https://doi.org/10.31891/csit-2024-2-11
UDC 681.3

Oleksandr BRYK, Ivan STEFANYSHYN,
Volodymyr STEFANYSHYN, Oleh PASTUKH
Ternopil Ivan Puluj National Technical University

# ROBUSTNESS EVALUATION OF MACHINE LEARNING ALGORITHMS FOR NEUROCOMPUTER INTERFACE SOFTWARE USING DISTRIBUTED AND PARALLEL COMPUTING

*This article highlights the field of information technology for brain-computer interaction, and the main goal is to use it to determine patterns of human brain activity using electroencephalography (EEG) data. During the execution of the article, machine learning methods were used, namely such classifiers as Random Forest, Multi-Layer Perceptron (MLP), and Logistic Regression. The investigation begins with real-world experiments recording EEG signals during finger movement tasks, providing valuable insight into the complex dynamics of brain operation and interaction.*

*Utilizing 10-fold cross-validation, the performance of each classifier is rigorously evaluated across various metrics, including accuracy, f1_weighted та roc_auc_ovr_weighted. Through this process, the robustness and consistency of classifier performance are assessed, with dispersion values computed to gauge variability across iterations. The findings reveal nuanced differences among the classifiers, with MLP demonstrating the highest robustness, followed by Logistic Regression and Random Forest.*

*The main goal of the article was to find out the importance of such a classifier performance parameter as robustness. Software robustness is a key characteristic, especially in medical applications, where consistent and reliable performance of information technology is paramount. Neural interfaces offer many avenues for solving various limb problems, spinal cord injuries, and neurological diseases in humans. These devices contribute to improving the quality of life by minimizing these problems in people, which leads to increased mobility and functional capabilities of people. The article also emphasizes the potential associated with the transformation of neurointerface technologies in expanding human capabilities and revolutionizing human-machine interaction.*

*In conclusion, the research contributes to advancing the field of brain-computer interaction by leveraging machine learning algorithms to decode neural signals and uncover hidden patterns within EEG data. By identifying the most stable classifier, the study lays the groundwork for the development of robust neurointerface technologies with practical applications in healthcare, rehabilitation, and beyond. Through interdisciplinary collaboration and innovative methodologies, the journey towards unlocking the full potential of brain-computer interaction continues, promising new horizons in human augmentation and technological innovation.*

*Keywords: information technology, neuro-interface of brain-computer interaction; artificial intelligence; parallel programming; high-performance computing; classifier; robustness; accuracy; scalar, dispersion.*

Олександр БРИК, Іван СТЕФАНИШИН,
Володимир СТЕФАНИШИН, Олег ПАСТУХ
Тернопільський національний політехнічний університет імені Івана Пулюя

# ОЦІНКА СТІЙКОСТІ АЛГОРИТМІВ МАШИННОГО НАВЧАННЯ ДЛЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ НЕЙРОКОМП'ЮТЕРНОГО ІНТЕРФЕЙСУ З ВИКОРИСТАННЯМ РОЗПОДІЛЕНИХ ТА ПАРАЛЕЛЬНИХ КОМП'ЮТЕРНИХ ОБЧИСЛЕНЬ

*Ця стаття висвітлює галузь інформаційних технологій для взаємодії мозок-комп'ютер, і головна мета полягає в тому, щоб використовувати їх для визначення моделей активності мозку людини за допомогою даних електроенцефалографії (ЕЕГ). Під час виконання статті були використані методи машинного навчання, а саме такі класифікатори, як випадковий ліс, багатошаровий персептрон і логістична регресія. Досліджування починається з реальних експериментів із записом сигналів ЕЕГ під час виконання завдань з рухами пальців, що дає цінне розуміння складної динаміки роботи та взаємодії мозку.*

*Використовуючи 10-кратну перехресну перевірку, продуктивність кожного класифікатора ретельно оцінюється за різними показниками, включаючи accuracy, f1_weighted та roc_auc_ovr_weighted. За допомогою цього процесу оцінюється надійність і узгодженість продуктивності класифікатора з обчисленням значень дисперсії для вимірювання мінливості між ітераціями. Результати виявляють тонкі відмінності між класифікаторами, причому багатошаровий персептрон демонструє найвищу стійкість, за ним йдуть логістична регресія та випадковий ліс.*

*Основною метою статті було з'ясувати важливість такого параметра продуктивності класифікатора, як стійкість. Стійкість програмного забезпечення є ключовою характеристикою, особливо в медичних додатках, де послідовна та надійна робота інформаційних технологій має першочергове значення. Нейроінтерфейси пропонують багато способів вирішення різних проблем кінцівок, травм спинного мозку та неврологічних захворювань у людей. Ці пристрої сприяють покращенню якості життя шляхом мінімізації цих проблем у людей, що призводить до збільшення мобільності та функціональних можливостей людей. У статті також наголошується на потенціалі, пов'язаному з трансформацією технологій нейроінтерфейсів у розширенні можливостей людини та революції у взаємодії людини з машиною.*

*Підсумовуючи, дослідження робить внесок у розвиток взаємодії мозку та комп'ютера шляхом використання алгоритмів машинного навчання для декодування нейронних сигналів і виявлення прихованих закономірностей у даних ЕЕГ. Визначивши найстабільніший класифікатор, дослідження закладає основу для розробки надійних технологій нейроінтерфейсу з практичним застосуванням у сфері охорони здоров'я, реабілітації тощо. Завдяки міждисциплінарному співробітництву та інноваційним методологіям продовжується шлях до розкриття повного потенціалу взаємодії мозку та*

*комп'ютера, що обіцяє нові горизонти в розвитку людського потенціалу та технологічних інноваціях.*
*Ключові слова: інформаційні технології, нейроінтерфейс взаємодії мозок-комп'ютер; штучний інтелект; паралельне програмування; високопродуктивні обчислення; класифікатор; стійкість; точність; скаляр, дисперсія.*

## Introduction

Improving neural interfaces for brain-computer interaction is an important scientific pursuit with far-reaching implications. These interfaces offer hope to people facing physical challenges such as limb loss or spinal cord injuries, promising to restore mobility and functionality. In addition, they symbolize the emergence of a new era that expands human capabilities in augmented and virtual reality.

Consider people struggling with limb loss or spinal cord injuries. For them, neural interfaces represent more than just technological progress; they embody the potential to regain control over their bodies, ease daily struggles, and regain independence. These interfaces open unprecedented opportunities for interaction with digital environments, thereby expanding the dimensions of the human experience in augmented and virtual reality.

In essence, the development of neural interfaces is a key effort aimed at revolutionizing the world and reimagining human capabilities. On the other hand, we do not need to forget about ethical considerations during the path of scientific innovation. These considerations must always be followed because these qualities serve as a guarantee that such research and similar ones will serve for the greater good of all mankind.

## Related works

In the burgeoning field of neural interfaces for brain-computer interaction, significant contributions have emerged from both industry and academia. Notable among these are pioneering efforts by leading neurotechnology companies such as Neuralink [1] and academic researchers like Gilja et al. [2]. While Neuralink focuses on developing invasive brain-computer interfaces, Gilja et al. have made substantial strides in the clinical translation of high-performance neural prostheses. These initiatives showcase the diverse approaches being explored to advance neuro-interface technology and its practical applications.

Supplementing these industry-driven endeavors are a multitude of academic research findings published in esteemed journals and presented at conferences. For example, Ajiboye et al. demonstrated the restoration of reaching and grasping movements through brain-controlled muscle stimulation in individuals with tetraplegia [3]. These studies underscore the ongoing progress in neuro-interface research and highlight the transformative potential of neural interfaces in improving the quality of life for individuals with motor impairments.

## Statement of the problem

In the realm of neuro-interface development, a critical challenge lies in ensuring the robustness of classifiers for accurate movement decoding. The primary goal is to enhance robustness to define precise movements reliably, without compromising computational efficiency. Robustness in classifiers is crucial for optimizing real-time interaction and accuracy.

The complexity of processing neural signals demands sophisticated algorithmic approaches and robust software and hardware integration. However, these advancements often pose a challenge to balancing efficiency and robustness. Innovative methodologies are required to enhance robustness while preserving computational efficiency.

Furthermore, optimizing robustness holds profound implications for the reliability and practicality of neuro-interface technology. By improving robustness in classifiers, researchers can maximize accuracy and enhance the integration of neuro-interface technology into practical applications. Therefore, prioritizing robustness represents a pivotal endeavor in unlocking the transformative potential of neuro-interface technology and revolutionizing human-machine interaction.

## Purpose

The primary objective of this paper is to decode human brain activity patterns using electroencephalography data, supplemented by rigorous comparative analysis of different classifiers to enhance robustness. By employing advanced machine learning techniques and comparing the performance of various classifiers, we aim to identify the most effective approach for decoding EEG signals with optimal robustness. Furthermore, we aspire to develop algorithmic software and hardware infrastructure for neuro-interfaces, leveraging parallel programming on high-performance computing clusters. Through this interdisciplinary approach and rigorous comparative analysis, we seek to advance the field of brain-computer interaction and pave the way for groundbreaking technological innovations.

## Presenting of the main material

The material presented in this article is derived from real experiments using EEG to collect data related to the neural activity of the human brain. These experiments study the complex dynamics of brain activity and interaction by using EEG technology to analyze neural signals in a variety of experimental settings. Researchers use EEG because it is the main tool for them to gain an invaluable understanding of the neural interaction of cognitive processes, motor functions, and sensory perception. After receiving the EEG data, experimental conclusions can be drawn, which will be the basis for understanding the mechanisms underlying brain-computer interaction in the future, and will guide the development of advanced neurointerface technologies.

In this experiment, participants were told to follow instructions and perform a series of finger movements while EEG data was recorded. Before the beginning, each finger was assigned a number from 1 to 5 before the start of the experiment. The count started with the thumb and was marked as 1. Participants performed continuous finger movements, including finger combinations 1 through 5, carefully choreographed to ensure consistency and control throughout the experiment. Multiple recording sessions were conducted, each lasting approximately 2 minutes, with participants instructed to perform only one combination of finger movements per session to maintain focus and minimize variability.

Below are two illustrations of the obtained EEG data. The first depicted EEG signals recorded during finger movements involving the third and fourth fingers, capturing the neural activity associated with the coordinated movements of these digits (see Fig. 1). The second figure displayed EEG signals recorded solely during movements of the third finger, offering insight into the neural activity specifically linked to the isolated movement of this digit (see Fig. 2). These visual representations offered valuable insights into the neural dynamics underlying different finger movements and contributed to our understanding of motor control and brain function.

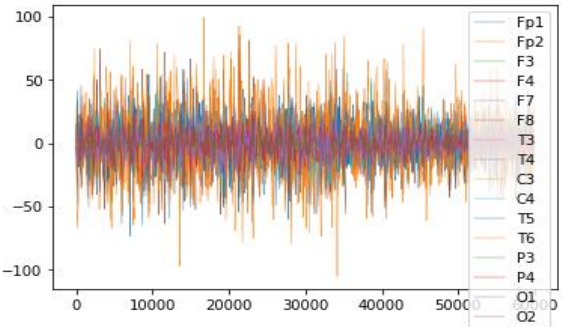


**Fig 1. Illustration of EEG Signals During Movements of Third and Fourth Fingers**

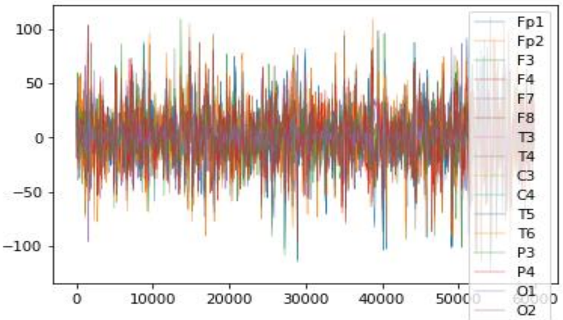


**Fig 2. Illustration of EEG Signals During Movement of Third Finger**

The EEG data depicted in the images reveal intricate neural patterns associated with finger movements. While the data appear chaotic to the human eye, modern technology enables the extraction of valuable information from these signals. Despite their complexity, these EEG signals offer valuable insights into neural dynamics during motor tasks, providing crucial information for advancements in brain-computer interface technology and our understanding of brain function.

## Explaining of investigation

In our experiment, we collected multiple files containing EEG data, each corresponding to a specific finger combination task. These files comprised time series of EEG signals captured during the execution of finger movement tasks. To ensure efficient analysis, we meticulously filtered the data, retaining only the columns relevant to our study and discarding extraneous information. Moreover, we augmented the dataset by adding a new column in each file, specifying the associated finger combination for every record. This additional metadata proved invaluable, enabling us to discern between different motor tasks effortlessly during subsequent analyses. Through this structured data preparation approach, we optimized the dataset for focused analysis on EEG signals relevant to finger movements, while retaining crucial metadata for accurate interpretation of the results.

In the next phase of our investigation, we employed three different classifiers: Random Forest, MLP, and Logistic Regression. Each of these classifiers offers unique advantages and operates on distinct principles. Random Forest is an ensemble learning method that builds many decision trees during training and outputs the class mode (classification) or average prediction (regression) of individual trees. Also, this classifier is known for its resistance to noise and conversion. It is especially effective for data processing where there is a large dimension with complex relationships. MLP is a classifier which is a type of artificial neural network. It is characterized by several layers of nodes, each of which is connected to the next layer. Such a classifier is capable of learning complex patterns in data and is widely used for tasks such as pattern recognition and classification. MLP classifiers are known for their flexibility and ability to capture non-linear relationships in data. As for Logistic Regression, it is a linear model for binary classification. This classifier estimates the probability that an instance belongs to a certain class. Although Logistic Regression is simple, it can be surprisingly effective in many cases, especially when the data is linearly distributed or when interpretability is important.

During our research, we compared the robustness parameter of these classifiers. Also, we evaluated their accuracy using multiple runs of the same data set. In this case, the robustness was expressed by the consistency of the performance of the classifiers in different data runs. After that, the best classifier that showed low variance between accuracies in several runs was selected. The low dispersion indicates a stable and reliable performance of these classifiers. By comparing the robustness of Random Forest MLP, and Logistic Regression, we aimed to determine the most appropriate classifier for our EEG data analysis task.

We applied cross-validation to calculate the robustness of each classifier. The *cross_val_score* function implemented in the *sklearn* package [9] was used to perform this check. This cross-validation consists of dividing the data set into several subsets or convolutions and iteratively training the classifier on each of the subsets while evaluating its accuracy on the remaining convolution. This iterative process is repeated several times, with each fold in turn serving as both the training and test sets. The cross-validation function from the *sklearn* package allowed us to specify the number of convolutions and select appropriate evaluation metrics, such as accuracy, for evaluation. For each classifier – Random Forest, MLP, and Logistic Regression – we used cross-validation to calculate an accuracy score over multiple iterations. This approach can provide a more reliable classifier estimate compared to a single test because it contains data variability and reduces the risk of overfitting.

We analyzed the arguments of the cross-validation function of the *sklearn* package and highlighted the *n_jobs* argument. This parameter plays a key role in the optimization of computing resources, especially in the context of high-performance computing and parallel programming. This option can ensure full utilization of the computing power of users of CPUs, GPUs, or distributed computing clusters. In high-performance computing environments where computing tasks are distributed across multiple processors to achieve maximum performance, setting n_jobs appropriately can significantly improve performance. For example, by setting *n_jobs* to -1, the cross-validation process can exploit all available computing resources, enabling parallel execution of multiple-fold iterations simultaneously. This parallelization not only accelerates the training and evaluation of machine learning models but also ensures efficient utilization of hardware resources. Furthermore, in parallel programming paradigms, such as shared-memory or distributed-memory systems, adjusting *n_jobs* allows developers to fine-tune the degree of parallelism based on the available hardware infrastructure and workload requirements. Therefore, the *n_jobs* argument serves as a crucial parameter for optimizing performance, scalability, and resource utilization in high-performance computing environments, ultimately enhancing the efficiency of machine learning algorithms in neural interface software development.

When performing the cross-validation function, we specified the metrics as *accuracy*, *f1_weighted*, and *roc_auc_ovr_weighted*. Each of these metrics was evaluated using 10-fold cross-validation for every classifier. This approach ensured a more reliable estimate of classifier performance compared to a single train-test split, as it considered data variability and minimized the risk of overfitting. By aggregating the accuracy scores obtained across all folds, we obtained a comprehensive assessment of each classifier's performance. Comparing the scores obtained through cross-validation allowed us to identify the classifier that exhibited the highest level of accuracy and robustness across iterations, guiding our selection of the most suitable model for our EEG data analysis task.

Next, we calculated the dispersion for each metric that was obtained from the 10-fold cross-validation. In this case, the dispersion serves to represent the consistency of the accuracy of the classifiers in different folds. We used the *std* function from the *NumPy* library to calculate the dispersion value for each of the metrics. After performing the calculations, we got a tuple containing three elements: the dispersion for each metric – *accuracy*, *f1_weighted*, and *roc_auc_ovr_weighted* – each representing the dispersion for the corresponding metric.

To compare the robustness of each classifier, we aimed to condense the dispersion values for *accuracy*, *f1_weighted*, and *roc_auc_ovr_weighted* metrics into a single scalar value. This scalar representation provided a holistic measure of the dispersion of the classifier's performance across all metrics. To achieve this, we represented the resulting tuple containing three elements as a vector in three-dimensional space. To obtain a scalar value, we calculated the modulus of the received vector for each of the classifiers. This involved computing the square root of the sum of the squares of the dispersion values for each metric (see Formula 1).

$$dispersion = \sqrt{{dispersion_{accuracy}}^2 + {dispersion_{f1_{weighted}}}^2 + dispersion\_roc\_auc\_ovr\_weighted^2} \qquad (1)$$

The modulus of the vector served as a measure of the total dispersion in the robustness of the classifier. By evaluating the total dispersion in this manner, we derived a single scalar value that facilitated direct comparison between different classifiers. This approach enabled us to identify the classifier with the most consistent and stable performance, aiding in the selection of the optimal model for our EEG data analysis task.

**Calculating of dispersion**

As a preliminary step, we subjected the Random Forest classifier to a 10-fold cross-validation process, yielding 10 distinct accuracy scores for each metric (see Table 1). Specifically, the *accuracy* metric exhibited a range of scores between 0.778237 and 0.881933, while the *f1_weighted* metric varied from 0.764104 to 0.874872. Similarly, the *roc_auc_ovr_weighted* metric demonstrated scores ranging from 0.984472 to 0.991242.

Table 1

**Accuracy Scores for Random Forest Classifier Across 10-Fold Cross-Validation**

| fold | accuracy | f1_weighted | roc_auc_ovr_weighted |
|---|---|---|---|
| 1 | 0.801216 | 0.764104 | 0.984472 |
| 2 | 0.806792 | 0.806311 | 0.988205 |
| 3 | 0.812796 | 0.834388 | 0.989931 |
| 4 | 0.835232 | 0.847424 | 0.991242 |
| 5 | 0.814671 | 0.874872 | 0.990672 |
| 6 | 0.825667 | 0.861554 | 0.989349 |
| 7 | 0.788241 | 0.855245 | 0.990568 |
| 8 | 0.881933 | 0.838916 | 0.990249 |
| 9 | 0.878251 | 0.853193 | 0.987962 |
| 10 | 0.778237 | 0.829191 | 0.985563 |

After calculating the dispersion values of each metric for the Random Forest classifier, we obtained the following results:

- *accuracy*: 0.032864
- *f1_weighted*: 0.030109
- *roc_auc_ovr_weighted*: 0.002157

So, the modulus of the vector representing the dispersion values for the Random Forest classifier is approximately 0.044623.

As the next step, we subjected the Logistic Regression classifier to 10-fold cross-validation, resulting in 10 different accuracy scores for each metric (see Table 2). The *accuracy* scores ranged from 0.092347 to 0.164045, while the *f1_weighted* metric varied between 0.081154 and 0.162578. Additionally, the *roc_auc_ovr_weighted* metric exhibited scores ranging from 0.495709 to 0.514799.

Table 2

**Accuracy Scores for Logistic Regression Classifier Across 10-Fold Cross-Validation**

| fold | accuracy | f1_weighted | roc_auc_ovr_weighted |
|---|---|---|---|
| [redacted] | | | |
| 10 | 0.915919 | 0.884695 | 0.993147 |

The dispersion values for each metric for the Logistic Regression classifier are as follows:

- *accuracy*: 0.020832
- *f1_weighted*: 0.027229
- *roc_auc_ovr_weighted*: 0.005647

Therefore, the modulus of the vector representing the dispersion values for the Logistic Regression classifier is approximately 0.034746.

For the last step, we subjected the MLP classifier to 10-fold cross-validation, yielding 10 distinct accuracy scores for each metric (see Table 3). Across the accuracy metric, scores varied from 0.857523 to 0.936045, while f1_weighted metrics ranged between 0.874954 and 0.931239. Moreover, the roc_auc_ovr_weighted metric displayed values ranging from 0.993147 to 0.998397.

Table 3

**Accuracy Scores for MLP Classifier Across 10-Fold Cross-Validation**

| fold | accuracy | f1_weighted | roc_auc_ovr_weighted |
|---|---|---|---|
| 1 | 0.905532 | 0.892042 | 0.994217 |
| 2 | 0.922678 | 0.908305 | 0.997135 |
| 3 | 0.858876 | 0.931239 | 0.997782 |
| 4 | 0.910443 | 0.926796 | 0.998244 |
| 5 | 0.857523 | 0.915227 | 0.997832 |
| 6 | 0.936045 | 0.923173 | 0.998397 |
| 7 | 0.911327 | 0.906324 | 0.996982 |
| 8 | 0.928695 | 0.904882 | 0.993417 |
| 9 | 0.868945 | 0.874954 | 0.996857 |
| 10 | 0.915919 | 0.884695 | 0.993147 |

Following the computation of dispersion values for each metric of the MLP classifier, the obtained results are as follows:

- *accuracy*: 0.027555
- *f1_weighted*: 0.017501
- *roc_auc_ovr_weighted*: 0.001915

Consequently, the modulus of the vector representing the dispersion values for the MLP classifier is approximately 0.032699.

The Random Forest classifier exhibited a modulus of the vector of approximately 0.044623, indicating a moderate level of dispersion in its performance across different metrics. While the dispersion values were relatively higher compared to the other classifiers, they still fell within an acceptable range.

The Logistic Regression classifier displayed a modulus of the vector of approximately 0.034746, suggesting a slightly lower level of dispersion in its performance compared to the Random Forest classifier. The dispersion values for this classifier were generally lower across all metrics, indicating a more consistent performance.

The MLP classifier demonstrated the lowest modulus of the vector, approximately 0.032699, indicating the highest level of robustness among the three classifiers. The dispersion values for the MLP classifier were consistently lower across all metrics, signifying a highly consistent performance across different evaluations.

## Conclusions

After calculating the dispersion of each classifier, it is possible to highlight such conclusions as that the MLP classifier has the lowest dispersion, indicating the highest robustness, followed by the Logistic Regression classifier and the Random Forest classifier in last place, showing the highest dispersion in its tests. Therefore, if robustness is a critical factor in choosing a classifier, then the MLP classifier would be the better choice in this situation, followed by the Logistic Regression classifier. However, the specific requirements and limitations of a task should also be considered in making the final decision.

It should also be noted that the robustness of a classifier is particularly important for medical applications, especially for people with various types of limb defects, spinal cord injuries, and other neurological diseases. From this, it can be understood that the constant and reliable operation of brain-computer interfaces is important for improving the quality of their lives and reducing the manifestation of their defects.

## References

1. Musk E., Neuralink. An Integrated Brain-Machine Interface Platform With Thousands of Channels. Journal of medical Internet research. 2019. Vol. 21. No. 10. URL: https://www.jmir.org/2019/10/e16194 (accessed: 07.10.2023). https://doi.org/10.2196/16194.
2. Gilja, V., Pandarinath, C., Blabe, C. H., Nuyujukian, P., Simeral, J. D., Sarma, A. A., ... & Henderson, J. M. Clinical translation of a high-performance neural prosthesis. Nature Medicine. 2015. URL: https://pubmed.ncbi.nlm.nih.gov/26413781 (accessed: 23.12.2023).
3. Ajiboye, A. B., Willett, F. R., Young, D. R., Memberg, W. D., Murphy, B. A., Miller, J. P., ... & Walter, B. L. Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration. 2017. URL: https://pubmed.ncbi.nlm.nih.gov/28363483 (accessed: 24.10.2023).
4. Ajiboye, A. B., Willett, F. R., Young, D. R., Memberg, W. D., Murphy, B. A., Miller, J. P., ... & Walter, B. L. Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration. 2017. URL: https://pubmed.ncbi.nlm.nih.gov/28363483 (accessed: 03.11.2023).
5. Pastukh. O., Yatsyshyn V. Development of software for neuromarketing based on artificial intelligence and data science using high-performance computing and parallel programming technologies. *Scientific Journal of TNTU*. 2024. № 113. C. 143–149.
6. Pastukh. O., Yatsyshyn V. Brain-computer interaction neurointerface based on artificial intelligence and its parallel programming using high-performance calculation on cluster mobile devices. *Scientific Journal of TNTU*. 2023. № 112. C. 26–31.
7. Pastukh O., Stefanyshyn V., Baran I., Yakymenko I., Vasylkiv V. Mathematics and software for controlling mobile software devices based on brain activity signals. *The International Workshop on Information Technologies: Theoretical and Applied Problems (ITTAP-2023)*: Proceedings of the International Scientific and Practical Conference, Ternopil, November 22-24, 2023. pp. 684 - 689.
8. Medic XAI. URL: https://xai-medica.com/en/equipments.html (accessed: 24.01.2024).
9. Scikit-Learn & Joblib. URL: https://ml.dask.org/joblib.html (accessed: 01.02.2024).
10. Bright brain – London's eeg, neurofeedback and brain stimylation centre. URL: https://www.brightbraincentre.co.uk/electroencephalogram-eeg-brainwaves (accessed: 01.02.2024).
11. Zuo C., Jin J., Yin E., Saab R., Miao Y., Wang X., et al. Novel hybrid brain-computer interface system based on motor imagery and p300. Cogn. Neurodyn. 2020. Vol. 15. No. 2. URL: https://pubmed. ncbi.nlm.nih.gov/32226566 (accessed: 13.11.2023). https://doi.org/10.1007/s11571-019-09560-x.



| | | |
|---|---|---|
| **Oleksandr BRYK**<br>**Олександр БРИК** | Postgraduate Student at the Department of Software Engineering, Ternopil Ivan Puluj National Technical University, Ternopil, Ukraine.<br>e-mail: alex.lenberg@gmail.com.<br>https://orcid.org/0009-0005-6564-1102 | аспірант кафедри програмної інженерії, Тернопільський національний політехнічний університет імені Івана Пулюя, Тернопіль, Україна. |

| | | |
|---|---|---|
| **Ivan STEFANYSHYN**<br>**Іван СТЕФАНИШИН** | Postgraduate Student at the Department of Software Engineering, Ternopil Ivan Puluj National Technical University, Ternopil, Ukraine.<br>e-mail: ivan_stefanyshyn0707@tntu.edu.ua.<br>https://orcid.org/0009-0008-6930-528X. | аспірант кафедри програмної інженерії, Тернопільський національний політехнічний університет імені Івана Пулюя, Тернопіль, Україна. |
| **Volodymyr STEFANYSHYN**<br>**Володимир СТЕФАНИШИН** | Postgraduate Student at the Department of Software Engineering, Ternopil Ivan Puluj National Technical University, Ternopil, Ukraine.<br>e-mail:<br>volodymyr_stefanyshyn3006@tntu.edu.ua.<br>https://orcid.org/0009-0007-2829-8995.<br>SCOPUS: 58614423400 | аспірант кафедри програмної інженерії, Тернопільський національний політехнічний університет імені Івана Пулюя, Тернопіль, Україна. |
| **Oleh PASTUKH**<br>**Олег ПАСТУХ** | Dr. Sc. Prof., Professor at the Department of Software Engineering, Ternopil Ivan Puluj National Technical University, Ternopil, Ukraine.<br>e-mail: ol_pas@tntu.edu.ua.<br>https://orcid.org/0000-0002-0080-7053.<br>SCOPUS: 57201343382 | докт. техн. наук, проф., професор кафедри програмної інженерії, Тернопільський національний політехнічний університет імені Івана Пулюя, Тернопіль, Україна. |