to worry about the system being overwhelmed by external events' quantity and frequency. Valid data is being ingested into the S3 bucket, which is associated with the AWS Glue database table.

S3 storage which is used in the proposed architecture, is much cheaper than regular databases, it does not limit the amount of data that can be stored. This way, we can be sure that as more data comes into the data lake, we will not run out of storage space.

With Infrastructure As A Code proposed in the architecture, we can easily maintain the infrastructure of the data lake, and extend it with new tables, data storage types, and schemas. Also, the requirement is to move the infrastructure into a different AWS account and patch or remove some of its pieces. In that case, it is safe and easy since CloudFormation remembers all the parts of the infrastructure and the connections between them for us. This is especially useful if we have an AWS account that hosts more things than our data lake, which is common among organizations.

Architecture is helpful for companies and individuals who want to set up their own BI data stores for many purposes like analysis of SAAS products usage, monitoring of IoT devices, etc. Data which is stored in the data lakes is also helpful for the creation of machine learning models.

In the future, this architecture can be used as a base of more specialized data warehousing solutions targeting specific business use cases, e.g, e-commerce, IoT devices, including medical ones, etc. It can be easily adjusted for this purpose we only require to change/add schema on AWS Glue Table and adjust filtering logic for events allowed for ingestion in the entry lambda function.

References

1. Muriithi, G. M. and J. E. Kotzé. (2013) "A conceptual framework for delivering cost-effective business intelligence solutions as a service," in Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference, p.96-100.
2. Marilex Rea Llave. (2018) "Data lakes in business intelligence: reporting from the trenches" Procedia Computer Science 138:p. 516-524.
3. Davenport, T. H. and D. J. Patil. (2012) "Data scientist: the sexiest job of the 21st century." Harvard Business Review 90 (10): p.70-79.
4. Dalla S., Palma D., Di Nucci D., Tamburri A. (2020) "AnsibleMetrics: A Python library for measuring Infrastructure-as-Code blueprints in Ansible" SoftwareX VOLUME 12, 100633
5. George L. C., Guo Ya., Stepanov D., Kumar V., Peri R., Elvitigala R. L., Spichkova M. (2020) "Usage visualization for the AWS services". Procedia Computer Science, Vol. 176, p. 3710-3717
6. Sudhanshu M., Lakhera G., Srivastava A. K., M. Kumar (2021). "Cost analysis of amazon web services – From an eye of architect and developer". Materials Today: Proceedings 46(20), p. 10757-10760
7. Eremenko K. "Data Science A-Z™: Real-Life Data Science Exercises Included by "Udemy course URL: https://www.udemy.com/course/datascience
8. Maarek S. "Ultimate AWS Certified Developer Associate" Udemy course URL: https://www.udemy.com/course/aws-certified-developer-associate-dva-c01
9. Weinberger A. "Serverless Framework Bootcamp: Node.js, AWS & Microservices" Udemy course URL: https://www.udemy.com/course/serverless-framework
10. Oyku I., Jones M. C., and Sidorova A. (2013) "Business intelligence (BI) success and the role of BI capabilities." Decision Support Systems 56 (1), p.361-370.
11. Owoyemi Y. "Build a Secure Data Lake in AWS using AWS Lake Formation" Udemy course URL: https://www.udemy.com/course/build-a-secure-data-lake-in-aws-using-aws-lake-formation
12. Davenport, T. H., and Dyché J.(2013) "Big data in big companies. "International Institute for Analytics 3. URL: https://www.iqpc.com/media/7863/11710.pdf
13. Kumar S., Tiwari P., Zymbler M. (2019) "Internet of Things is a revolutionary approach for future technology enhancement: a review" Journal of Big Data 6, 111
14. César J., Ríos C., Embury S. M., Eraslan S. (2022) "A unifying framework for the systematic analysis of Git workflows" Information and Software Technology 145: 106811
15. Larson D. and Chang V. (2016) "A review and future direction of agile, business intelligence, analytics and data science." International Journal of Information Management 36 (5): p.700-710.
16. Stein B. and Morrison A.. (2014) "The enterprise data lake: Better integration and deeper analytics." PwC Technology Forecast: Rethinking integration 1 p.1-9.

Markiyan Pyts **Маркіян Пиц**	Ph.D. student at Artificial Intelligence Department, Lviv Polytechnic National University https://orcid.org/0009-0005-1814-1586 e-mail: markiyan.pyts@gmail.com	аспірант кафедри систем штучного інтелекту Національний університет «Львівська політехніка»
Ivanna Dronyuk **Іванна Дронюк**	Ph.D., Associate Professor of Artificial Intelligence Department, Lviv Polytechnic National University https://orcid.org/ 0000-0003-1667-2584 e-mail: Ivanna.m.droniuk@lpnu.ua	доцент кафедри систем штучного інтелекту, Національний університет «Львівська політехніка»

https://doi.org/10.31891/csit-2023-2-10
UDC 004.42

Volodymyr TYKHOKHOD, Anton PASICHNIUK
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"

MODELING AND IMPLEMENTATION OF DOMAIN EVENTS IN THE DOMAIN-DRIVEN ARCHITECTURE ON THE .NET CORE PLATFORM

A software architecture centered on a domain model can provide significant advantages over other types of architectures in the long-term development and maintenance of systems with complex domain logic. At the same time, domain-driven design and approaches to software implementation of systems are relatively new concepts and continue to develop in application to various platforms, technologies and programming languages, and are of considerable interest to designers and developers. The presented work examines existing approaches to modeling and implementing domain events on the .NET Core platform in domain-driven architecture, which is one of the newest patterns. There are two approaches to implementing domain event behavior: immediate and delayed event propagation. These two approaches are analyzed and their features are described in detail. The implementation of instant propagation of domain events within a command execution transaction in the CQRS architecture is described. This implementation allows you to get rid of external dependencies, achieve purity of domain entities, as it eliminates the need to inject services, repositories in entities, and also prevents memory leaks and is safe for multithreaded use. Schematically depicts the abstract process of an external command entering a domain model, which causes a change in the state of an aggregate and the propagation of side effects with domain events. This process takes into account the capabilities of the Entity Framework object-relational mapping framework to retrieve context objects that have been changed during process. The entire stack of objects involved in this activity is located in the shared process memory, and the interaction occurs in synchronous mode. For the conceptual detection of events and aggregates, the event storming technique is used, the features of which are discussed in the article.

Keywords: domain-driven design, DDD, design of complex domain areas, , events of the domain area.

Володимир ТИХОХОД, Антон ПАСІЧНЮК
НТУУ «Київський політехнічний інститут імені Ігоря Сікорського»

МОДЕЛЮВАННЯ ТА РЕАЛІЗАЦІЯ ПРЕДМЕТНИХ ПОДІЙ В ПРЕДМЕТНО-ОРІЄНТОВАНІЙ АРХІТЕКТУРІ НА ПЛАТФОРМІ .NET CORE

Архітектура програмного забезпечення, в центрі якої знаходиться модель предметної області, здатна принести значні переваги в довгостроковій розробці та підтримці систем зі складною логікою предметної області в порівнянні з іншими типами архітектур. В той же час предметно-орієнтоване проектування та підходи до програмної реалізації систем є відносно новими концепціями та продовжують розвиток в застосунку до різних платформ, технологій та мов програмування, та становлять значний інтерес у проектувальників та розробників. В представленій роботі розглянуто існуючі підходи до моделювання та реалізації предметних подій на платформі .NET Core в предметно-орієнтованій архітектурі, що є одним з найновіших шаблонів. Існує два підходи до реалізації поведінки предметних подій: миттєве та відкладене розповсюдження подій. Ці два підходи проаналізовано та детально описано їх особливості. Описана реалізація миттєвого розповсюдження подій предметної області в межах транзакції виконання команди в архітектурі CQRS. Дана реалізація дозволяє позбавитись зовнішніх залежностей, досягти чистоти сутностей домену, оскільки позбавляє необхідності у впровадженні сервісів, репозиторіїв у сутності, а також дозволяє запобігти витокам пам'яті та безпечна при багатопотоковому режимі використання. Схематично зображено абстрактний процес надходження зовнішньої команди в модель предметної області, яка спричиняє зміну стану агрегату та розповсюдження побічних ефектів з подіями домену. Цей процес враховує можливості каркасу об'єктно-реляційного відображення Entity Framework для отримання об'єктів контексту, що були змінені в процесі роботи. Весь стек об'єктів, що задіяні в цій діяльності, розташовані в пам'яті спільного процесу, а взаємодія відбувається в синхронному режимі. Для концептуального виявлення подій та агрегатів використовують техніку штурм подій, особливості якої розглянуті в роботі.

Ключові слова: предметно-орієнтоване проектування, DDD, проектування складних проблемних областей, події предметної області.

Introduction

Domain-driven design (DDD) refers to the field of software engineering used to build software systems that implement complex domain logic. DDD focuses on building a system architecture, the central link of which is a domain model (a pattern classified by Martin Fowler [1]). The term domain-driven design was proposed by Eric Evans [2], who described the methodological foundations of DDD and practical techniques for implementing these concepts in the Java programming language. Later, the theoretical and practical aspects of DDD were developed in the works of Vaughn Vernon [3], Martin Fowler [1], Scott Millet [4], Jimi Nielsen [5] and other authors.

The importance of the methodology is evidenced by the field of application of domain-driven design, in particular, in the development of land resource management systems [6], maritime navigation systems [7], delivery organization systems using unmanned aerial vehicles [8].

The methodology of domain-driven design involves the use of various templates of strategic and tactical levels. To form system components with Low Coupling и High Cohesion, domain area decomposition is used using the bounded context template at the strategic level of design, as well as the aggregate template at the tactical level of design.

Events are used to communicate side effects between aggregates and to communicate information about system state changes between bounded contexts. They allow to ensure interaction between system components, without direct interconnection.

Modeling using the behavior of the event system has its own characteristics and requires the use of new approaches. The DDD concepts underlying event modeling are evolving, and events have specific implementation features on different platforms and programming languages. This is especially true for the .NET Core platform, as it is a young technology for developing cross-platform applications that is actively developing.

Therefore, the purpose of this article is to analyze modern approaches to event modeling in domain-driven design on the .NET Core platform.

Features of the architectural style

Domain-driven design is an architectural style designed to create a software model that most accurately reflects a model of a subject area (domain), including business processes and the rules that operate in it. DDD includes strategic and tactical levels of design. At each of the levels, strategic and tactical design templates are used, respectively. The goal of strategic planning is to decompose the problem area into the most conceptually isolated areas in order to curb complexity and eliminate contradictions. These isolated areas are called bounded contexts.

The goal of tactical design is to build a domain model within individual contexts. Isolation of bounded contexts is preferably achieved by implementing a separate microservice for each bounded context. However, there is usually a need to communicate information about side effects between bounded contexts.

As a result of tactical design, a set of aggregates is obtained within limited contexts. Each aggregate has its own internal state and can perform certain actions with this state. Aggregates ensure separation of domain objects from the outside world and data consistency within their borders

When an aggregate changes its state, it can generate domain events that notify other components of the system about the change, other aggregates can respond to these events and change their own state according to the received changes. This behavior, when changes in one aggregate affect the behavior of another aggregate, is also called side effects.

Simulation of events using the technique "Event storming"

Events occupy an important place in modeling the behavior of domain objects. The activity of the subject area is modeled as a sequence of events. Using subject events allows you to build software that contains components that adhere to the principle of single responsibility as much as possible.

For the conceptual identification of events and aggregates, the Event Storming technique is used, which consists in the collective discussion of concepts in order to find the events of the subject area and the internal processes that occur at the same time, drawing clear boundaries between aggregates and determining dependencies between aggregates. The storm of events can be carried out using a board with stickers, or online tools, for example, fig. 1 displays a fragment with the results of the storming of events from a board created by the Miro online system [9].

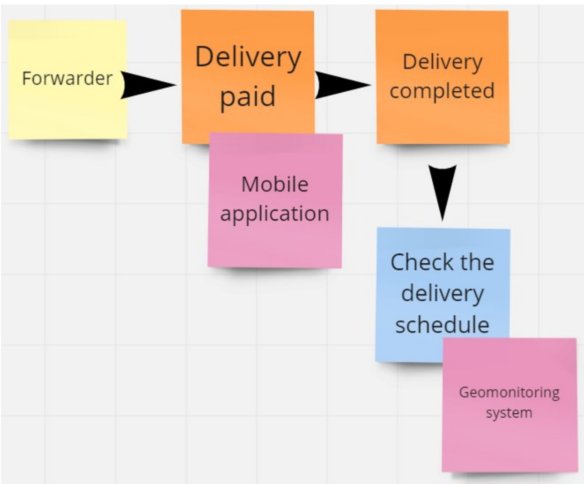


Fig. 1. A fragment of the board with the results of the Event Storming

The color of the cards on the board has a certain semantics, in figure 1 the color of the cards has the following purpose:

- The yellow color represents the object that causes the event to be generated.
- Orange color — a card with this color represents an event.
- Pink represents the system component generating the event or the external system to which the side effect is propagated.

- The blue color of the card reflects the action performed as a result of the use of side effects.

In DDD, the formation and observance of a common language of the domain area in the program code plays an important role. All conceptual and programmatic elements must be consistent and use terms and words from the dictionary of a common language. Therefore, the "Delivery Paid" and "Delivery Completed" events shown in Figure 2 can be named, for example, "DeliveryPaid" and "DeliveryCompleted" in the domain model. In nouns, verbs are used in the past tense, because the event represents a certain fact that happened.

Implementation of domain model events

Events in DDD are divided into two conceptual types: domain events and integration events, their distribution is carried out in synchronous and asynchronous modes, respectively.

Domain events are modeled in the form of simple program objects included in the domain model, which is a simple C# class that contains state but no behavior. An event represents some fact that happened in the past, so it is advisable to prohibit the change of the event object after its creation. Therefore, the following requirements are possible for the event class

1. Read-only properties with event information.
2. A public constructor with the arguments required to initialize the event instance.
3. No behavior, i.e. no methods in the event class.

Their main purpose is to spread side effects between aggregates — when certain changes in the state of one aggregate (that is, when a certain fact has occurred) affect another aggregate. At the same time, aggregates can be located in a common bounded context or divided between different contexts

The event template of the domain area is conceptually separated into a separate template, but its implementation is based on another design template Publish-subscribe [10], which separates two sides of the process — the sender (publisher), which sends messages about important facts of its state change, and subscribers (subscriber), which subscribes to messages from the sender and responds to them.

There are two approaches to the implementation of the behavior of distribution and processing of domain events:

1. Immediate distribution of events.
2. Delayed event propagation.

The first approach is to propagate an event immediately after the domain state changes, resulting in the new state being immediately committed to persistent storage, then an integration event can be published to propagate the state and achieve consistency between different microservices, bounded contexts, or external systems.

The second approach is to store events in memory objects and propagate them during persistent storage.

Implementation of immediate distribution of events

Currently, the following approaches to the implementation of domain events are used:

- A classic implementation with static methods of class [11].
- Using the MediatR Nuget package [12], which implements the MediatR pattern [10] and can be used to build an infrastructure for event distribution and management.

Implementation based on a class with static methods [11] allows you to get rid of external dependencies, to achieve purity of domain entities, as it eliminates the need to implement services and repositories in the entity. This implementation prevents memory leaks and is safe for multi-threaded use. Figure 2 shows a UML class diagram that represents the abstract infrastructure of domain events.

The DomainEvent class is responsible for maintaining the domain event infrastructure, it is templated and closed by the class that models the specific event. The class interface provides Register and Raise methods. The Register method allows you to register an event handler as a delegate, with a reference to the delegate added to the private actions collection. The Raise method performs event pushing — in a loop, the handler delegates contained in the actions collection are enumerated and called.

The Register method returns an object that implements the IDisposable interface, so a client of the DomainEvent class can safely implement a mechanism for freeing unmanaged resources. In this scheme, the DomainEvent class creates an instance of the DomainEventRegistrationRemover class, passing a delegate to the constructor that removes the event handler from the actions collection. This delegate is called in the Dispose method of the DomainEventRegistrationRemover class. This mechanism avoids a memory leak where a reference to an unnecessary event handler would block the release of memory by the garbage collector.

The ThreadStatic property next to the private variable actions is thread-safe, indicating to the .NET runtime that each executing thread will have access to a separate instance of the collection. An element with a ThreadStatic attribute must be static - this is a limitation of the .NET framework.

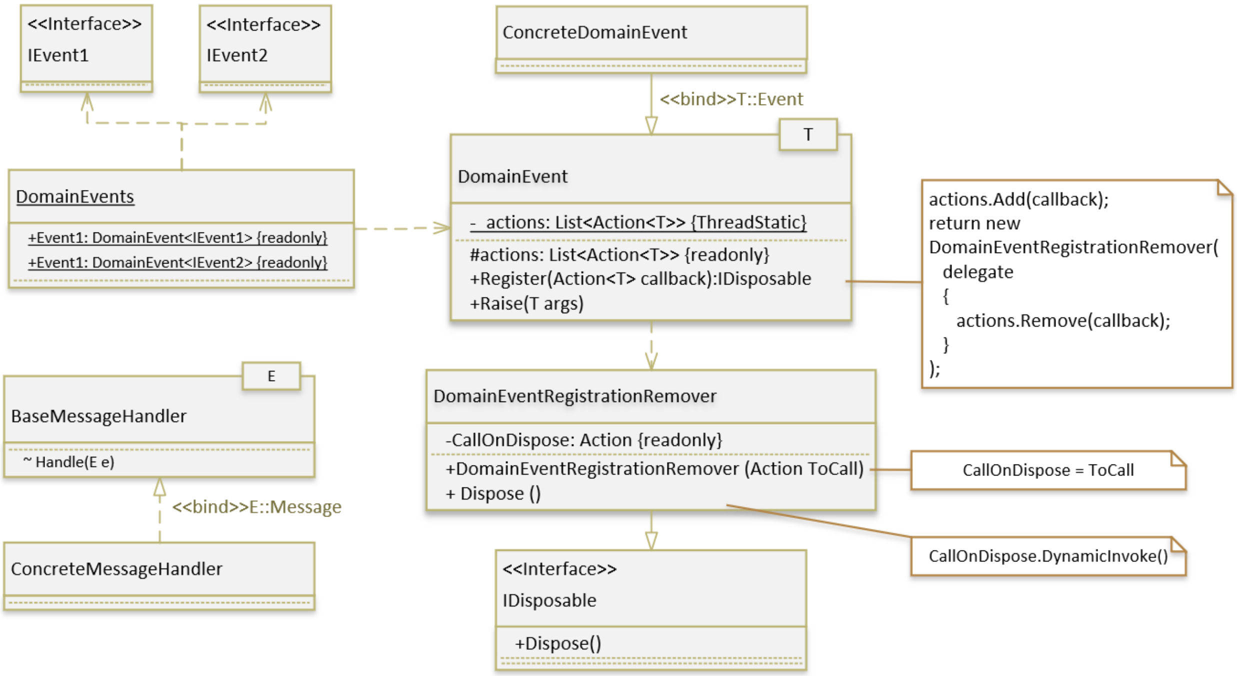


Fig. 2. Abstract infrastructure of domain events

In fig. 3 presents a sequence diagram of some abstract process of propagating domain events during the execution of a web request in the CQRS architecture.

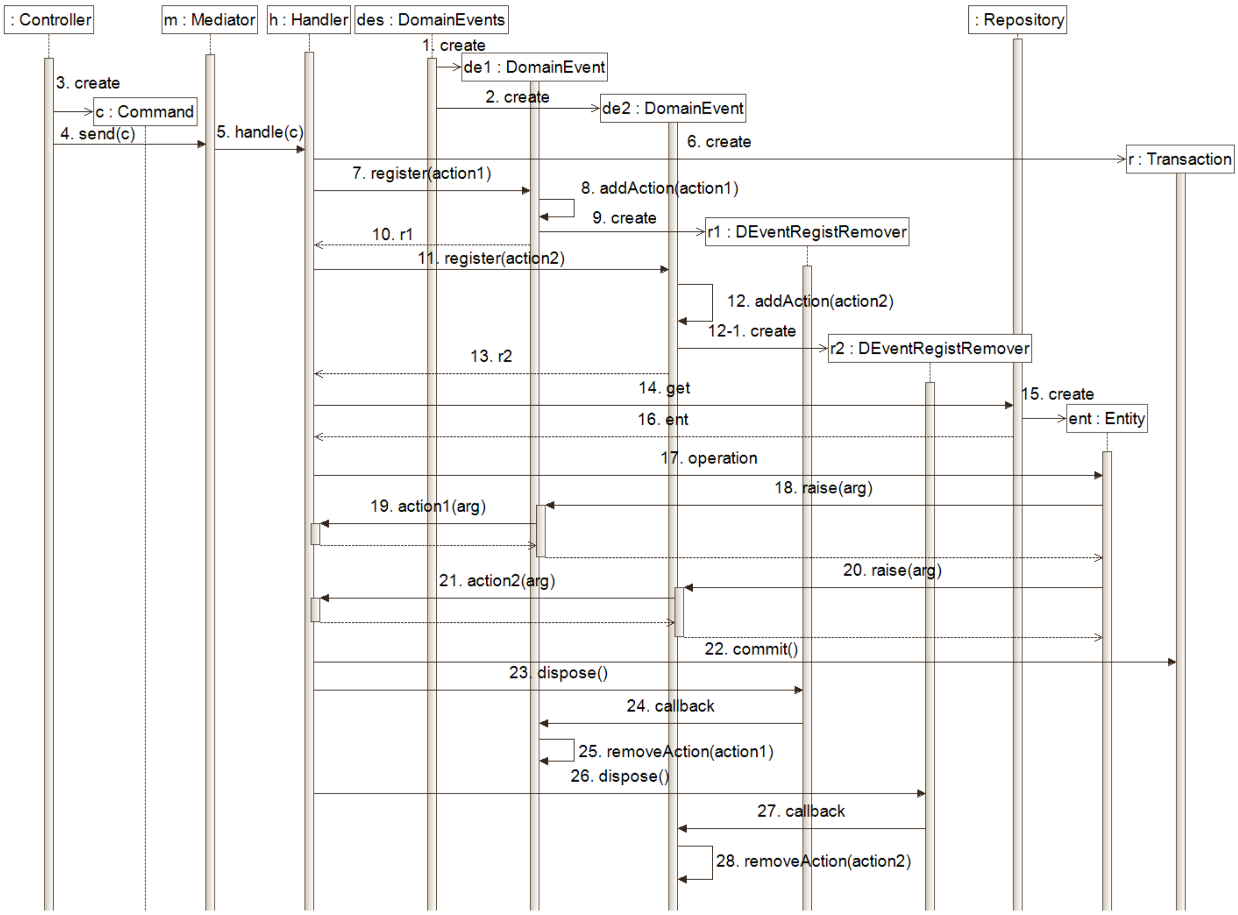


Fig. 3. Sequence diagram of the immediate propagation of domain events within a command execution transaction in the CQRS architecture

After receiving a request, a command object is created in the controller (3), which is passed to the mediator (4), the mediator finds the executor of the command and passes it to them for processing (5). In the event handler, a transaction is created (6), the object event handlers are registered (7, 11), during which handlers are added to the internal collection of DomainEvent objects (8, 12) and objects responsible for cleaning resources are created (9 , 12-1). In step 14, the domain entity is requested from the repository based on the command data, and control is passed to the operation method of this object. Next, the entity, after performing certain actions, generates two events, calling the Raise methods (18, 20) of the event objects created in steps 1, 2. They then call delegates (19, 21) responsible for processing subject events, these delegates are contained in the Handler object. Event handlers can change the states of other aggregates or pass integration events to external systems. After the work of all handlers, the transaction is completed (22) and resources are cleaned (23, 26), during which events are unsubscribed, that is, references to delegates (25, 28) are removed from the internal collections of objects of type DomainEvent

Delayed launch and dispatch of domain events

In [14], the implementation of delayed generation and sending of events is proposed. The approach of delayed launch and dispatch of domain events [16] can use the features of the Entity Framework object-relational mapping framework to obtain context objects that have been changed during operation. Figure 3 schematically depicts the abstract process of the arrival of an external command into the domain model, which causes a change in the state of the aggregate and the propagation of side effects by domain events

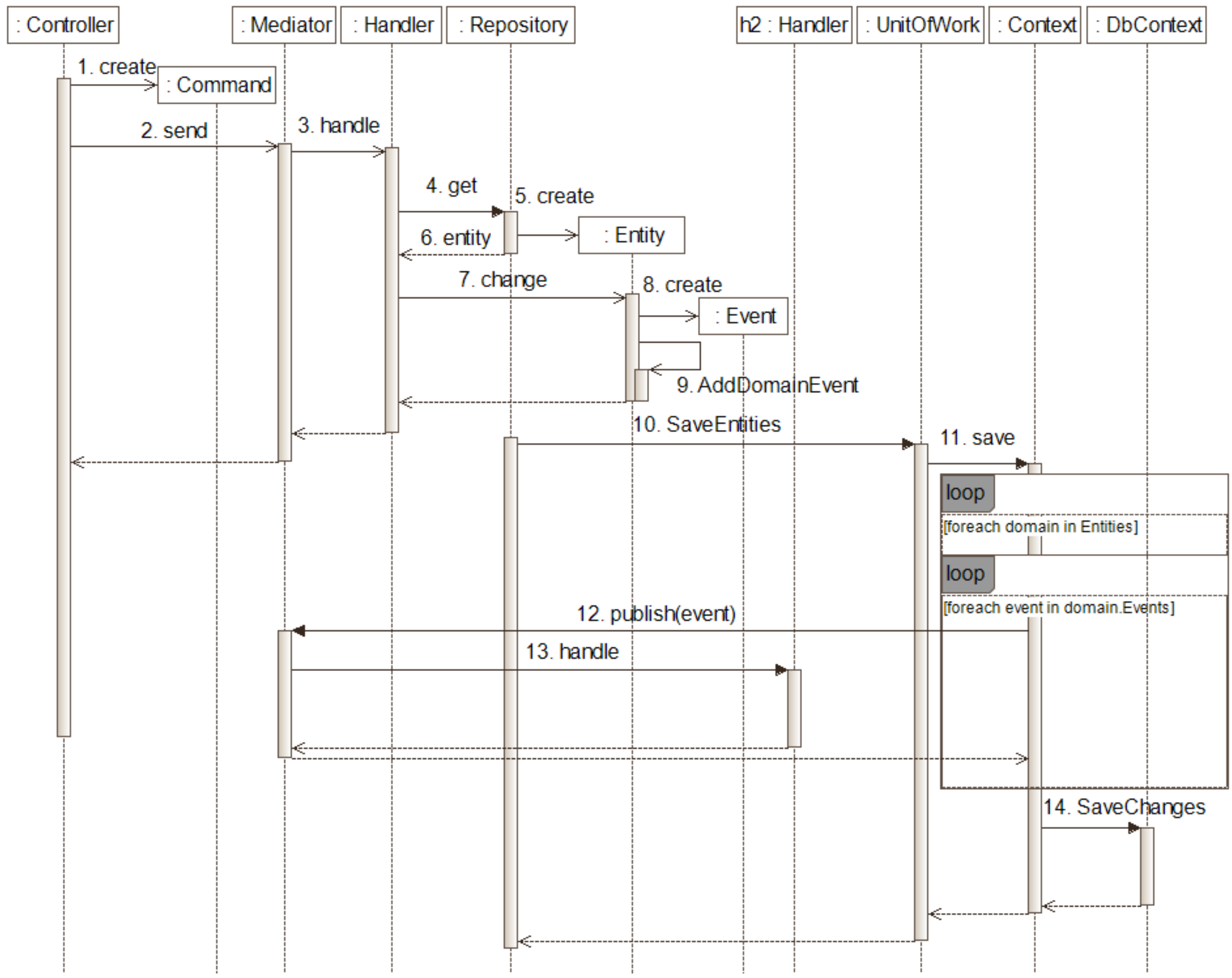


Fig. 3. The process of delayed application of side effects

The key points for understanding this process are the following: after changing the entity (Entity) in step 7, an event is generated in step 8, which is added to the collection of events of the aggregate (step 9), then when saving the aggregate in step 11, the objects are iterated of the context that were changed, the event objects published by the mediator (step 12) and processed by handlers (step 13) are selected from them. Handlers perform updates on context objects, thus propagating side effects. In the final step 14 of the process, the entire entity graph is stored in the repository (DbContext represents the base repository class and is provided by the Entity Framework .NET Core).

The entire stack of objects involved in this activity is located in the shared process memory, and the interaction occurs in synchronous mode.

Conclusions

The paper examines the main concepts of domain area events in domain-driven architecture. Models of immediate and delayed event distribution and features of their software implementation on the .NET Core platform were analyzed.

References

1. Fowler M. Patterns of Enterprise Application Architecture. — Addison-Wesley Professional. — 2003. — 560 p.
2. Evans E. Domain-Driven Design: Tackling Complexity in the Heart of Software. — Addison-Wesley Professional.— 2004. — 560 p.
3. Vernon V. Implementing Domain-Driven Design. — Addison-Wesley Professional. — 2013. — 656 p.
4. Millett, S., & Lippert, J. (2009). Patterns, Principles and Practices of Domain-Driven Design
5. Jimmy Nilsson. Applying Domain-Driven Design And Patterns: With Examples in C# and .net 1st Edition. (2006) Addison-Wesley Professional
6. Domain-Driven Design applied to land administration system development: Lessons from the Netherlands P Oukes, M Van Andel, E Folm
7. Jinsong Zhang, Yan Chen, Shengjun Qin. The Application of Domain-Driven Design in NMS. Proceedings of SPIE - The International Society for Optical Engineering (2011). DOI:10.1117/12.920133
8. Design microservices for drones [Electronic resource]. — Access Mode: https://learn.microsoft.com/en-us/azure/architecture/microservices/design — (Viewed 22.05.2023). — Title from the screen.
9. The Visual Collaboration Platform for Every Team | Miro [Electronic resource]. — Access Mode: https://miro.com/ — (Viewed 22.05.2023). — Title from the screen.
10. Gamma, E., Helm, R., Johnson, R., Vlissides, J. (1994). Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley.
11. Domain Events – Take 2 [Electronic resource]. — Access Mode: https://udidahan.com/2008/08/25/domain-events-take-2/ — (Viewed 22.05.2023). — Title from the screen.
12. Immediate Domain Event Salvation with MediatR [Electronic resource]. — Access Mode: https://ardalis.com/immediate-domain-event-salvation-with-mediatr/ (Viewed 22.05.2023). — Title from the screen.
13. jbogard/MediatR: Simple, unambitious mediator implementation in .NET [Electronic resource]. — Access Mode: https://github.com/jbogard/MediatR — (Viewed 22.05.2023). — Title from the screen.
14. A better domain events pattern [Electronic resource]. — Access Mode: https://lostechies.com/jimmybogard/2014/05/13/a-better-domain-events-pattern/ — (Viewed 22.05.2023). — Title from the screen.

Volodymyr Tykhokhod **Володимир Тихоход**	Ph.D, Senior Lecturer of Department of Automation of Design of Energy Processes and Systems, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute” e-mail: tykhokhod@i.ua https://orcid.org/0000-0002-1525-4770	кандидат технічних наук, старший викладач кафедри автоматизації проектування енергетичних процесів і систем, Національний технічний університет України «Київський політехнічний інститут»
Anton Pasichniuk **Антон Пасічнюк**	Student, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute” e-mail: pasichniuk@ukr.net	студент, Національний технічний університет України «Київський політехнічний інститут»

https://doi.org/10.31891/csit-2023-2-11
UDC 004.942:669.18

Kyrylo KRASNIKOV, Tanzilya FATYKHOVA
Dniprovksyi state technical university

COMPUTER ANALYSIS OF INFLUENCE OF MELT BLOWING MODES ON LADLE LINING MECHANICAL EROSION

The steel ladle lining protects the metal ladle body from overheating and the liquid steel from solidification. One of the reasons for the lining thinning is its erosion by fast flows of liquid steel near the walls and bottom of the ladle (for example, near the blow tuyere). Metallurgists periodically replace the lining, looking for places of the greatest wear, which consumes time and material resources. Predicting the lining wear process opens the way to optimizing and saving these resources.

In the process of blowing, turbulent melt flows erode the ladle lining and its thickness de-creases due to mechanical erosion. The thickness of the steel ladle lining gradually becomes thinner with each casting. The degree of erosion depends on the tangential melt speed. In steel production, they try to slow down this wear, because each lining repair costs considerable time and resources. Scientists paid attention to this problem in publications, in particular, on mathematical modeling of lining wear. A large number of conditions of this process are subject to research, in particular, the number and location of blowing tuyeres, as well as blowing power. Firstly, it is necessary to quickly mix the impurity in the melt, and secondly, to preserve the lining of the ladle. Computer visualization and analysis of this process involves its course and results in the form of calculated fields, in particular, wear. The result fields are stored in a database. Also they are added and processed through a specially designed website. It allows researchers to register and fill in the experiment form, as well as add literature sources of data. The list of literature is used in almost all experiments to compare results. Simulation of the process at blowout rates of 40, 60 and 90 l/min and the number of blowout plugs (tuyeres) from one to three showed that the greatest scouring is predicted at the bottom, near the blowout plugs, and the transition to each higher blowout rate increases the scouring intensity by about 15%. Turning off the tuyeres after 1 minute of blowing significantly reduces erosion by at least 35%. If we consider the ladle wall, without disconnecting the tuyere, the flow rate of 90 l/min is the most destructive.

Keywords: software, computational fluid dynamics, mechanical erosion of lining, teeming ladle.

Кирило КРАСНІКОВ, Танзіля ФАТИХОВА
Дніпровський державний технічний університет, Кам'янське, Україна

КОМП'ЮТЕРНИЙ АНАЛІЗ ВПЛИВУ РЕЖИМІВ ПРОДУВКИ РОЗПЛАВУ НА МЕХАНІЧНУ ЕРОЗІЮ ФУТЕРІВКИ КОВША

Під час транспортування з одного цеху в інший температура розплаву у ковші зберігається завдяки шару вогнетривкої футерівки всередині на стінках і дні. У процесі продування турбулентні потоки розплаву розмивають футерівку і товщина її шару зменшується через механічну ерозію. Особливістю ерозії є неоднорідність розмиття, викликану анізотропією поля швидкості. Дослідження різних конфігурацій фурм і витрат газу дозволить визначити оптимальні технологічні умови для збереження вартісної футерівки. Метою роботи є математичне моделювання механічного зносу футерівки ковша потоками розплаву під час продування аргоном за допомогою розробленого програмного забезпечення наукових досліджень. Ступінь розмиття залежить від швидкості дотичного розплаву. При виробництві сталі намагаються уповільнити це зношування, бо кожний ремонт футеровки вартує значного часу і ресурсів. Цій проблемі науковці приділяли увагу у публікаціях, зокрема, по математичному моделюванню зносу футеровки. Дослідженню підлягає велика кількість умов цього процесу, зокрема, кількість і розташування фурм продування, а також потужність продування. По-перше треба і домішку швидко перемішати у розплаві, а по-друге і футеровку ковша зберегти. Комп'ютерна візуалізація і аналіз означеного процесу передбачають його хід і результати у вигляді розрахованих полів, зокрема, зносу. Поля зберігаються у базі даних і додавання та їх обробка здійснюється через спеціально розроблений вебсайт. Він надає можливість дослідникам зареєструватися і заповнювати форму досліду, а також додавати літературні джерела даних. Перелік літератури використовується практично у всіх дослідах для порівняння результатів. Моделювання процесу в умовах потужності продування 40, 60 і 90 л/хв та кількістю пробок продування від однієї до трьох показало: найбільше розмиття прогнозується на дні, поблизу фурм продування, причому перехід на кожну вищу потужність продування підвищує інтенсивність розмиття приблизно на 15%. Вимикання фурм після 1 хвилин продування значно зменшує розмиття мінімум на 35%. Якщо ж розглядати стінку ковша, то без відключення фурм витрати 90 л/хв є найбільш руйнівними.

Ключові слова: програмне забезпечення, комп'ютерне моделювання, модифікації розплаву в ковші, газове перемішування.

Introduction

The steel ladle lining protects the metal ladle body from overheating and the liquid steel from solidification. One of the reasons for the lining thinning is its erosion by fast flows of liquid steel near the walls and bottom of the ladle (for example, near the blow tuyere). Metallurgists periodically replace the lining, looking for places of the greatest wear, which consumes time and material resources. Predicting the lining wear process opens the way to optimizing and saving these resources.

Related works

The article [1] is devoted to the study of two ladle designs (standard and with a reinforced striking part in the center of the bottom of the ladle). The figure of the experimental setup and the results of comparison of the two cases are given. The comparison showed a significant improvement in the resistance of the bucket lining to erosion wear.

In [2], the results of physical and mathematical modeling of the process of blowing the melt with one tuyere near the ladle wall are compared. The physical model includes oil and water to replace slag and metal melts, respectively. The comparison showed sufficient adequacy of the mathematical model, which takes into account the free surface. In papers [3-4], the authors investigate the process of destruction of a typical ladle lining. They present its physical properties and conduct numerical experiments to find out the best technological conditions in terms of the depth of lining destruction.

In [5], the finite volume method is used for numerical simulation of lining destruction. Using the model, the authors obtained the places of the deepest erosion – the middle of the bucket height. The bucket has a cylindrical geometry. The numerical study [6] is devoted to two cases: the lining is a homogeneous medium with average physical properties; the lining is a porous medium with liquid slag.

The aim of the work is to mathematically model the mechanical wear of the ladle lining by melt streams during argon purging using the developed research software.

Method

Simplifying the problem of mathematical modeling, it is accepted:

1) The melt has the geometry of a cylinder (Fig. 1). The melt depth is a constant value.
2) Wave motion of the melt surface is neglected and it is assumed to be flat.
3) The melt has all the features of a Newtonian viscous fluid with gas content.
4) The melt density is assumed to be constant.
5) Boussinesq's assumption of gas accelerating the vertical component of velocity is used.
6) Slag layer is not taken into account.
7) Lining erosion has a rate that depends linearly on the melt velocity in the vicinity.

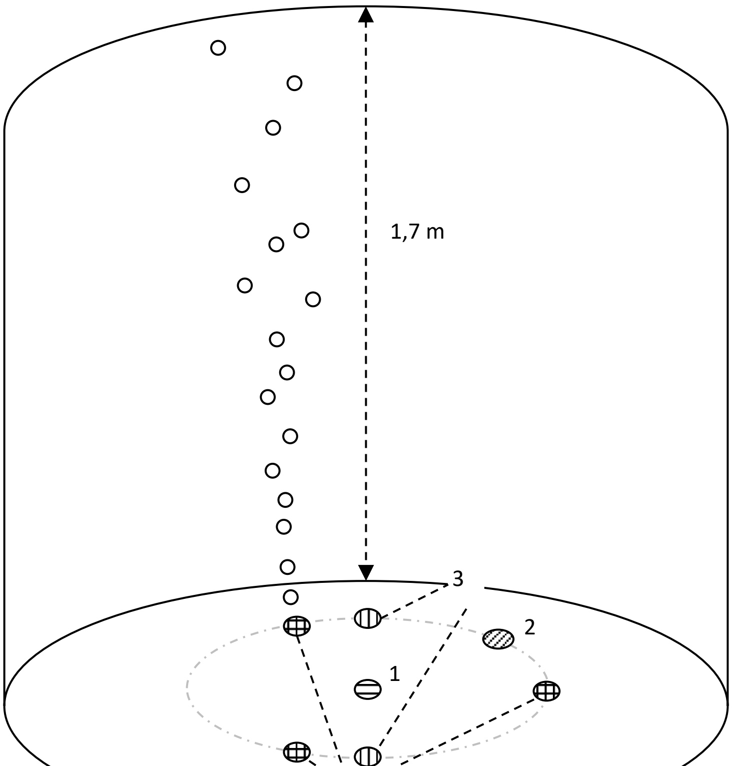


Fig.1. A geometry of the body melt (1/20 scale) with placements of tuyeres

Fluid dynamics is determined by the law of conservation of momentum. Continuity of fluid and gas motion corresponds to the law of conservation of mass:

$$\frac{\partial \vec{\mathbf{v}}}{\partial t} + (\vec{\mathbf{v}} \cdot \nabla)\vec{\mathbf{v}} - D_{\mathrm{v}}\nabla^2 \vec{\mathbf{v}} = -\nabla\left(\frac{p}{\rho_0}\right) - \alpha \vec{\mathbf{g}}, \tag{1}$$

$$\nabla \cdot \vec{\mathbf{v}} = 0, \tag{2}$$

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot \left[\alpha\left(\vec{\mathbf{v}} + \vec{\mathbf{v}}_f\right)\right] - D_\alpha \nabla^2 \alpha = S_\alpha, \tag{3}$$

$$S_\alpha = \frac{q}{V}\frac{T_{air}}{T_m}, \tag{4}$$

where D_v, D_a – effective coefficients of kinematic viscosity and gas diffusion respectively; α – fraction of argon in the melt; g – free fall acceleration (assumed to be 9.81 m/s2); p – pressure field (found due to the condition of solenoidal velocity field); q – gas consumption; T_{air} and T_m – initial gas temperature and melt temperature (1800 C is assumed).

The field of residual damage to the lining is determined by the melt rate along the surface of the walls and bottom of the bucket:

$$\frac{\partial s}{\partial t} = k_s\left|\vec{v}^{||}\right|, \tag{5}$$

where s – is the depth of lining erosion; k_s – coefficient, which adjusts the adequacy of the model to the real erosion process.

The velocity equation is supplemented by boundary conditions that correspond to its components – perpendicular and parallel to the surface w of the solid:

$$\left.\frac{\partial \vec{\mathbf{v}}^{||}}{\partial n}\right|_{w} = 0, \tag{6}$$

$$\left.\vec{\mathbf{v}}^{\perp}\right|_{w} = 0, \tag{7}$$

where n – normal to the wall.

The gas transport equation is supplemented by impermeability boundary conditions on solid surfaces. A constant gas release rate is set on the upper surface of the melt.

The finite volume method and the Euler method are used. The problem of the explicit solution scheme is a significant limitation of the maximum value of the time step due to the presence of the Laplacian operator in the equations. This limitation can be avoided by using an implicit scheme.

The peculiarity of the model of the subject area of numerical study of bucket lining erosion, as well as other scientific studies, is the availability of a list of literature sources, which is used to review modern solutions to the problem, obtain reference data and verify the results.

It is proposed to implement the subject area model using a database based on MS SQL Server and a website based on MS ASP.NET MVC. These technologies are well combined with the MS Visual Studio programming environment, in which we will program in the popular C# language.

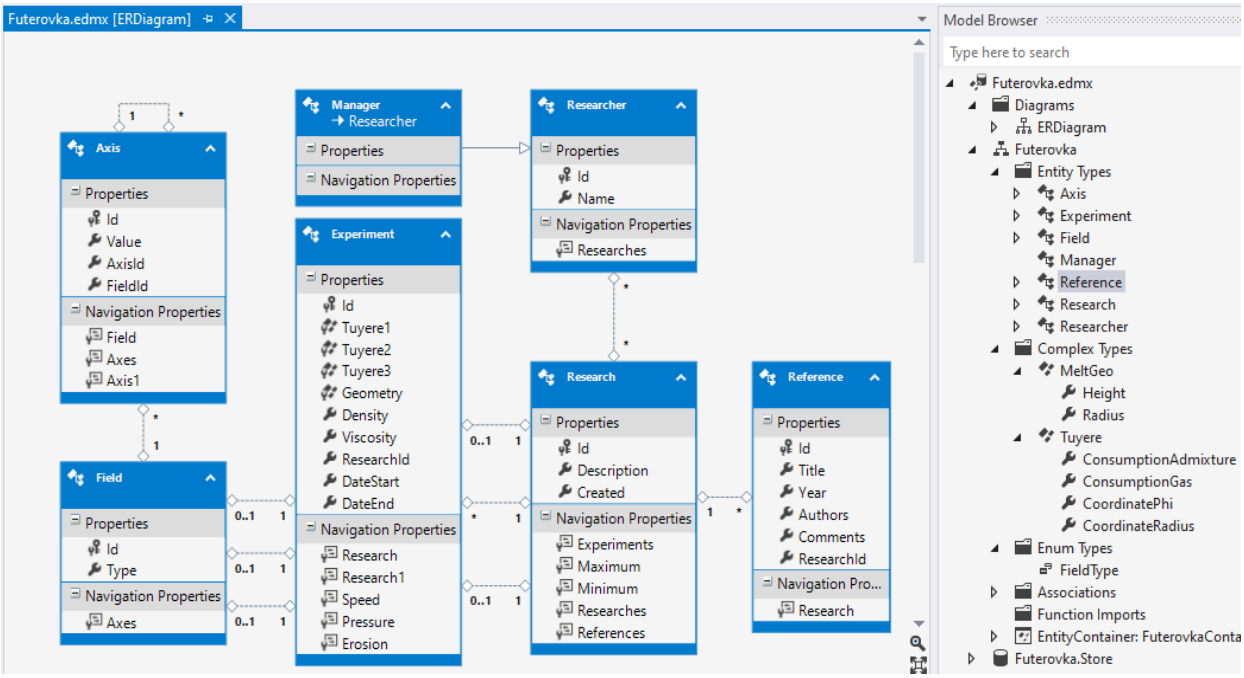


Fig. 2. Entity–Relationship diagram for the domain model of ladle erosion research

Experiments

To solve equations above the finite volume method was used having second order of derivative approximation and conservation property. Cylindrical coordinate system was used as the best choice for the ladle geometry. Time axis is divided on equal intervals and unknown values on the new timestep are calculated using previous ones. To conduct numerical experiments, we use the technological parameters and conditions given in the table 1 and 2.

Table 1

Melt parameters for lining erosion experiment

Parameter	Value
Radius	0,956 m (22 cells)
Height	1,7 m (18 cells)
Angular coordinate	2π (36 cells)
Density	7000 kg/m^3
Temperature	1800 K

Table 2

Parameters of blowing lances for the numerical experiment

Parameter	Value
Total gas consumption	40/60/90 l/min
Diameter of lances	0,1 m
1st setting	1 axial tuyere
2nd setting	1 tuyere at half radius
3rd setting	2 opposite tuyeres at half radius
4th setting	3 tuyeres at an angle of 120° between them at half radius
Erosion coefficient	0,1

Fig. 3 shows charts of the erosion maximum depth depending on time. As seen, increasing gas blowing rate intensifies lining erosion almost linearly. Lines start not at zero time, because swirls reach lining in few seconds after experiment beginning. Turning off the blowing after the first minute significantly slows down further lining erosion.

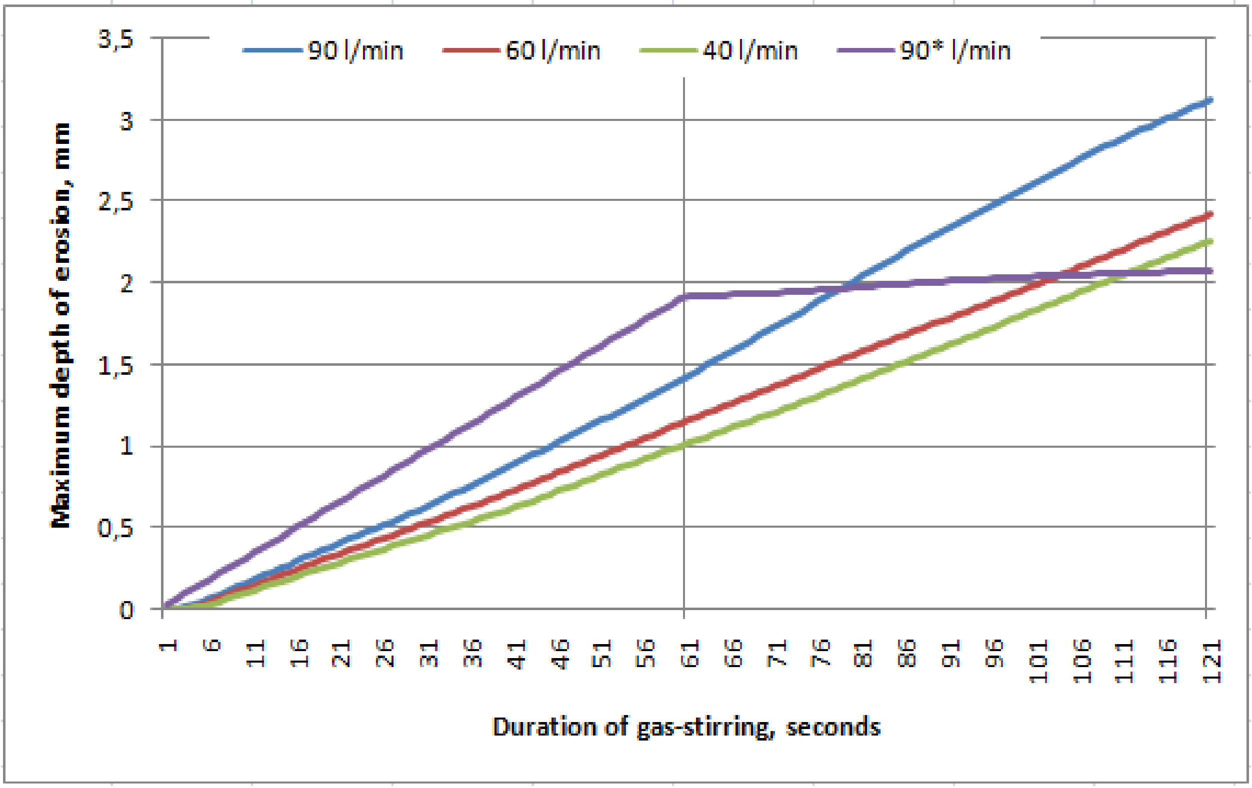


Fig. 3. Erosion in configuration with three tuyeres (the asterisk means turning blowing off at the 60 second)

Results of mathematical modeling can be seen at the website https://www.tensorion.com/lab/metallurgy/ladle/erosion.

Conclusions

The simulation of the process at blowout rates of 40, 60 and 90 l/min and the number of blowout plugs from one to three showed that the greatest erosion is predicted at the bottom, near the blowout tubes, and the acceleration of blowout intensifies erosion by about 15%. Turning off the tuyere after 1 minute of blowing significantly reduces the erosion by at least 35%. If we consider the bucket wall, without disconnecting the tuyere, the flow rate of 90 l/min is the most destructive.

References

1. Molchanov L. S., Lantukh O. S., Synehin Y. V. Physical Modeling of Erosion Destruction of Teeming Ladle Lining in the Process of Its Exploitation. *Bulletin of Vinnitsa Polytechnic Institute*. 2019. № 1. 65–71. DOI: 10.31649/1997-9266-2019-142-1-65-71.

2. Morales R. D., Hurtado F. A., Chattopadhyay K. Physical and Mathematical Modeling of Flow Structures of Liquid Steel in Ladle Stirring Operations. *Metallurgical and Materials Transactions B*. 2020, № 51 (2). 628-648. DOI: 10.1007/s11663-019-01759-x

3. Pandhari A. Modeling of Thermal Stress Cycling in Refractory Materials. Thesis for the degree of PhD, The university of British Columbia, The Faculty of Graduate and Postdoctoral Studies, Vancouver. 2017.

4. Salmash I., Smirnov A. Investigation of stress state and strength of refractory lining steel ladles. *Steel and casting of Ukraine* . 2010. № 9. 48-54.

5. Schmitt N., Berthaud Y., Hernandez J. F., Meunier P., Poirier J. Damage of monolithic refractory linings in steel ladles during drying. *British Ceramic Transactions*. 2004. № 1. 1–20. DOI: 10.1179/096797804225012873.

6. Schmitt N., Hild F., Blond E. Thermal stresses in the working lining of a ladle during the steel refining process. *British Ceramic Transactions*. 2004. № 6. 1–12

Kyrylo Krasnikov **Кирило Красніков**	PhD, Associate Professor of Systems Software Department, Dniprovskyi State Technical University, Kamianske, Ukraine, e-mail: kir_kras@ukr.net orcid.org/0000-0002-4241-0572, Scopus Author ID 57205319971, ResearcherID: Q-8835-2017	кандидат технічних наук, доцент кафедри програмного забезпечення систем, Дніпровський державний технічний університет, Кам’янське, Україна
Tanzilya Fatykhova **Танзіля Фатихова**	Master student of Systems Software Department, Dniprovskyi State Technical University, Kamianske, Ukraine, e-mail: skazochkagrimm@gmail.com	магістрант кафедри програмного забезпечення систем, Дніпровський державний технічний університет, Кам’янське, Україна

https://doi.org/10.31891/csit-2023-2-12
UDC 004.89

Dmytro OKRUSHKO, Antonina KASHTALIAN
Khmelnytskyi National University

SYSTEM OF DISTRIBUTION AND EVALUATION OF TASKS IN THE SOFTWARE DEVELOPMENT PROCESS

The paper is devoted to improving the allocation and evaluation of tasks in software development. Applied aspects of the development of a task allocation and evaluation system are considered in the process of developing software for further analysis, which ensures the most accurate determination of the person who should perform the task and the corresponding task classification tags based on its description. The proposed system provides accurate and fast identification of a person and a group of tags based on the task description. The main goal of the work is to provide an overview of the current state of the art in this field, the advantages and disadvantages of existing approaches, and to propose improvements to the solution.

Challenges related to task allocation and estimation in software development include the need for accurate task estimation, the difficulty of ensuring quality control, and the need for effective communication between developers. To this end, an analysis of the current state of task allocation and estimation was conducted, and a variety of tools and methods available for task allocation and estimation were reviewed, including task tracking systems, project management software, and automated testing tools. Also covered are the various methods used to evaluate tasks, such as peer review, code review, and automated testing.

The future of task allocation and estimation in software development is explored, including the potential for further automation and the need for improved communication between developers, as well as the potential for using artificial intelligence to improve task allocation and estimation. Methods used to measure the efficiency of task allocation and evaluation are also discussed, such as time tracking, percentage of tasks completed, and percentage of defects. The paper proposes AI-based approaches such as natural language processing, machine learning, and deep learning.

Keywords: task planning, task distribution, task classification, frequency characteristics, search for the most optimal criterion for determining the best candidate.

Дмитро ОКРУШКО, Антоніна КАШТАЛЬЯН
Хмельницький національний університет

СИСТЕМА РОЗПОДІЛУ ТА ОЦІНЮВАННЯ ЗАВДАНЬ У ПРОЦЕСІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Стаття присвячена удосконаленню розподілу та оцінювання завдань при розробці програмного забезпечення. Розглянуто прикладні аспекти розробки системи розподілу та оцінювання завдань у процесі розробки програмного забезпечення для подальшого аналізу, яка забезпечує максимально точне визначення особи, яка має виконувати завдання, та відповідних класифікаційних тегів завдання на основі його опису. Запропонована система забезпечує точну та швидку ідентифікацію людини та групи тегів на основі опису завдання. Основна мета роботи - надати огляд поточного стану справ у цій галузі, переваги та недоліки існуючих підходів, а також запропонувати вдосконалення рішення.

Проблеми, пов'язані з розподілом та оцінкою завдань при розробці програмного забезпечення, включають необхідність точної оцінки завдань, складність забезпечення контролю якості та необхідність ефективної комунікації між розробниками. З цією метою було проведено аналіз поточного стану розподілу та оцінки завдань, а також розглянуто різноманітні інструменти та методи, доступні для розподілу та оцінки завдань, включаючи системи відстеження завдань, програмне забезпечення для управління проектами та інструменти автоматизованого тестування. Також були розглянуті різні методи, що використовуються для оцінки завдань, такі як експертна оцінка, перегляд коду та автоматизоване тестування.

Досліджується майбутнє розподілу та оцінки завдань у розробці програмного забезпечення, включаючи потенціал для подальшої автоматизації та необхідність покращення комунікації між розробниками, а також потенціал використання штучного інтелекту для покращення розподілу та оцінки завдань. Також обговорюються методи, що використовуються для вимірювання ефективності розподілу та оцінки завдань, такі як відстеження часу, відсоток виконаних завдань і відсоток дефектів. У статті пропонуються підходи на основі штучного інтелекту, такі як обробка природної мови, машинне навчання та глибоке навчання.

Ключові слова: планування завдань, розподіл завдань, класифікація завдань, частотні характеристики, пошук найоптимальнішого критерію для визначення найкращого кандидата.

Introduction. Problem Setting

Today, developing software for various computer systems is a complex process that requires careful planning, adherence to the plan, and execution. To ensure successful completion of all these tasks, it is very important to have an effective method, a system that will be responsible for distributing these tasks among the performers and evaluating these tasks. After researching some works [1] which presented many tools, information and methods for task distribution and evaluation in the process of software development and computer systems, it was found that the described methods and algorithms have their peculiarities, which do not allow to automate the system of task distribution and classification, so these approaches should be optimized and improved as much as possible, since automation of these processes will affect the coordination in the team, will give the opportunity to determine the exact deadlines for the performance of certain tasks. During the failure of certain nodes of the system, the process of forecasting, distribution and classification of tasks and all elements of the system that were not completed should be transferred to the endpoints or nodes for re-execution. The above-described process greatly affects the speed of the entire system [2] as a whole and is a bad tone for the end user, although the mechanisms considered often have this concept as a basis, it is possible to improve the mechanisms and methods in different ways.

The purpose of the research. Development of a full-fledged autonomous automated information system for determining the best candidate for task execution, as well as task classification and adding to it pre-defined tags or tag groups.

Object of research. The process of creating an information system with ratings of the best candidates based on their practical skills assessments and classification of the task description using data used in previous learning iterations.

Subject of research. Method and information system for determining the best candidate for task execution and model or method for task classification according to tags.

Research methods. To determine the best candidate for task execution, there should be a method that provides for the creation of a rating system based on ratings given by other users, taking into account the load of a person with existing tasks, as well as taking into account the classification of the task for the presence of tags that a person gives preference to and wants to work with. For determining the classification, there will be a method that provides for the search of a group of tags that belong to a particular task and updating the task structure with new weights for classification tags.

Scientific novelty of results. Creation and modernization of algorithms for creating a new system that will work faster, more accurately and will use fewer machine resources.

Practical significance of the obtained results. Creation of a system with some number of independent APIs that will use cloud technologies and a database to store results, track the accuracy of forecasts and provide fast access to resources and their protection.

1. Analysis of existing tools for task distribution and classification.

Waterfall is a model used during project development. It was one of the first models used in the process of creating various software. The Waterfall model provides developers and managers with a step-by-step plan of work, where before moving on to the next step, all previous ones must be completed, as they are blocking for the next steps. This is very similar to the visualization in Canban, where managers and developer teams can monitor the process of existing and completed tasks with the possibility of monitoring in real time which tasks block the next stage and which tasks are more prioritized. Waterfall is implemented as a model of a sequential life cycle. Waterfall starts with the collection of the team and documentation of all necessary requirements. Then the solution design is carried out, which will solve the set tasks. Then it is all tested, covered with tests and given to the customer. But it is important to note that developers must complete the step before they start a new one.

The Waterfall model has six phases, which are shown in Figure 1. Where each phase was developed in such a way that it was clear when you can start the next step. During the first phase, the team only receives all the requirements from the client, makes basic documentation. They must be thoroughly studied taking into account clearly defined final goals.

On the second phase, the software design team creates all the necessary basic architecture. After determining all the necessary basic structure and necessary software, developers begin the main phase, that is, writing code.

After the basic structure is created, the necessary services are determined and access to third-party services is obtained, developers begin the main phase of implementation of the code, where each module is developed separately, which is then necessarily tested and the entire code is covered with various tests. Also, each module must be fully operational before the next phase.

During the testing and integration phase, all created modules are combined into one system, which will then become the final product. Developers or a special testing team must test the entire system and fix all the errors found.

On the next delivery and deployment phase, developers must install the program or transform it into a state that can be used by end users. Usually during this phase, automatic tests are deployed on servers. And only after the above points, the system falls into the so-called state that is used by end users.

The last phase is the launch of the software, where the developers' task is to support the product or incorporate new customer requests as well as fix any bugs that were not detected during testing.

The waterfall model is very simple to use which gives it a number of advantages such as:

1. Clearly defined phases of development.
2. Offers a large amount of documentation. All stages must be documented.
3. Teams know the amount of work that needs to be done.
4. Presence of clients only at the initial and final stages.

Over time, it became clear that the model also has its drawbacks:

1. Since the requirements may not always be clear, there is not enough flexibility to change the final result in an easy way.
2. Strict division into development phases.
3. These two simple constraints led to failures during development and task distribution, and the reasons were:

4. Presence of unrealistic project goals.
5. Inaccurate project estimates.
6. Presence of risks that are difficult to control.
7. Presence of not precisely defined requirements.
8. Lack of normal reporting on the development status.
9. Lack of communication with the customer, which affects the development process of the project in a negative way and increases the development time when it is necessary to add or correct changes.
10. Usually the use of old or inappropriate technologies for writing code.
11. Also, it may be due to excessive complexity of the project.
12. Presence of commercial pressure.
13. Lack of project development practices.

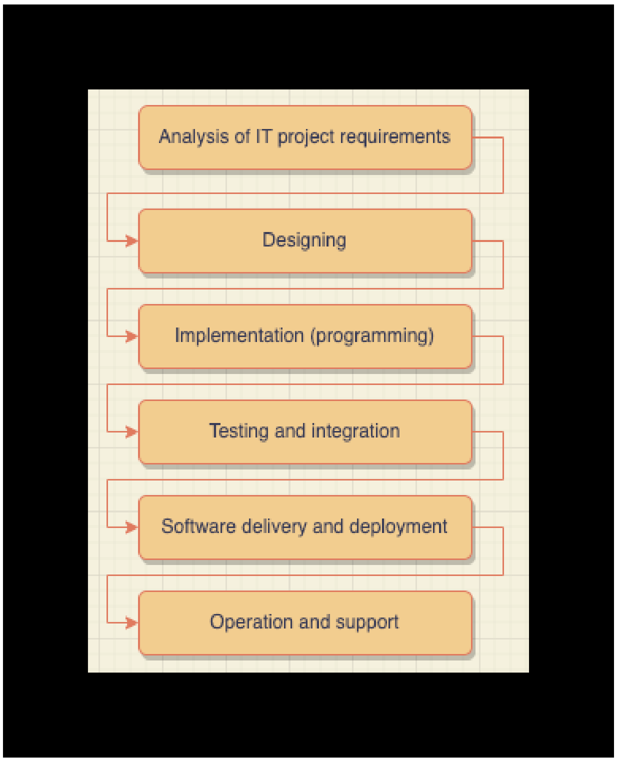


Fig. 1. Waterfall Model

In 2001, the Agile Alliance was formed to create teams of the most flexible and dynamic methods to optimize the software development process. A document was also created that defined the basic concepts of the Agile approach and was called the Agile Software Development (Agile Alliance) [4-7]. Four main concepts were created that defined the basic values of Agile, and these values should be used in all Agile technologies (Figure 2).

The new values and elements contrast sharply compared to the traditional approach, which in turn relied on clearly defined plans. Following the basic principles will help increase the likelihood of success in creating the final product for wide user use.

The differences between the two models that were analyzed and studied are given in Table 1.1.

As practice shows, the total number of failed projects, or projects that are rejected, is quite large. In the work [8, 9] it was reported that only almost 40% of all existing projects are successful. About 45% were rejected for various reasons (there are no all planned functions and capabilities of software, the allocated budget was exceeded, tasks were completed untimely, etc.). And 15% were completely unsuccessful (tasks were completed, but the software did not reach the market, or the order for software was canceled). It was also noted that since 2004 there has been a gradual increase in all successful projects from 30% on average by 5%.

It was also said in the report that the size of the development project has a great influence and this is more important than the chosen methodology regardless of whether it is fast and flexible or traditional.

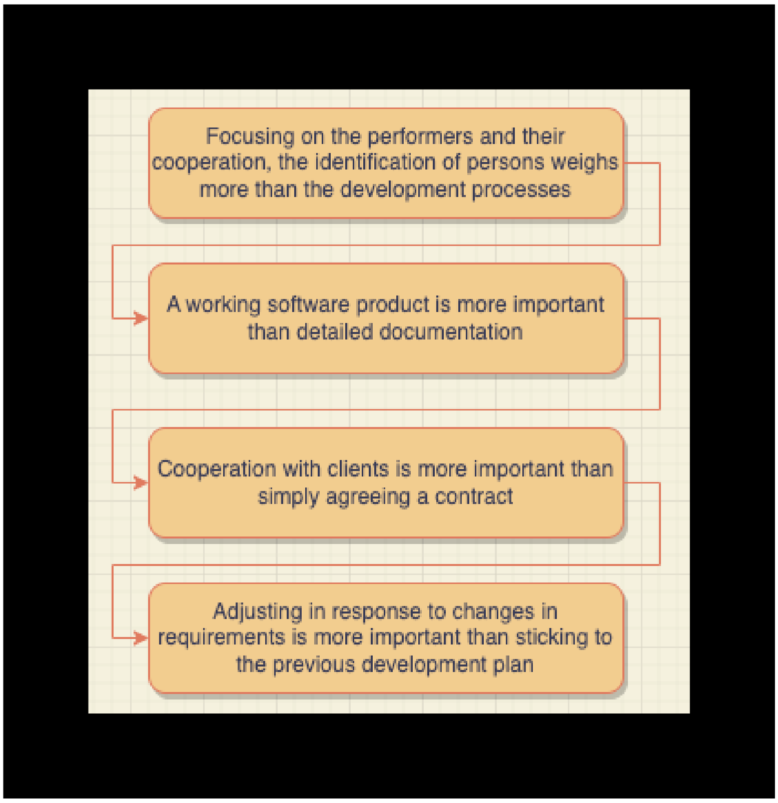


Fig. 2. Four Core Values of Agile Methodology

In the aforementioned work it was stated that a large project is much more likely to encounter problems (10 times) than a small project, and for small projects, flexible methods and approaches greatly simplify the work.

Today, the Agile methodology is expanding in use. As research [10, 11] shows, almost 45% of users who use flexible software development methods use Agile in almost all of their projects. A large percentage of organizations that participated in the survey (90%) say that they use flexible software development methods in all of their projects, as this increases the chances of successful project completion.

The Scrum approach (considered the best method) is used by 55% of Agile teams of managers and developers. If we take into account various modifications of the Scrum methodology, the overall usage rate increases to 73% [12].

Table 1

Comparison of two Agile and Waterfall methods

Agile	Waterfall
Short-term planning with the possibility of dynamic supplementation of necessary requirements.	Planning takes a lot of time since this process requires long-term planning, and requirements cannot be changed during the development of the software according to the approved execution plan and existing infrastructure within the system.
All team members will play an equal role in the team.	All team members follow the established hierarchical structure within the organization.
Constant communication with the client is carried out at all stages of development, which in turn allows to receive timely feedback from the client and understand whether everything was done correctly and clearly.	Completely independent from communication with the customer.
Thanks to the fact that continuous testing is carried out, there is a constant control of the quality of the software.	Late planning can significantly lag behind quality control, as the project is only tested at the last stages of the model and testing takes a lot of time. Also, bug fixes must be structured and present in the system.
It is convenient to measure and track the progress of the product due to measurements in sprints or units.	The entire progress is measured by the steps implemented
The deadlines for software implementation can be conveniently changed if additional requirements for the final product appear during one of the sprints.	Deadlines can be easily determined, since no new changes are expected and if there are no unexpected system failures.
Depending on the projects, they can be divided into different categories such as complex, current or long-term, etc.	The best approach for projects that are not of large size, and all steps can be planned or they are predictable and can be influenced. It is necessary to know all the requirements.

In the IT sphere, there is a widespread belief that teams using Agile in their approach significantly increase the chances of successful software development. Although this is not always confirmed by empirical data, Agile technology has long been seen as a methodology that will make a big breakthrough in software development for projects of various scales. Recent research in [14, 15, 16] studying the success of Agile projects shows that the possibility of such a belief can be proven on scientific data. The works showed a quantitative study to check whether the use of flexible Agile methods during software development affects the success rate of products. They noticed signs that the use of flexible planning methods in the software development process has a significantly higher registered success rate of products. Evaluation and display of results were conducted for three categories of success: overall project success, stakeholder success, and effectiveness.

Agile projects are well known for their flexible software development process compared to traditional projects. The process itself consists of a large set of practices that in turn describes a set of procedures that teams of developers then use to achieve the project's final goals.

For IT projects with the goal of creating quality software, the agile approach helps to determine the scope and size of the project at all stages, including the final stage when the project is in constant use by users. Size, required time, as well as cost and quality are quite important criteria when considering the success of a software product.

2. Development of an algorithm that will allow automation of the task distribution process.

In the begin, it is necessary to form a corresponding list of skills for each team member (Figure 3).

Basic skills	Basic skills
English	E(m)= E / 1.6
Self management	SM(m)= SM / 1.6
Team management	TM(m)= TM / 1.6
Experience duration	ED(m)= ED / 1.6
Calm	Ca(m) = Ca / 1.6
Stability	St(m) = St / 1.6
Responsibility	Re(m) = Re / 1.6
Quality	Qu(m) = Qu / 1.6
Design skills	DES(m) = DES / 1.6
UI skills	UIS(m) = UIS / 1.6
Backend skills	BS(m) = BS / 1.6
Devops skills	DOS(m) = DOS / 1.6
React native skill	RNS(m) = RNS / 1.6
Testing skill	TS(m) = TS / 1.6
Learning speed	LS(m) = LS / 1.6
Development speed	DS(m) = DS / 1.6

Fig. 3. A list of necessary skills describing a person

The maximum total score of all skills will be 160 points, since there are 16 skills in total. To find out how much each skill is worth in percentage, we divide each skill by 6 and get 6.25%. - . This is the maximum value that a skill can get.

Next step, we will divide these skills into 4 main groups. These are Skill (S), Management(M), Reliability(R), Prospects(P). Let's calculate the sum of all skills that belong to the Skill group.

$$S = \sum \begin{pmatrix} DES(m), UIS(m), BS(m), DOS(m), RNS(m) \\ ,TS(m) \end{pmatrix},$$

where m - this is the maximum grade in percentage. Let's now calculate the average and relative value of each skill for this group.

$$S(a) = S/6,\ S(r) = \frac{S}{0{,}375},$$

де 0,375 це –

$$(S_{max} * 6)\ /100.$$

Now let's calculate the corresponding ratings for Management (M), Reliability (R), and Prospects (P) using the same formulas.

$$M = \sum\{E(m), SM(m), TM(m), ED(m), S_a\},$$

Where S_a – this is a previously calculated value, i.e. the average value for Skill, and m is the maximum rating in percentage.

$$R = \sum\{Ca(m), SM(m), TM(m), ED(m), Re(m)\},$$

Where m - this is the maximum grade in percentage.

$$D = \sum\{LS(m), DS(m), S\},$$

where *m* – This is the maximum score in percentage ratio, and S is the sum of all skills of the Skill group.

And accordingly, the average values.:

$$M(a) = M/5,$$

$$R(a) = B/4,$$

$$D(a) = D/3,$$

And relative meanings:

$$M(r) = \frac{M(a)}{31{,}25},$$

Where 31,25 is –

$$(S_{max} * 5)/100,$$

$$R(r) = \frac{R(a)}{0.25},$$

where 0,25 is –

$$(S_{max} * 4)/100,$$

$$D(r) = \frac{R(a)}{n},$$

Where n is –

$$D(r)\frac{(S_{max} * 2 + S(r))}{100}.$$

For the tag determination system, a data set needs to be created. This data set is formed from a graph where each vertex is connected to another vertex, i.e., a many-to-many relationship.

We will identify 5 main tag groups. These will be: Priorities, Projects, Skills, Types, Phase. Besides these main groups, there may be others. Each such tag group can contain a subset of size N, where N is a natural number. To determine the tags that are related to the task, we will use the formula:

$$Tags\{N\} = \sum_{i=1}^{N}(\overrightarrow{max}(Priotities\{SubPriorities(i)\}) + \sum_{i=1}^{N}(\overrightarrow{max}(Projects\{SubProjects(i)\}) + \sum_{i=1}^{N}(\overrightarrow{max}(Phases\{SubPhases(i)\}) + \sum_{i=1}^{N}(\overrightarrow{max}(Skills\{SubSkills(i)\}) + \sum_{i=1}^{N}(\overrightarrow{max}(Types\{SubTypes(i)\}),$$

where SubPriorities $\in$ Priorities, Sub Projects $\in$ Projects, SubPhases $\in$ Phases, SubSkills $\in$ Skills.

To determine who will be responsible for completing the task, a rating system needs to be created. The ratings include: the number of tasks completed by tag group, the number of tasks completed by subgroups, the number of tasks planned to be completed, the total sum of estimations for all planned tasks, the total sum of ratings, S(r) value, M(r) value, R(r) value, D(r) value, and the sum of all other skills. After creating the corresponding ratings with positions in the sub-ratings, we form one general rating. Then, taking into account the preferred and not preferred tags of each participant, we lower or raise his position in the rating.

To calculate the impact of preferred and not preferred tags, we use the following formulas:

$$preferred = \prod_i 10 * \{P_i\}/100,$$
$$not\ preferred = \prod_i 10 * \{P_i\}/100,$$

Where - P(i) is a set of preferred and not preferred tags for a task.
The formula for the influence on the overall rating in the ranking is:

$$mark = total\ mark + (preferred - not\ preferred).$$

To influence the general time for task completion, simply multiply the rating by the corresponding value in the time range {2,5,10,15,20}. The next step is to find the maximum value in the rating and in this way we find out who our potential person is best suited for the task.

4. Results of distribution and evaluation of tasks

For the experimental research, a dataset generated by us was used. To generate the dataset, a graph structure was taken as a basis, where each node has a connection with all other nodes (Figure 4). For a simple visualization of the prediction of the optimal candidate, let's take as an example the description of the task - "Me as a super admin, should be able to create a form for inviting users", and no qualification tags will be added. The results of auto-tagging the task can be seen in Figures 5, 6. In Figure 5, it can be seen that the best classification tag for priority is High. Sprint is the best for phase (Figure 6). With a big gap in Skills, the tag is React. And for the mandatory Type we have Task.

Next, the obtained tags need to be used to search for the best candidate to perform the task.

At the first stage, you can see the rating of all players for all tags (Figure 7). Then from the current rating it is necessary to form a new one taking into account the load of people. Then another one with the influence of preferred tags (Figure 8) and not preferred (Figure 9). Then build the last rating to select the best performer of the task (Figure 1.10). From the example we see that Attendee - 8 is the best option.

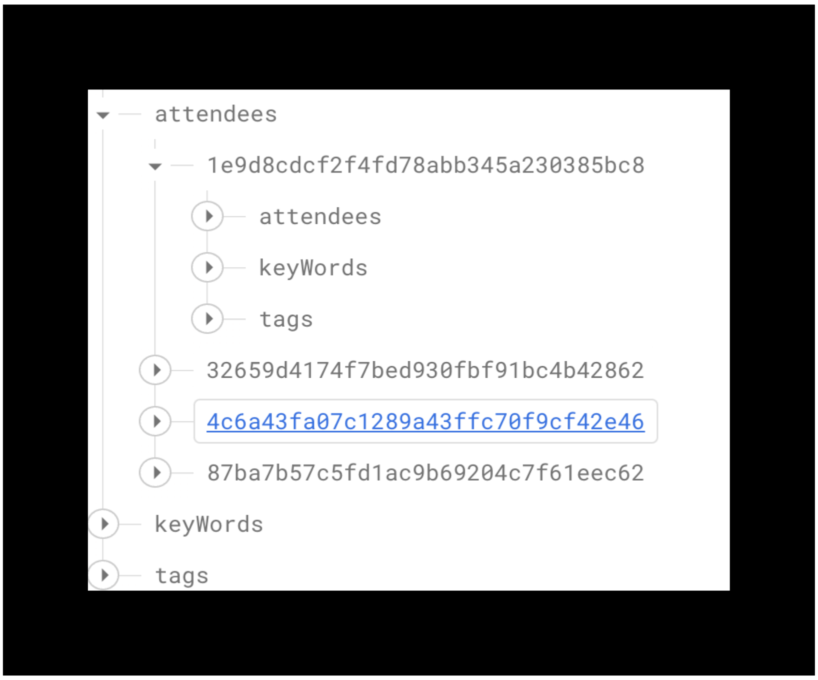


Fig. 4. Dataset structure

In Figure 4 you can see a clearly structured entity where each key has an attendee Id which is hashed. In turn, each attendee has a link to each attendee, including itself. This includes the key words which are also hashed, of the task described and all existing and used tags in the system. This will help in the future when we generate the most compatible and appropriate tags for task description.

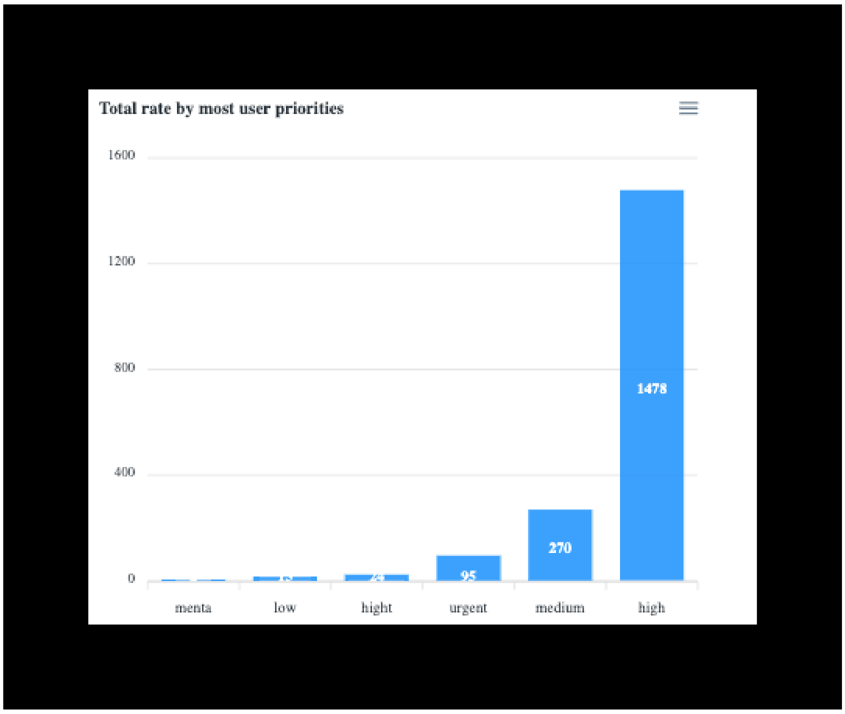


Fig. 5. Total rate by most used priorities

In the Figure 5 it can be seen that out of all the tags present in the graph structure (height, low, urgent, medium), the tag 'high' is the most frequently used for all the words described in the task "Me as a super admin, should be able to create a form for inviting users". This tag was used 1478 times. It has a huge gap from the tag 'medium' which has only 270. This suggests that the tag 'high' is probably the most optimal option for the described task.

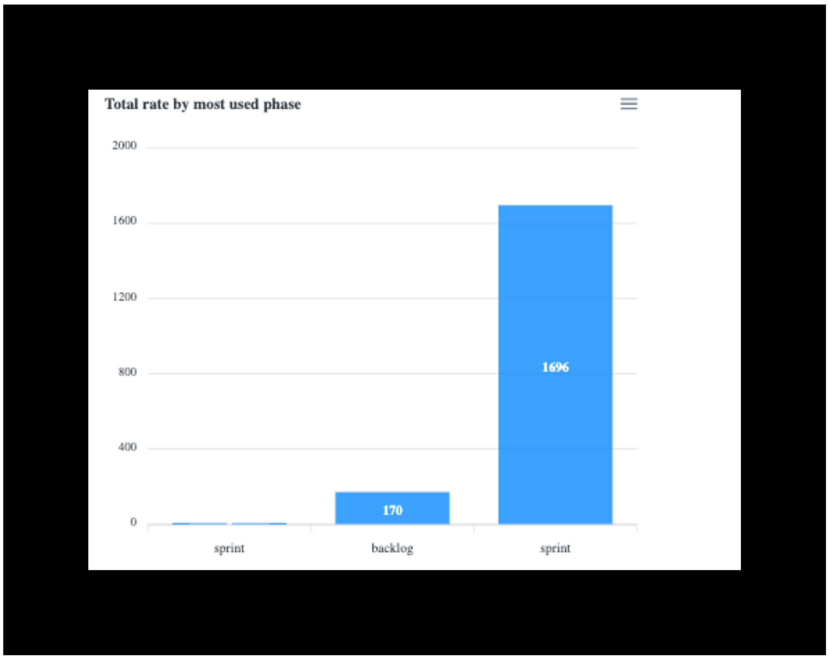


Fig. 6. Total rate by most used phase

In the Figure 6 it can be seen that out of all the tags present in the graph structure (sprint, backlog), the tag 'sprint is the most frequently used for all the words described in the task "Me as a super admin, should be able to create a form for inviting users". This tag was used 1696 times. It has a huge gap from the tag 'backlog' which has only 170. This suggests that the tag 'sprint' is probably the most optimal option for the described task.

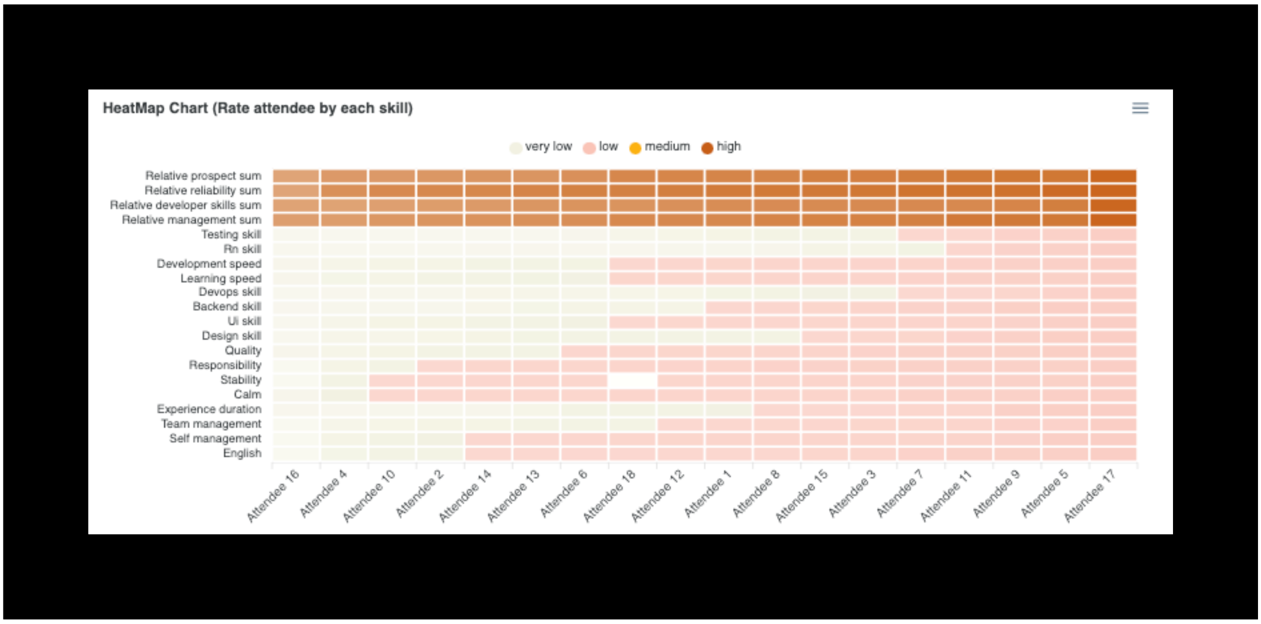


Fig. 7. Total heatmap of attendees by all tags

In the Figure 7, you can see the rating of the attendees and the rating of each attendee for each skill. High is the value from 10 to 110, medium is the value from 7 to 10, low is the value from 3 to 7, and very low is the value from 0 to 3. As we can see from the graph, the top 3 best candidates according to the ratings of all skills are Attendee 9, Attendee 5, and Attendee 17. However, these top 3 are not our most optimal candidates yet, as we need to make a rating list only for the tags we got in Figures (3-4) and also take into account the preferred and not preferred tags of each participant in the rating.

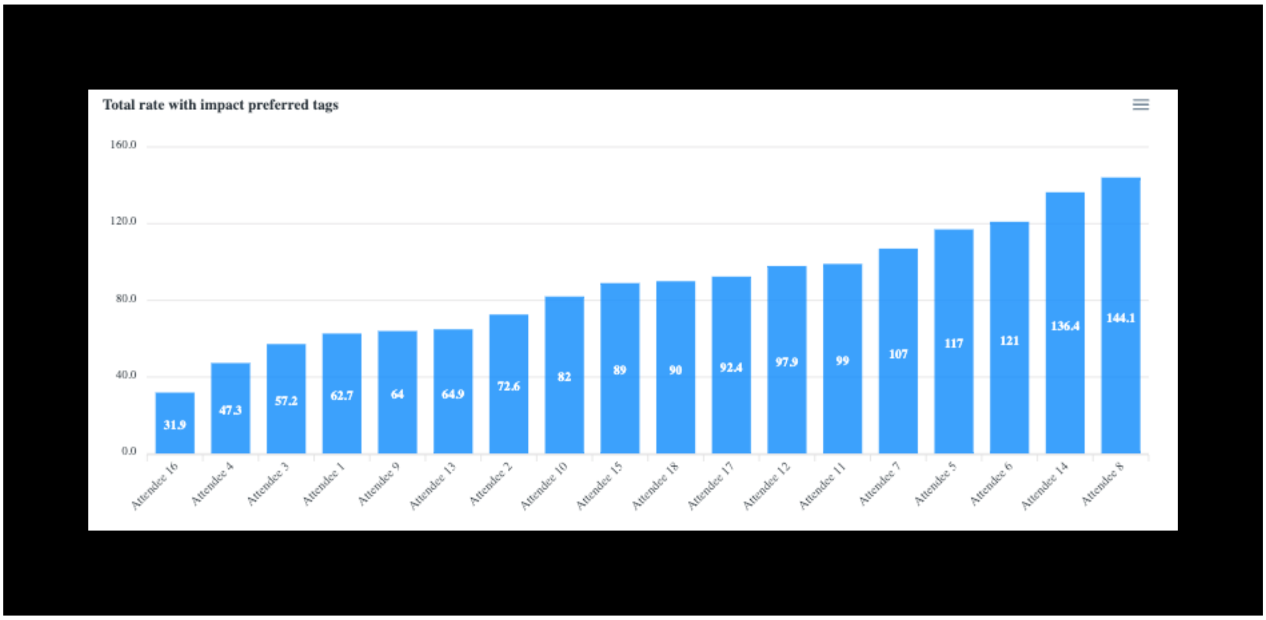


Fig. 8. Total rate with impact preferred tags

Taking into account the tags that participants would not like to work with, we added all the scores by tags and took into account the wishes we received, we got one overall rating as in Figure 6. From it we see that the candidates who were in the previous rating Attendee 9, Attendee 5, and Attendee 17 changed their position in this rating. Now the top 3 are taken by participants Attendee 6 - 121 points, Attendee 14 - 136.4 points, and Attendee 8 - 144.1 points. The next step is also to take into account the tags that participants did not want to work with in the already fading rating.

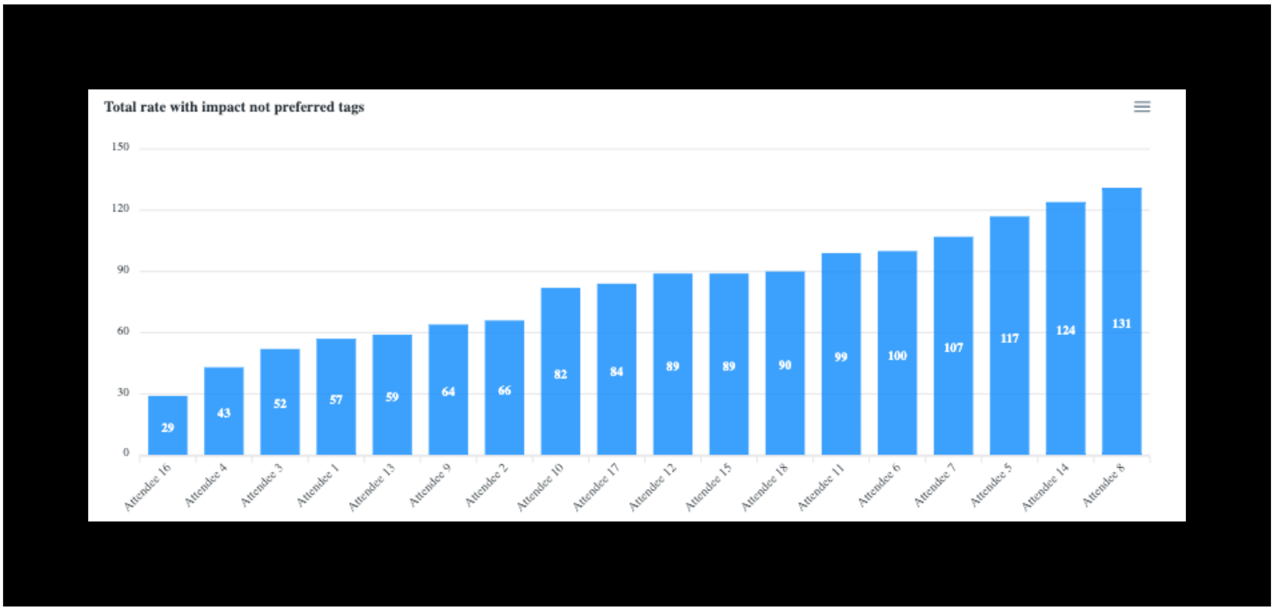


Fig. 9. Total rate with impact not preferred tags

Taking into account the tags that participants would not want to work with, we changed the ranking based on the number of tags that participants would not want to work with. We got one overall ranking as shown in Figure 7. From it we see that the candidates who were in the previous ranking Attendee 6 - 121 points Attendee 14 - 136.4 points, and Attendee 8 - 144.1 points changed their position in this only Attendee 6 to Attendee 5. Now the top 3 are occupied by participants Attendee 5 - 117 points Attendee 14 - 124 points, and Attendee 8 - 131 points. As we see, the scores of all participants from the top 3 have decreased. This indicates that these participants from the top 3 have at least one tag with which they would not want to work. The next step is also to take into account the workload of people according to the general estimation of tasks and the total number of tasks with the status to do.

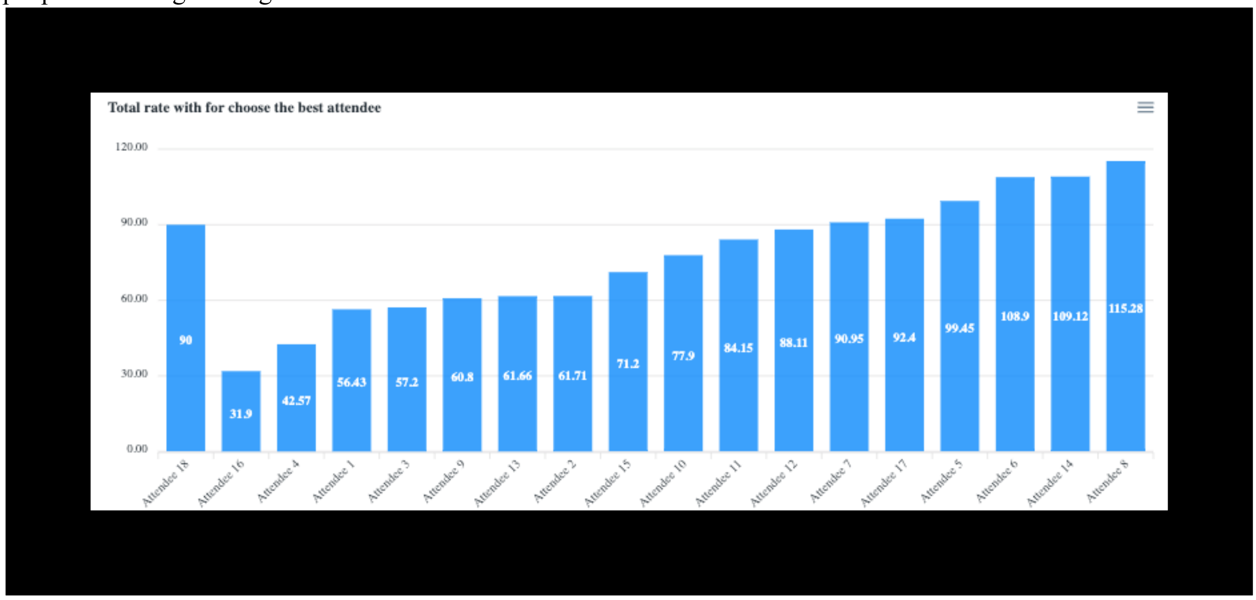


Fig. 10. Total rate for choose the best attendee

Taking into account the workload of people according to the general estimation of tasks and total tasks with status to do, we changed the rating and got one general rating as in Figure 8. From it we see that the candidates who were in the previous rating Attendee 5 - 117 points, Attendee 14 - 124 points, and Attendee 8 - 131 points again changed their position Attendee 5 to Attendee 6. Now the top 3 are occupied by participants Attendee 5 - 108.9 points, Attendee 14 - 109.12 points, and Attendee 8 - 115.28 points. As we see, the scores of all participants from the top 3 have decreased. This speaks of the fact that these participants from the top 3 have a certain number of tasks that affects the overall rating, but not significantly. And Attendee 5 is already loaded to a significant extent and cannot be given new tasks. Since there are no participants with the same scores, it can be said from Figure 8 that the most optimal option for task execution is Attendee 8. Since he has 115 points and he has a slight gap with participants Attendee 6 (108) and Attendee 14 (109).

Conclusions:

An automated information system has been proposed for determining the best candidate to perform a task, as well as for classifying the task and adding classification tags to it, which ensures accurate selection of the best performer of the task and performs accurate task classification and determines the corresponding tags. Further research is focused on increasing the accuracy and autonomy of the model and improving the level of classification to the level of subgroups of tags.

The prosed system of distribution and evaluation of tasks in the software development process is aimed at improving the accuracy and autonomy of the model, as well as improving the level of classification to the level of subgroups of tags. The system will use machine learning algorithms to identify the best candidate for a task, as well as to classify the task and add the corresponding tags. The system will also be able to identify the most suitable tasks for a particular candidate, based on their skills and experience. Additionally, the system will be able to provide feedback to the candidate on their performance, allowing them to improve their skills and become more efficient in their work. The system will also be able to track the progress of the task and provide timely updates to the stakeholders. Finally, the system will be able to provide detailed reports on the performance of the task and the candidate, allowing for better decision-making.

The proposed system of distribution and evaluation of tasks in the software development process for Scrum and Agile methods is designed to provide an automated and accurate way of assigning tasks to the best candidate, as well as classifying the task and adding the corresponding tags. The system is designed to be highly accurate and autonomous, and to provide a level of classification that is accurate down to the level of subgroups of tags. The system is also designed to be highly efficient and to provide a streamlined process for assigning tasks to the best candidate. The system is also designed to be highly flexible and to be able to adapt to changing requirements and tasks. Further research is focused on improving the accuracy and autonomy of the system, as well as improving the level of classification to the level of subgroups of tags. Additionally, research is being conducted to improve the system's ability to handle complex tasks and to provide a more efficient and streamlined process for assigning tasks to the best candidate.

References

1. Sudarkina S., Klimentova M., Anichkina I. Methods and Tools of Strategic Marketing Planning. / Sudarkina S., Klimentova M., Anichkina I. // Bulletin of the National Technical University "Kharkiv Polytechnic Institute" (Economic Sciences), (24), 2019. - Kharkiv: KhNTU, 2019. - Pp. 66-78.
2. Barkova K. Methodological Foundations of Strategic Planning of the Enterprise / Barkova K. // Recommended for printing at the meeting of the Scientific Council of the Kharkiv National Economic University named after S. Kuznetsov, Protocol No. 8 of May 3, 2019. - Kharkiv: KhNEU, 2019. - Pp. 41-49.
3. Kashtalyn I.V. Using Artificial Intelligence Systems to Improve Task Planning Tools / I.V. Kashtalyn, R.V. Kulinyak // III Scientific and Practical Conference of Young Scientists and Students "Intelligent Computer Systems and Networks". November 26, 2020, -Ternopil, Ukraine. Ternopil: ZUNU, 2020. - Pp.13-21.
4. Il'kov A.V. Features of Strategic Planning in Small Enterprises in Conditions of Uncertainty / Il'kov A.V. // Ensuring Sustainable Development of the Agricultural Sector of the Economy: Problems, Priorities, Prospects: Materials of the XI International Scientific and Practical Internet Conference October 29-30, 2020: In 2 vol. - Volume 1. - Dnipro: Publishing and Polygraphic Center "Garant SV", 2020. - Pp.72-74.
5. Ostapenko R.M. Organizational Mechanism of Strategic Planning of Agricultural Enterprises / Ostapenko R.M. // Materials of the II International Scientific and Practical Conference (online form) "Formation and Prospects of Development of Entrepreneurial Structures within the Integration into the European Space" - Poltava, 2019. - 715 p.
6. Kazimir V.V. Information Bases for Building Telecommunication Networks / V.V. Kazimir, V. Litvinov, S.M. Shkarlet, S.V. Zaitsev // Bulletin of Chernihiv State Technical University. - Chernihiv: ChDTU, 2013. - 340 p.
7. Steklov V.K. Information System: A Textbook for Students of Higher Educational Institutions in the Direction of "Telecommunications" / V.K. Steklov, L. Berkman. - K .: Technique 2014. - 792 p.
8. Stasev Yu.V. Computer Networks. Technologies and Protocols for Modeling: Tutorial / I.V. Ruban, S.V. Dudenko, O.I. Timochko. - Kh.: KhU P S, 2014. - 359 p.
9. Krivutsa V.G. Management of Telecommunications with the Application of Modern Technologies / V. Krivutsa, V.K.Steklov, L.N.Berckman, B.Kostik, B.Oliynyk, S.Sklyarenko // Textbook for Higher Education Institutions. - K.: Tekhnika, 2007. - 384 p.
10. Steklov V.K. Information System: Textbook for Students of Higher Educational Institutions in the Direction of "Telecommunications" / V.K. Steklov, L. Berckman. - K.: Tekhnika, 2014. - 792 p.
11. Krivutsa V.G. Management of Telecommunications with the Application of Modern Technologies / V.G. Krivutsa, V. Steklov, L. Berckman, B.Ya. Kostik, B. Oliynyk, S.M. Sklyarenko // Textbook for Higher Education Institutions. - K.: Tekhnika, 2007. - 384 p.
12. Gorbatiy, I.V. Telecommunications Systems and Networks. Principles of Operation, Technologies and Protocols: Tutorial / I.V. Gorbatiy, A.V. Bondarev // - Lviv: Publishing House of Lviv Polytechnic, 2016. - 336 p.
13. Sovetov, B.Ya. Modeling Systems: Textbook for Bachelors / B.Ya. Sovetov, S.A. Yakovlev. - 7th ed. - M.: Yurait Publishing, 2015. - 343 p.Chen S. Speaker, environment and channel change detection and clustering via the
14. Klimash M.M. Modern Transformations in Distributed System Architectures: Monograph / M.M. Klimash, A. Luntovsky, V. Romanchuk. - Lviv-Drohobych: Kolo, 2015. - 328 p.
15. Luntovsky A.O. Stages of Development of Modern Info-Telecommunication Services and Energy Efficiency of Network Technologies / A.O. Luntovsky, P. Gus'kov, A. Masiuk // Bulletin of the National University "Lviv Polytechnic". Series: Radioelectronics and Telecommunications. - Lviv: Publishing House of Lviv Polytechnic, 2014. - No. 796. - P. 131-139.
16. Brooks F. Mythical Man-Month, or How Software Systems Are Created. - St. Petersburg: Symbol-Plus, 2006. - 304 p.
17. Vendrov A.M. Design of Software for Economic Information Systems: Textbook. - 2nd ed., rev. and add. - Moscow: Finances and Statistics, 2005. - 544 p.: ill.
18. Iles P. What is Software Architecture? - Resource: http://www.interface.ru/home.asp?artId=2116
19. Kotzubay I.Yu., Chunaev A.V., Shikov A.N. Basics of Designing Information Systems: Tutorial. - St. Petersburg: ITMO University, 2015. - 206 p.

20. Minukhin S.V., Besedovsky O.M., Znakhur S.V. Methods and Models of Design Based on Modern CASE Tools. Tutorial. - Kharkiv: Vyd. KhNEU, 2008. - 272 p. (Ukrainian).
21. Pushchin M.N. Design of Information Systems: Tutorial. - Moscow: MIET Publishing House, 2008. - 234 p.
22. Design of Information Systems: Textbook / V.I. Grekul, G.N. Denischenko, N.L. Korovkina. - M. Internet-Un-t Information Technologies, 2005. - 304 p.
23. Reengineering of Business Processes. - Resource: https://library.if.ua/book/28/1899.html
24. Sommerville I. Software Engineering, 6th Edition.: Trans. from English. - M.: Williams, 2002. - 624 p.
25. Design Technologies. - Resource: https://studfile.net/preview/3997729/page:5
26. Fowler M. et al. Architecture of Corporate Software Systems. - M.: Williams, 2005. - 576 p.

Dmytro Okrushko **Дмитро Окрушко**	Master, Khmelnytskyi National University e-mail: okrushko@i.ua	магістр, Хмельницький національний університет
Antonina Kashtalyan **Антоніна Каштальян**	Ph.D, Associate Professor of Department of Computer Engineering and Information Systems, Khmelnytskyi National University e-mail: yantonina@ukr.net https://orcid.org/0000-0002-4925-9713	кандидат технічних наук, доцент кафедри комп'ютерної інженерії та інформаційних систем, Хмельницький національний університет

Full requirements for the design of the manuscript
Повні вимоги до оформлення рукопису
http://csitjournal.khmnu.edu.ua/

No editorial responsibility is required for the content of messages sub.
За зміст повідомлень редакція відповідальності не несе

To print 29.06.2023. Mind. Printing. Arch. 10,35. Obl.-vid. Arch. 9,72
Format 30x42 / 4, offset paper. Another risography.
Overlay 100, deputy. №

Підп. до друку 29.06.2023. Ум. друк. арк. 10,35. Обл.-вид. арк. 9.72
Формат 30x42/4, папір офсетний. Друк різографією.
Наклад 100, зам. №

Replication is made from the original layout, made edited
by the magazine "Computer Systems and Information Technology"

Тиражування здійснено з оригінал-макету, виготовленого
редакцією журналу "Комп'ютерні системи та інформаційні технології"

Editorial and publishing center of Khmelnytskyi national university
29016, Khmelnytskyi, street Institutska, 7/1, tel. (0382) 72-83-63

Редакційно-видавничий центр Хмельницького національного університету
29016, м. Хмельницький, вул. Інститутська, 7/1, тел. (0382) 72-83-63