VOLUME 1 NUMBER 4 2023 ISSN 2958-0846
eISSN 2958-0854

AL-FARABI KAZAKH NATIONAL UNIVERSITY

Journal of Problems in Computer Science and Information Technologies

№4 (1) 2023

Al-Farabi Kazakh National University

ISSN: 2958-0846; eISSN: 2958-0854

Journal of Problems in Computer Science and Information Technologies №4 (1) 2023

03.02.2023 Registered with the Ministry of Information and Social Development of the Republic of Kazakhstan

№ KZ61VPY00064018

The journal is published 4 times a year
(March, June, September, December).

Signed to publishing 27.12.2023. Format 60x84/8. Offset paper.
Digital printing. Volume 3,6 printer's sheet.
Publishing house «Kazakh University»
www.read.kz Telephone: +7 (727) 3773330, fax: +7 (727) 3773344
Al-Farabi Kazakh National University KazNU, 71 Al-Farabi, 050040, Almaty

IRSTI 50.43.31 https://doi.org/10.26577/jpcsit2023v1i4a1

M.A. Soomro[1*], A.R. Nangraj[1], H. Javed[2]

[1]University of Sindh, Jamshoro, Pakistan
[2]Central South University, Changsha, China
*e-mail: mudasirasimsoomro@gmail.com

INTELLIGENT MONITORING AND ALERT SYSTEM FOR GIRLS' HOSTELS

Abstract. The issue of security is a critical concern in today's world and ensuring the safety of individuals is a top priority. Girls' hostels face unique security challenges that require innovative solutions to address. The purpose of the research is to develop an innovative security system that will provide an enhanced level of protection to Girl's hostel residents while also streamlining the hostel's management operations. This study proposes an Intelligence Security System for Girls Hostels that combines the latest advancements in security technology with a user-friendly interface to provide enhanced protection for the residents. The proposed system utilizes fingerprint authentication as a password, allowing residents to enter and exit the hostel with ease and security. Upon entry, a notification is sent to the parents' registered cell phone number, providing them with peace of mind. The system also includes a comprehensive database that stores information about the students, their enrollment, and payments, enabling the efficient management of the hostel's operations.

The design of the system is based on the identification of the weaknesses of traditional security systems, incorporating user-friendly and Graphical User Interface-oriented features to make it more accessible to users. The computerization of the hostel's transactions and data storage will increase efficiency and provide valuable insights and reports that can help with decision-making. Overall, the Intelligence Security System for Girls Hostels aims to create a safe and secure environment for residents while also making the hostel's management tasks easier and more efficient.

Key words: Intelligence Security System, Fingerprint Authentication, Parental Notifications, Global System for Mobile Communications, Global Positioning System, Innovative Security System, Graphical User Interface-Oriented Features.

1 Introduction

Ensuring safety, especially in girls' hostels, is a critical concern in today's world. Girls' hostels face unique security challenges that need innovative solutions [1]. The Intelligence Security System for Girls Hostels is an innovative response to the growing need for enhanced security measures in these facilities [2]. The Intelligence Security System for Girls Hostels uses advanced security technology, including fingerprint authentication, for secure and convenient access [3]. It also features a comprehensive database that stores student information, enrollment, and payments, facilitating efficient hostel management. The Intelligence Security System for Girls Hostels aims to provide a safe environment for residents while enhancing hostel management. It leverages computerization for data storage and transaction processing, increasing efficiency and providing valuable insights for decision-making. It's an innovative, user-friendly solution addressing security concerns and operational efficiency in girls' hostels.

The safety and security of girls is a critical issue that involves protection from violence, abuse, exploitation, and discrimination. It encompasses ensuring physical safety, access to education and healthcare, and promoting gender equality. All stakeholders, including governments, communities, and individuals, have a role in this crucial task [4]. Hostel management has a responsibility to ensure the safety of girls. This includes implementing security measures like proper locks and systems, regular supervision, and staff training on issues like sexual harassment. Clear policies should be in place to handle incidents and provide support to affected residents. Cooperation with local authorities and community organizations is also crucial to create a safe environment. The "Intelligent Monitoring and Alert System for Girls' Hostels" is a technology-driven solution designed to streamline hostel management. It covers various operations like admission, fee payment, and report generation. Developed using languages like HyperText Markup Language, Cascading Style Sheets, Bootstrap, JavaScript, and Hypertext Preprocessor, and a Database Management System like Microsoft SQL

Server, it offers a user-friendly interface for easy record management. The software ensures security with authorized access and allows students to generate challans during hostel allotment.

Fingerprint authentication is a crucial feature of the Intelligence Security System for Girls Hostels. It uses unique fingerprints to verify identities, providing secure access control in girls' hostels. The system automatically monitors student movement, enhancing security. It also offers a user-friendly interface for record management and authorized access. The use of fingerprint authentication mitigates the risk of lost or stolen keys, thereby boosting hostel security [5].

The main objective of the study was to develop an innovative security system that will provide an enhanced level of protection to Girl's hostel residents.

The literature review of all papers highlights the importance and advancement of intelligent security systems in various fields. From residential environments to educational institutions, public transport, health care and even smart grids, these articles show the increasing demand for innovative and secure systems. The authors provide a comprehensive overview of the current state of smart security systems and discuss their strengths, weaknesses, and potential for future development.

The authors also emphasize the need for effective security systems that can protect individuals, property, and data from potential threats. This is particularly important in the context of women's safety and health care, where the safety and privacy of patients and their data is of utmost importance. The authors highlight the need for smart security systems that are reliable, user-friendly, and cost-effective.

This assortment of scholarly articles provides a comprehensive overview of intelligent security systems across a variety of sectors. Ibrahim [6] offers an in-depth look at intelligent video surveillance systems, discussing their design, implementation, and evaluation. Jadon [7] explores smart home security systems, their features, applications, and strategies for implementation. Bhati et al. [8] in their study reviews security systems for smart campuses, while Al-Ahmadi et al. [9] discusses various smart security systems, including their features and implementation strategies. In their study, Yerragolla et al. [10] conducted a research on intelligent security systems in residential settings. In addition, the study of Baig et al. [11] provides a comprehensive overview of smart security systems, including those that utilize IoT technology, biometrics, and artificial intelligence, and discusses their benefits, limitations, and potential areas for future research. Moreover, Singh et al. [12] in their study offers an overview of various intelligent surveillance systems currently available, discussing their applications, benefits, and limitations, and suggesting directions for future research. Furthermore, the study of Priya et al. [13] reviews smart security systems designed specifically for women's safety. Al- Badshah et al. [14] delves into the topic of smart security systems designed specifically for educational institutions. It provides a comprehensive review of these systems, discussing their applications, benefits, and limitations, and also suggests potential directions for future research. On the other hand, Lu et al. [15] reviews smart security systems for healthcare, while the study of Sharma et al. [16] discusses smart security systems designed for smart grids. Finally, Paradkar et al. [17] conducts a systematic review of the literature on Intelligent Safety System for Women Security. The focus is on the design and implementation of these systems, offering valuable insights into the practical aspects of women security.

2 Design and Development

The methodology chosen for this research is Object Oriented Methodology (OOP). This approach recognizes that objects play a significant role in our lives, be it in nature, human-made entities, business, or everyday products. OOP aims to bring this perspective to the creation of computer software. It was first proposed in the late 1960s as a new way of developing software. This methodology requires the analyst to identify the objects in the system, understand their behavior and interactions, and determine their responsibilities and relationships.

The object-oriented analysis focuses on the examination of all objects in the system, their similarities, differences, and the manipulation needed to develop the system. The process of OOP begins with analyzing and observing the system to be developed, defining the requirements, and identifying the objects within the system such as students, administrators, computer systems, and online allocation systems. In essence, OOM involves identifying objects in a system and their interconnections, which serves as the basis for implementing the system.

The system environment for the Intelligence Security System for Girls Hostels is designed to ensure optimal performance and user experience. It is configured with both hardware and software components that are tailored to meet the system's requirements and ensure its smooth operation. The following table 1 and table 2 describes hardware and software components of system environment.

Table 1 – Hardware Configuration

Component	Specification
Processor	Pentium
RAM	4GB
Hard Disk	40GB
Monitor	15" Color Monitor
Mouse	Yes
Keyboard	Yes

Table 2 – Software Configuration

Component	Specification
Operating System	Windows7
Languages	HTML, CSS, Bootstrap, JavaScript, and PHP
Database	MySQL

The system will consist of a central database that will store all the information regarding the students and the hostel. The system will also have an authentication mechanism which will be used to verify the identity of the user. The authentication mechanism in this case will be the fingerprint recognition system. The system will also have a notification system that will notify the parents of the students whenever they enter or exit the hostel.

The flow of the system will be as follows: the user logs into the web application using their credentials and verifies their identity via a fingerprint scan. Upon successful verification, access to the hostel is granted and entry/exit times are logged. Parents receive notifications on their registered mobile number each time the user enters or exits the hostel. The application includes sections for student management and payment management, and a reporting section that generates various reports. The user logs out to ensure information security.

When a student enters or exits the hostel, they authenticate their identity via a fingerprint scanner. Upon successful verification, the system sends a notification to the parents about the student's arrival or departure. This process ensures secure and monitored access to the hostel.

The admin is key in setting up and managing the Girls Hostel security system. They handle the initial setup, student enrollment, and database updates. They also add parents' contact details for notifications. The admin sets up the fingerprint system at the hostel's access points and manages all information in the database, including generating reports. They maintain the system, ensure its functionality, and have the authority to modify student information as needed.

The "Intelligence Security System for Girls Hostel" is designed based on a client-server architecture, with the student as the client and a central database as the server. The server handles authentication and notifications. The system is implemented using programming languages like HTML, CSS, Bootstrap, JavaScript, PHP, and a database management system like MySQL. It incorporates devices such as a fingerprint scanner and a computer system for database management.

In the security system depicted in Figure 1, several components work together to create a secure environment. The core of the system is the microcontroller unit, which controls all operations. Secure access is ensured by the fingerprint scanner, while communication with the warden's PC is handled by the GSM module. Information is displayed and stored by the LCD and memory card, respectively. The keyboard facilitates data input, and time logging is done by the Real-Time Clock.

The process a student undergoes when entering or exiting a hostel, as showcased in Figure 2, involves a fingerprint scan for identity verification, followed by a parental notification. This systematic approach ensures a secure and transparent hostel management system.

As depicted in Figure 3, a flowchart of the login and access process for a girls' hostel security system web application. It begins with user login, followed by fingerprint authentication. Post-authentication, the system verifies hostel access and notifies parents. The application manages student information, transactions, and generates reports. The process ends with user logout, ensuring system and data security.

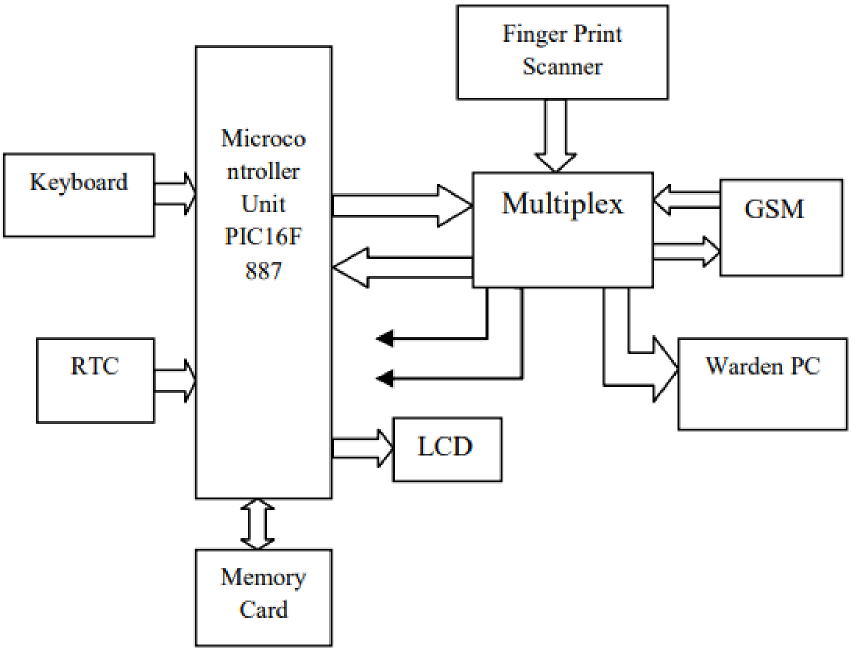

Figure 1 – Block diagram Fingerprint Security Module System [1]

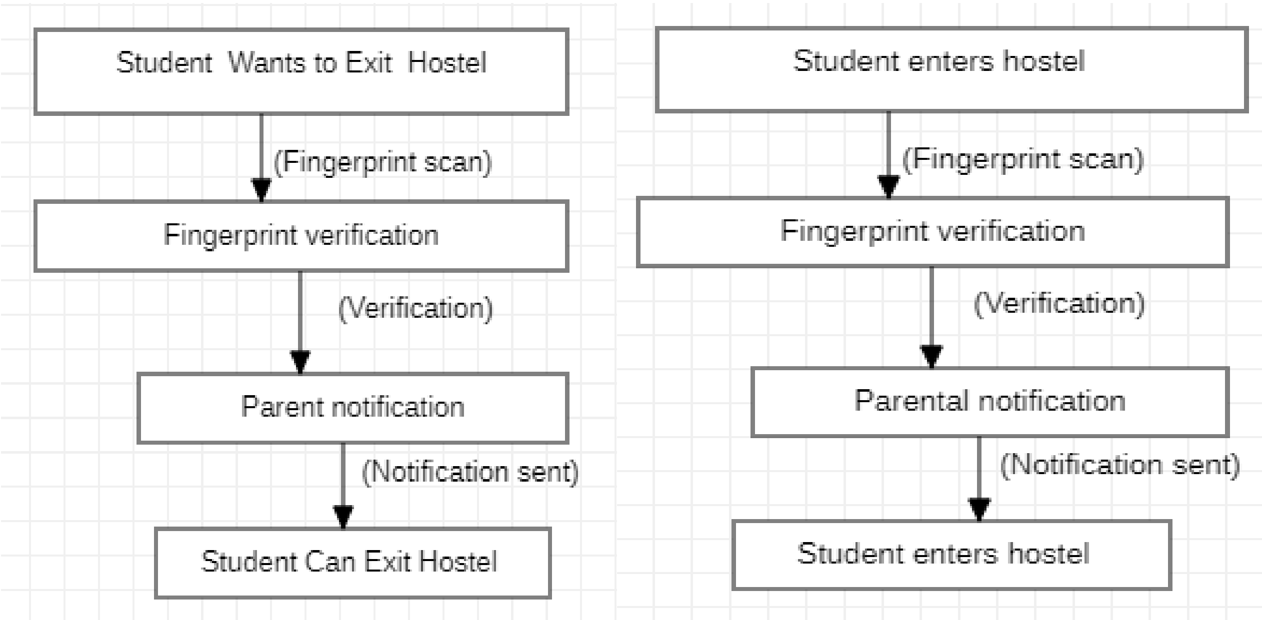

Figure 2 – Flow diagram Fingerprint System

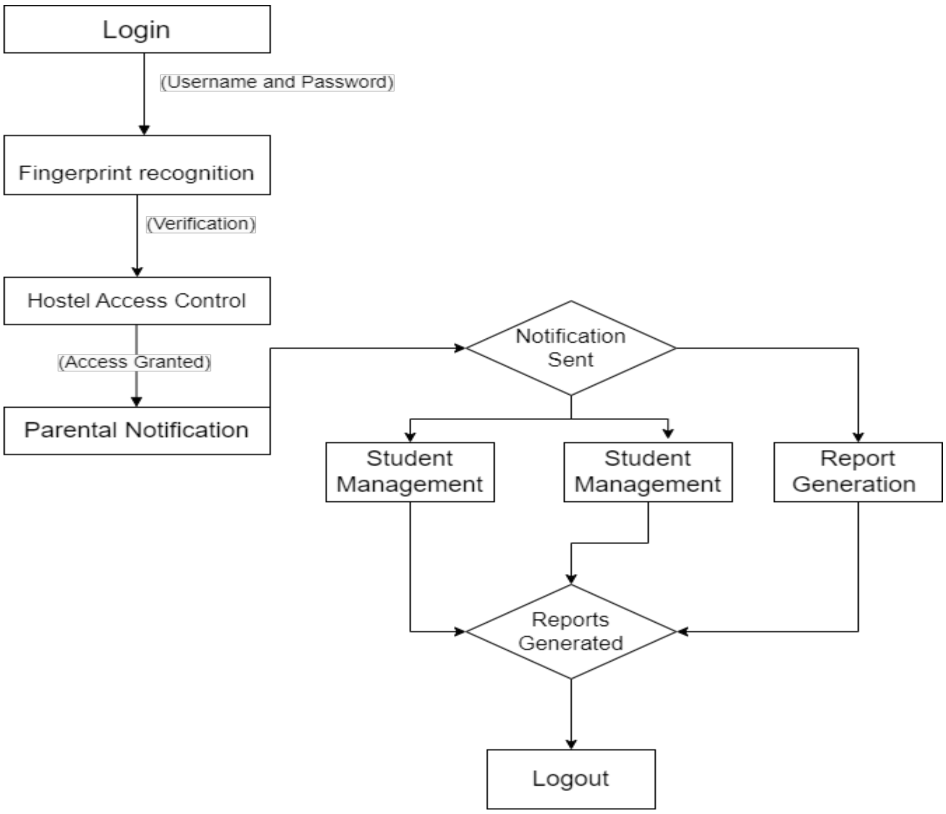

Figure 3 – Web app of the girl security system

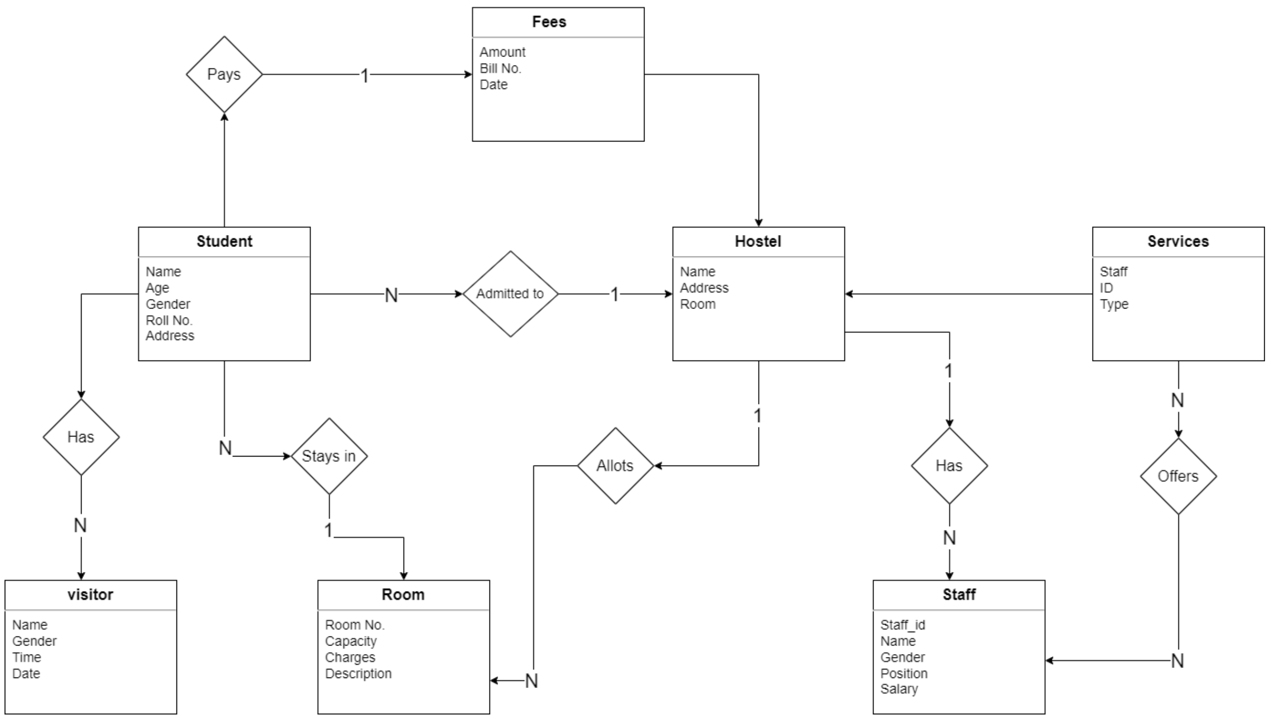

Figure 4 – UML-Diagram of Hostel Monitoring System

The Unified Modelling Language Diagram (UML-Diagram) in Figure 4 provides a graphical tool typically used in computing to structure data within databases or information systems. This specific ER-Diagram features six entities: STUDENT, HOSPITAL, SERVICES, ROOM, STAFF, and VISITOR, all interconnected to illustrate their relationships. For example, the STUDENT is "Admitted to" the HOSPITAL, which in turn "Offers" SERVICES. These SERVICES are associated with a "Type" of ROOM, which is linked to a "Staff Id" from STAFF. This diagram serves as a clear, structured overview of the system's interactions, proving invaluable for understanding and designing database systems.

As outlined in Figure 5, the Hostel Management System orchestrates the entire process, with Students applying for accommodation and the admin handling hostel setups.

3 System Architecture

Intelligence security systems are becoming a crucial component in ensuring the safety and privacy of residents in girls' hostels. The architecture of such a system is typically organized into four key layers as you can see in the figure 6.

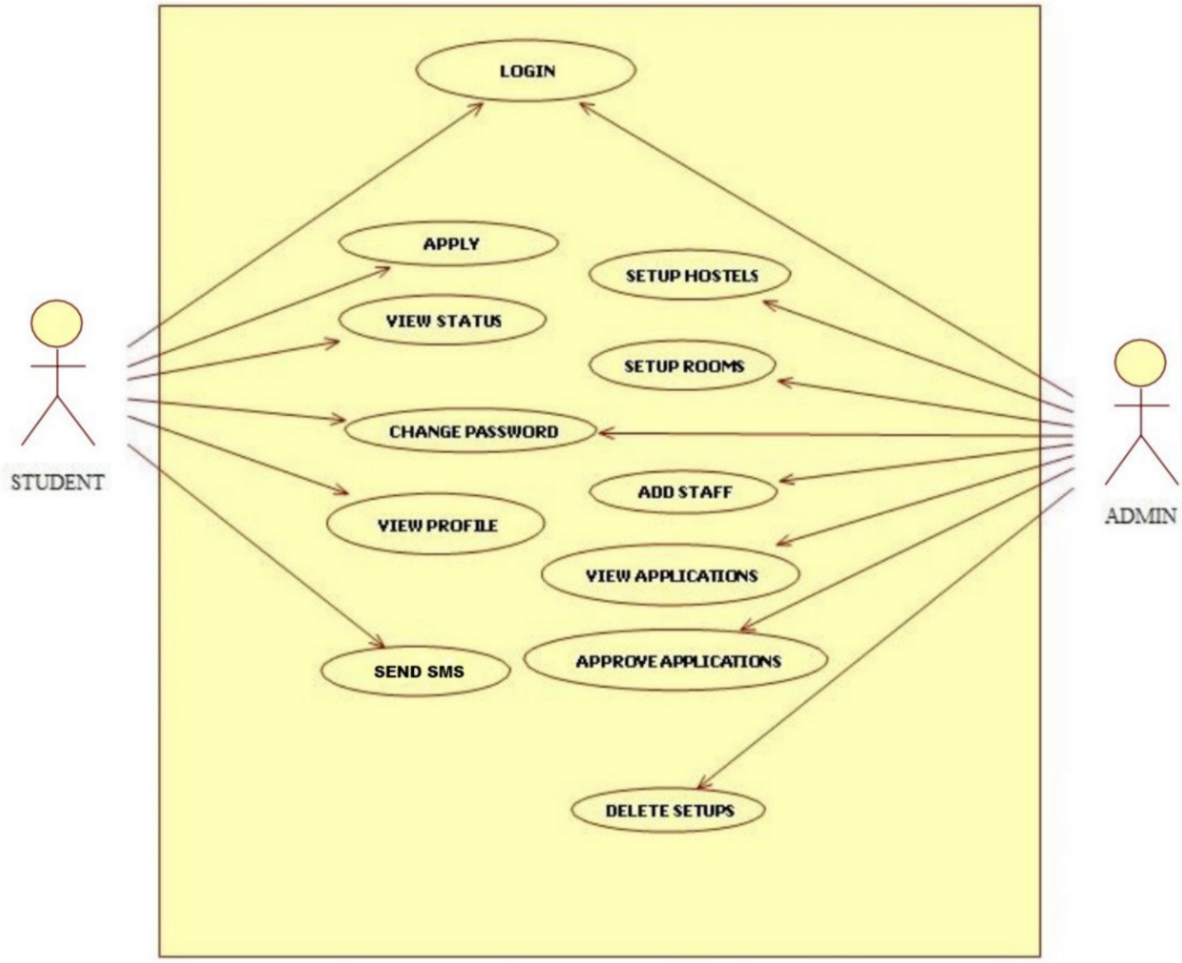

Figure 5 – Use Case Model of Hostel Management System

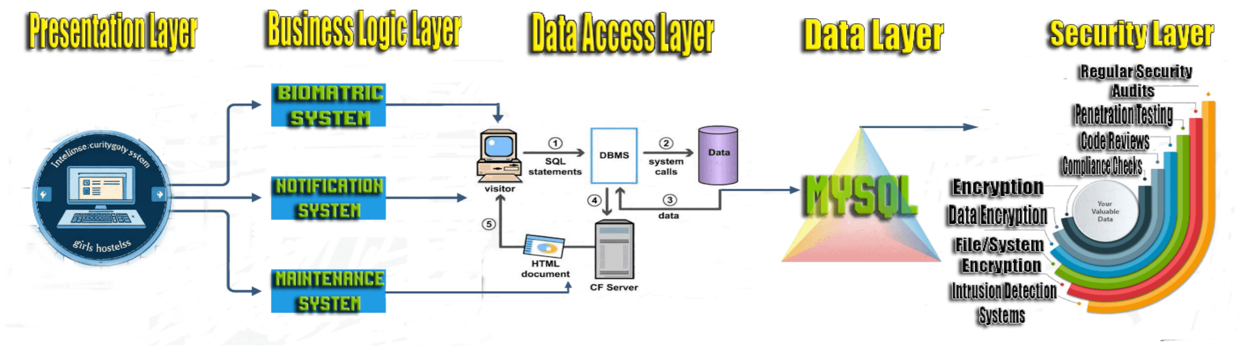

Figure 6 – System Architecture of Integrated Intelligence Security Solutions

Each layer has a specific role, and they work together to provide a secure and efficient biometric system. The architecture is designed in such a way that each layer can function independently, yet they are interconnected to ensure smooth operation of the system. This layered approach also makes the system more manageable and scalable. As depicted in Figure 6, the architecture of a biometric system is divided into four layers. Please refer to it for further details.

- Presentation Layer: This layer consists of the biometric system, notification system, and monitoring system. These components interact with the user and present the information in a user-friendly manner.

- Business Logic Layer: This layer contains an SQL database and a visitor system. It handles the processing of data and implements the core functionality of the system.

- Data Access Layer: This layer is responsible for system calls and data management. It provides an interface for the business logic layer to access and manipulate data.

- Security Layer: This layer includes a MySQL database, a compliance check, and a security check. It ensures the security and integrity of the system by implementing various security measures and compliance checks.

The methodology of developing an intelligent security system for a girls' hostel involves several steps. First, requirements are gathered through interviews, surveys, and focus groups to identify the specific needs and security concerns of the hostel. This is followed by a detailed system analysis to evaluate the current security system and identify areas for improvement. Based on the gathered requirements and system analysis, a comprehensive solution design is created to address the security needs of the hostel. A prototype of the proposed system is then developed and tested for functionality and performance. Once the prototype is refined based on feedback, the final version of the security system is implemented in the hostel, which includes the installation of hardware and software, user training, and system testing. Regular maintenance and support are provided to ensure the system's effectiveness, and continuous evaluation is conducted to make necessary improvements to meet the evolving security needs of the hostel.

4 Result of the implementation

This study underscores the collective responsibility in safeguarding girls in hostels. The "Intelligence Security System for Girls Hostel" plays a pivotal role in this context, offering heightened security through cutting-edge technology and an intuitive interface. The system leverages fingerprint authentication, offering a secure and user-friendly way for residents to access the hostel. It also keeps parents informed about their child's movements by sending notifications each time their child enters or exits the hostel. In addition, the system incorporates a comprehensive database for effective hostel management. It maintains records of students, their enrollment, and payments, thereby optimizing operations and enhancing efficiency. The system's design addresses the shortcomings of traditional security systems by integrating user-friendly and GUI-oriented features. The digitization of the hostel's transactions and data storage not only boosts efficiency but also yields valuable insights and reports that can assist in decision-making. In summary, the results indicate that the Intelligence Security System for Girls Hostels successfully fosters a safe and secure environment for residents while also simplifying hostel management tasks. However, it's crucial to remember that this system is part of a broader security framework, and additional measures should be implemented to ensure the overall safety and well-being of the residents.

Testing is indeed a crucial part of the development process for any system. It involves a series of steps designed to identify errors, verify objectives, and ensure user requirements are met. In our case, the testing of the "Intelligence Security System for Girls Hostels" involved several stages to ensure its effectiveness and reliability. Let's delve into each of these testing methods:

As shown in Figure 7, we conducted four levels of testing in the development process. Unit Testing tested individual modules independently, such as the fingerprint authentication module and the notification system. Following this, Integration Testing then checked the interaction between these modules. Next, System Testing simulated operations like user login and fingerprint verification on the integrated system. Finally, User Acceptance Testing involved real-world testing by the intended audience, collecting feedback, and making necessary adjustments for deployment. This process ensured the system's usability, performance, and reliability.

In each of these stages, we were actively involved in designing the tests, executing them, and analyzing the results. This rigorous testing process helped us ensure that the "Intelligence Security System for Girls Hostels" is reliable, secure, and user-friendly.

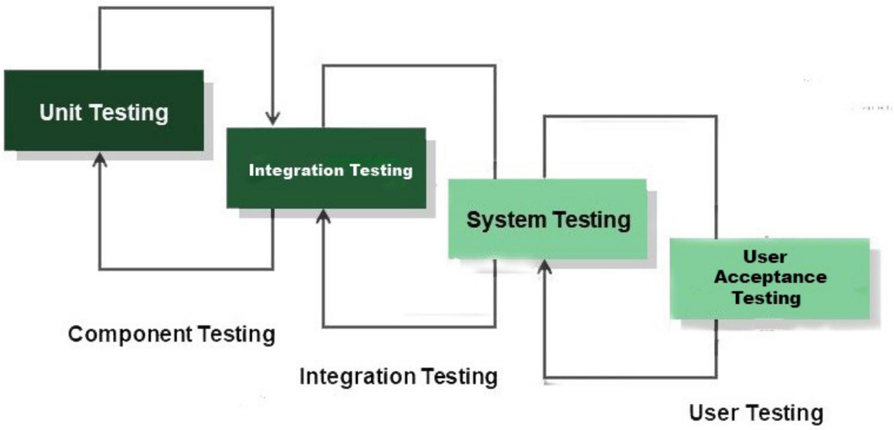

Figure 7 – Testing in the development process

Table 3 – Component of the system

Components	Features	Responsibilities
Web Application	Login, Fingerprint Recognition, Hostel Access Control, Parental Notification, Student Management, Payment Management, Reports Generation, and Logout.	Provide access to the hostel based on fingerprint verification, store and manage student information, handle payment transaction, send notification to parents, generate reports, and ensure information security
Fingerprint System	Fingerprint Verification at Entry and Exit, and Parental Notification	Verify student identity and notify parents of student's entry and exit
Admin	Setup and Configuration, Enrolling Students, Adding Parent's Information, Setting up Fingerprint Recognition, and Maintenance	Setup and manage the security system, enroll students, update student and parent information, and maintain the system and ensure its proper functioning

As you can see in Table 3, a hostel security system is composed of three main components. The first is the Web Application, which handles various functions such as login, fingerprint recognition, and report generation. The second is the Fingerprint System, responsible for verifying student identities and notifying parents about their movements. Lastly, the Admin oversees system setup, student enrollment, and system maintenance.

Table 4 – Features and the benefit for user

Feature	Benefit
Login	Provides secure access to the system by requiring a username and password.
Fingerprint Recognition	Enhance security by verifying the identity of the user based on their unique biometric information.
Hostel Access Control	Limits access to the hostel to only authorized users, and logs entry and exit times for tracking purposes.
Parental Notification	Keeps parents informed for their daughter's entry and exit from the hostel, providing added peace of mind.
Student Management	Enables the hostel to store, update, and retrieve information about students residing in the hostel.
Payment Management	Handles payment of the hostel fees and other dues by students, providing a secure and efficient way to manage financial transcations.
Reports Generation	Generates reports based on various criteria such as hostel fees, payment status, etc., allowig for easier and analysia od data.
Logout	Secures information stored in the system from unauthorized access by requiring users to log out.

The system, as detailed in Table 4, offers several beneficial features. It provides secure access through a login feature and enhances security with fingerprint recognition. It also restricts hostel access to authorized users, keeps parents informed about their child's activities, manages student information and financial transactions efficiently, and includes a report generation feature for easy data analysis. Additionally, it has a logout feature to prevent unauthorized access.

By implementing these features, this paper provides a comprehensive and secure solution for managing the security of girls in the hostel. However, it is important to remember that security is a complex and ongoing process that requires the cooperation and engagement of all stakeholders. The use of technology is just one part of a larger effort to ensure the safety and well-being of girls in hostels.

5 Conclusion

In conclusion, the development of the security system has been a valuable learning experience, providing insight into security, PHP, SMS notifications, file transfer protocol, and team collaboration. The hostel management and security software were designed to meet the requirements of the user and improve upon the existing system. The increasing number of educational institutes and hostels requires efficient management, which this software addresses through its user-friendly interface and compatibility with the existing system. The research has provided good experience in working with security systems and the team is eager to apply their knowledge in future security-based applications.

The current research has developed a security system for a girls' hostel that sends SMS notifications to guardians. However, there are potential enhancements that could further improve the system's functionality and efficiency. One such improvement could be the addition of a multiple notification system, which would extend the reach of alerts to other guardians and the hostel provost. Another potential enhancement is the implementation of a recognition system. This system would verify whether the guardian has communicated with their child after receiving the SMS. While these enhancements would require additional development, they could significantly improve the robustness of the security solution for girls' hostels.

References

1. Sapkal, A., Katait, S. U., Ingole, P. S., Dumre, N., & Ughade, P. (2015). Intelligent Security System For Girls In Hostel. International Journal of Advance Research in Science and Engineering, IJARSE, Vol. No.4, Special Issue (01), March 2015 (in Indian). http://www.ijarse.com/images/fullpdf/1426236698_JNU_663.pdf
2. Jain, S. S., Singal, G., Garg, D., & Gupta, S. K. (2019). SecureDorm: Sensor-Based Girls Hostel Surveillance System. In Smart Systems and IoT: Innovations in Computing (pp. 327–337). Springer Singapore. https://doi.org/10.1007/978-981-13-8406-6_32

3. Shoewu, O. O., Akinyemi, L. A., Mumuni, Q. A., Ajasa, A. A., Folorunso, C. O., & Yakubu, R. (2018). Development of a Fingerprint-Inspired Hostel Access Control. ABUAD Journal of Engineering Research and Development (AJERD), Volume 5, Issue 2, pp. 11-18. https://www.researchgate.net/publication/366734078_Development_of_a_Fingerprint-Inspired_Hostel_Access_Control

4. Bobhate, A., Chutke, D., Udan, M., Pal, K., & Hulke, P. (n.d.). Hostel Girls' Security System International Journal of Advance Research, Ideas and Innovations in Technology (Volume 4, Issue 2, pp. 315-219), (IJARIIT), (in Indian). https://www.ijariit.com/manuscripts/v4i2/V4I2-1247.pdf

5. RAJKUMAR, G., & SUNDARI, T. S. (2018). Hostel Management System Based on Finger Print Authentication. In Oriental journal of computer science and technology (Vol. 11, Issue 4, pp. 230–234). Oriental Scientific Publishing Company. https://doi.org/10.13005/ojcst11.04.08

6. Warsono Ibrahim, S. (2016). A comprehensive review on intelligent surveillance systems. In Communications in Science and Technology (Vol. 1, Issue 1). Komunitas Ilmuwan dan Profesional Muslim Indonesia (KIPMI). https://doi.org/10.21924/cst.1.1.2016.7

7. Pooja Jadon (2021). Security in Smart Home Development: A Review in International Journal of Innovative Research in Computer Science & Technology Volume-9, Issue-4), (IJIRCST), (in Indian). https://ijircst.org/DOC/13-ugc_21_3516_v3.pdf

8. BHATI, K. D., JAIN, V., & BHAT, N. K. (2017). Smart Campus Security: A Review by in Journal of Network and Computer Applications (in Indian).

9. AL-AHMADI, M. H., & AL-RASHDAN, S. (2016). Smart Security Systems: A Review by in International Journal of Distributed Sensor Networks.

10. Yerragolla, M., Pallela, K., & Gera, I. P. (2016). Intelligent security system for residential and industrial automation. In 2016 IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics Engineering (UPCON). 2016 IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics Engineering (UPCON). IEEE. https://doi.org/10.1109/upcon.2016.7894657

11. BAIG, M. A., ALI, S. Z., & KHAN, S. M. (2017). An Overview of Smart Security Systems in Journal of King Saud University – Computer and Information Sciences.

12. SINGH, S. K., & SINGH, M. K. (2017). Intelligent Surveillance Systems: A Review in Journal of Advanced Research in Dynamical and Control Systems (in Indian).

13. E. Anu Priya, A.Alsameema, M.R. Elakhia, B. Jaya Meera, Prof. Dr. S. Maheswari (2020). Smart Safety System for Women Security. INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 9, ISSUE 03), (IJSTR), (in Indian). https://www.ijstr.org/final-print/mar2020/Smart-Safety-System-For-Women-Security.pdf

14. Badshah, A., Ghani, A., Ahsan Qureshi, M., & Shamshirband, S. (2019). Smart Security Framework for Educational Institutions using Internet of Things (IoT). In Computers, Materials & Continua (Vol. 61, Issue 1, pp. 81–101). Computers, Materials and Continua (Tech Science Press). https://doi.org/10.32604/cmc.2019.06288

15. Lu, Y., & Sinnott, R. O. (2020). Security and privacy solutions for smart healthcare systems. In Innovation in Health Informatics (pp. 189–216). Elsevier. https://doi.org/10.1016/b978-0-12-819043-2.00008-3

16. Sharma, Jeetu & Bhattacharya, Partha & Jain, Vinod. (2018). A Review on Security in Smart Grids. International Journal of Allied Practice, Research and Review. 5. 92-96. https://www.researchgate.net/publication/324106126_A_Review_on_Security_in_Smart_Grids

17. Paradkar, A., & Sharma, D. (2015). All in one Intelligent Safety System for Women Security. In International Journal of Computer Applications (Vol. 130, Issue 11, pp. 33–40). Foundation of Computer Science. https://doi.org/10.5120/ijca2015907144

IRSTI 28.23.01 https://doi.org/10.26577/jpcsit2023v1i4a2

N.M. Kassymbek[1*], A.S. Rakhimzhanova[2]

[1]U.A. Joldasbekov Institute of Mechanics and Engineering, Almaty, Kazakhstan
[2]Astana IT University, Astana, Kazakhstan
*e-mail: nuryslam.qassymbek@gmail.com

COMPARATIVE STUDY OF PARALLEL ALGORITHMS FOR MACHINE LEARNING METHODS

Abstract. In the modern world, as the amount of data used in machine learning is constantly growing, the task of accelerating the training of models on large datasets becomes relevant. To solve this problem, methods of parallel data processing are used. This paper discusses methods of parallel data processing for machine learning. Linear regression and random forest are considered as machine learning methods. Parallel algorithms based on the MPI interface were developed for each method. The results of the experiments showed that both methods give acceleration compared to the sequential algorithm. However, the acceleration in the case of random forest was significantly higher than in the case of linear regression. This is because random forest is a more computationally efficient method than linear regression. Therefore, it can be concluded that Random Forest is the most effective machine learning approach for parallel data processing. This statement is confirmed by the results of experiments conducted in this work. Overall, the experimental results show that the use of parallel algorithms in machine learning can significantly speed up model training when working with large data sets. Random forest is the most efficient method for parallel data processing, as it is more computationally efficient and has higher scalability.

Key words: Machine Learning, Linear Regression, Random Forest, Parallel Computing.

1 Introduction

Machine learning (ML) is a branch of artificial intelligence dedicated to creating algorithms that can learn from data without the need for explicit programming [1,2]. These algorithms find applications in various fields such as image recognition, natural language processing, and predictive analysis.

One of the most widely used ML methods is linear regression [3]. Linear regression is employed to forecast continuous values using a dataset containing pairs of independent and dependent variables [4]. Traditional linear regression training operates sequentially, and for substantial datasets, it can be computationally intensive. This is attributed to the quadratic nature of the matrix operation involved in the normal function, utilized to calculate the weights for linear regression, which scales with the number of observations in the dataset.

The random forest serves as an ensemble machine learning method applied to both classification and regression tasks. It involves numerous decision trees trained on diverse subsets of the dataset [5,6]. Conventional random forest training is likewise a sequential procedure, and its computational demands can be high, particularly for sizable datasets. This is due to the necessity of computing numerous data splits for each decision tree.

Parallel programming is a software development methodology that allows multiple tasks to be run simultaneously on multiple processors or computing nodes [7,8]. Parallel computing can significantly speed up the execution of tasks that can be divided into multiple independent parts. The importance of parallel computing in ML is due to the growth in the size and complexity of datasets used in ML applications. Large datasets can require significant computational resources to train ML algorithms. Parallel computing can help to speed up the training process and make it more accessible [9].

Parallel learning in machine learning is an urgent task. There are various approaches to parallelization of machine learning methods, for example, in [10-12] the authors developed a parallel algorithm for the linear regression method. In [10,12], an implementation was given for multithreaded systems, and in [11] for systems with distributed memory. A similar problem, but for the method of support vectors for regression, can be seen in [13,14]. In these papers, the authors used the MPI (Message Passing Interface) standard for parallelization. And in the works [15,16] one can see the use of the same standard, but for the Random Forest method. These works show the relevance of the problem under consideration.

In this paper, we present a parallel linear regression training algorithm that uses MPI to parallelize the matrix operations of the normal function. Our algorithm can significantly speed up the training of linear regression for large datasets.

2 Linear regression

Linear regression stands out as one of the most extensively utilized machine learning methods, employed for predicting continuous values using a dataset that comprises pairs of independent and dependent variables. Linear regression has many applications in various fields, including:

- Demand forecasting
- Price forecasting
- Sports outcome forecasting
- Image classification
- Natural language processing

In traditional linear regression, the dependent variable y is assumed to be the linear sum of the independent variables $x_1, x_2, \dots, x_n$:

$$y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_n x_n.$$

The constants $\theta_0, \theta_1, \dots, \theta_n$ are called regression weights. They are calculated in such a way as to minimize the loss function, often formulated as the sum of the squares of the differences between the predicted and actual values of the dependent variable:

$$\sum_{i=1}^{n} (y_i - (\theta_0 + \theta_1 x_i + \theta_2 x_i^2 + \dots + \theta_n x_i^n))^2.$$

This loss function is called the normal function.

A solution to the problem of minimizing the normal function is offered by the least squares method. This method includes the following steps:

1. Calculate the matrix X, consisting of the observations of the independent variables.
2. Calculate the vector y, containing the actual values of the dependent variable.
3. Calculate the matrix W, the inverse of the matrix $X^T X$.
4. The regression weights are determined as follows:

$$\theta = W X^T y. \qquad (1)$$

3 Random forest

Random Forest serves as an ensemble learning method designed for both classification and regression purposes. It includes many decision trees, each trained on separate subsets of the data set.

Leo Breiman introduced the random forest method in 2001, and since then it has become one of the widely used machine learning approaches, finding applications in various fields such as image recognition, natural language processing and prediction.

The learning algorithm for a random forest comprises the following steps:

1. Initialize a set of decision trees T.
2. For each decision tree:
   - Compute a random subset of size n_s from the original dataset.
   - Train the decision tree on the subset.
3. Make a prediction by voting the decision trees.

On step 2, the random forest algorithm uses bagging. Bagging is an ensemble learning method that trains a multitude of models on different subsets of the original dataset. This helps to reduce overfitting, which can occur when training a single model on the entire dataset.

On step 3, the random forest algorithm uses voting. Voting is a method that combines the predictions of multiple models to obtain a more accurate predictive result.

4 Parallel architecture

Parallel algorithms of machine learning methods were developed as part of the program. There are various parallel architectures. Modern cluster and supercomputer systems are built on a hybrid architecture, i.e. the system consists of computing nodes with shared memory, and each node can have a multicore CPU and GPU as an accelerator. For multicomputer systems and systems with a hybrid architecture, the MPI standard is usually used in parallel programs. MPI (Message Passing Interface) is a programming interface (API) for message passing that allows processes to exchange messages in a parallel computing system. MPI is the most widely used standard for data exchange interfaces in parallel programming, and there are implementations for a large number of platforms. In this work, this standard was used for the parallel implementation of the methods under consideration.

There are two main approaches to implementing parallel algorithms: model parallelism and data parallelism. In the first approach, calculations are performed in parallel within the model. In the second approach, the data is divided into computing nodes, training takes place in parallel, and when forecasting, each node issues its own forecast, and the final decision can be made based on these forecasts by choosing the majority or calculating the average value.

A parallel linear regression algorithm has been developed. In this method, the weights can be found using gradient descent, and in some cases using the normal equation (1) by the analytical method. The equation consists of matrix multiplication, finding the inverse matrix and multiplying the matrix by a vector. These operations were parallelized using the MPI standard for shared memory systems. The Gauss-Jordan method was used to find the inverse matrix (W). The method itself is not well-parallelized, as the calculations have explicit dependencies. Therefore, a pipelined parallel algorithm was developed. To balance the computations, the matrix was distributed to the processes in a cyclic way. Matrix multiplication ($X^T X$ and WX^T) and matrix-vector multiplication (result with y) were implemented with row-wise distribution of the matrix.

The most convenient method for parallelization on cluster systems is the Random Forest method. The method builds N decision trees on random subsamples. When forecasting, the average value from each tree is found. With a parallel implementation, N trees are divided into NP parallel processes and each process builds its decision trees independently of each other. As shown in Figure 1, independent decision trees are constructed. To do this, the dataset is sent to all processes using the MPI_Bcast functions, and the mean value, the solution to the problem, is combined using the MPI_Reduce functions.

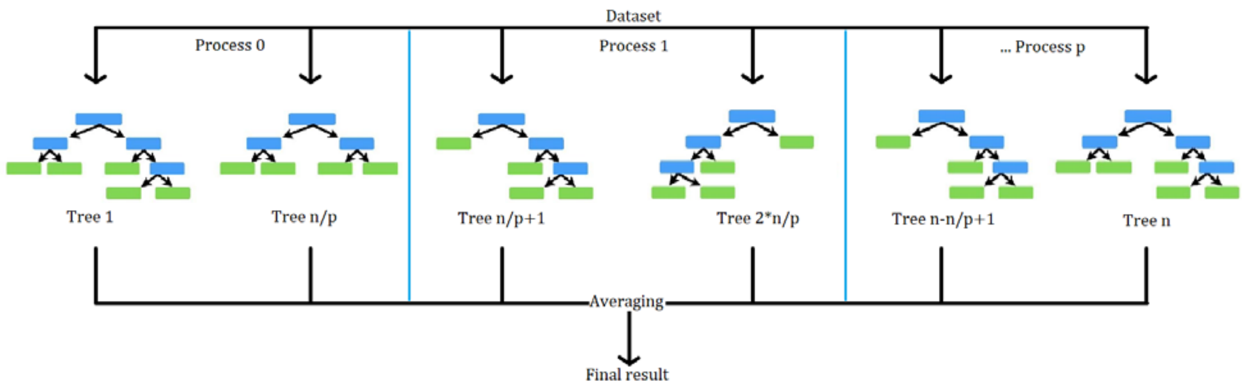

Figure 1 – Parallel Random Forest algorithm

5 Experimental results

Experiments were conducted on a dataset of the oil displacement problem. The characteristics of the dataset and the results obtained for sequential algorithms can be found in the work [17]. Tests of the developed parallel algorithms were conducted on a computer with an Intel® Core™ i7-10750H processor with 6 cores and 12 threads.

The results of parallelization of the Linear regression method can be seen in Table 1 and the results of parallelization of random forest can be seen in Table 2. The tests of the linear regression program were conducted on a dataset with a sample size of up to 2,400,000, and the tests of random forest up to 400,000. The random forest method, although more accurate, is computationally expensive. Therefore, the tests were conducted on a significantly smaller dataset size.

In all tables, the first column means the number of parallel processes, the first row means the sample size (the number of rows of the matrix X), and each subsequent row is the elapsed time of the parallel program's execution. Based on the information presented in the tables, you can see that both algorithms gave acceleration relative to the sequential program. The acceleration of programs can be seen in Figure 2.

Table 1 – Execution time of the parallel LR algorithm, sec

NP\m	600000	1200000	2400000
1	4,22	7,52	14,93
2	1,61	4,05	8,44
4	1,29	3,18	6,18
8	1,24	2,83	5,74

Table 2 – Execution time of the parallel RF algorithm, sec

NP\m	100000	200000	400000
1	2,25	4,77	11,3
2	1,2	2,5	5,46
4	0,68	1,36	2,96
8	0,59	1,02	1,91

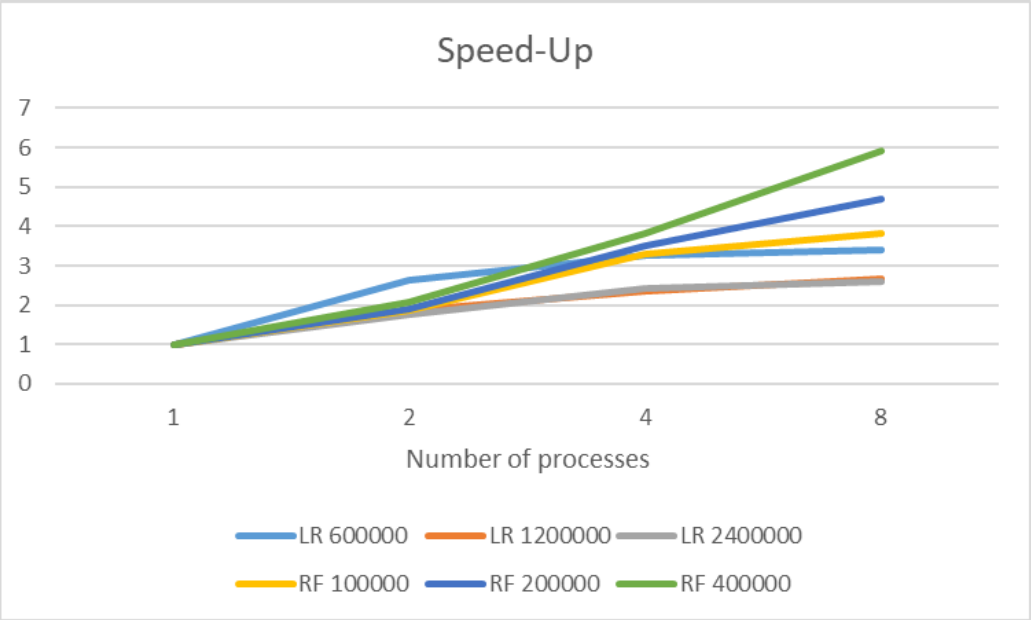

Figure 2 – Acceleration of LR and RF algorithms

From the graph, it can be seen that in linear regression, the acceleration decreases with increasing dataset size, while in random forest, the acceleration increases. Additionally, in random forest, a better acceleration was obtained in all cases. As mentioned earlier, random forest is the best machine learning method for parallelization. In linear regression, there is still a data dependency in the calculations, and the amount of communication is higher than in random forest. The increase in acceleration with increasing dataset size suggests that the parallel program could give even greater acceleration on even more data. This assumption requires experimental confirmation, which will be carried out in the future.

These experiments show that for a large dataset size on one processor, it is better to use the linear regression method, but if you have access to a system with multiple or more processors, in these cases it is better to use random forest.

6 Conclusion

In this work, methods of parallel data processing for machine learning were considered. Two of the most common machine learning methods were considered: linear regression and random forest. Parallel algorithms based on the MPI interface were developed for each method.

The experimental outcomes revealed that both methods give acceleration compared to the sequential algorithm. However, the acceleration in the case of random forest was significantly higher than in the case of linear regression. This is because random forest is a more computationally efficient method than linear regression. Hence, one can infer that random forest is the best machine learning method for parallel data processing. This conclusion is supported by the results of the experiments conducted in this work.

In addition, the experiments showed that the acceleration of linear regression decreases with increasing sample size, while the acceleration of random forest increases. This is because there is a data dependency in the calculations in linear regression, and the amount of communication is higher than in random forest.

Overall, the experimental findings indicate that employing parallel algorithms for machine learning can markedly expedite the training of models when dealing with extensive datasets. Random forest is the most efficient method for parallel data processing, as it is more computationally efficient and has higher scalability.

Acknowledgments

This research was funded by the Science Committee of the Ministry of Science and Higher Education of the Republic of Kazakhstan, grant number BR18574136.

References

1. Murphy, Kevin P.. *Machine learning : a probabilistic perspective*. Cambridge, Mass. [u.a.]: MIT Press, 2013.
2. Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York, 2009. https://doi.org/10.1007/978-0-387-84858-7.
3. Stock, James H., and Motohiro Yogo. "Testing for Weak Instruments in Linear IV Regression." *Identification and Inference for Econometric Models*. Cambridge University Press, June 17, (2005). https://doi.org/10.1017/cbo9780511614491.006.
4. Weisberg, Sanford. "Applied Linear Regression." *Wiley Series in Probability and Statistics*. Wiley, January 14, (2005). https://doi.org/10.1002/0471704091.
5. Tin Kam Ho. "Random Decision Forests." *Proceedings of 3rd International Conference on Document Analysis and Recognition*. IEEE Comput. Soc. Press, n.d. https://doi.org/10.1109/icdar.1995.598994.
6. Breiman, L. "Random Forests." *Machine Learning* 45, 5–32 (2001). https://doi.org/10.1023/A:1010933404324
7. Jin, Haoqiang, Dennis Jespersen, Piyush Mehrotra, Rupak Biswas, Lei Huang, and Barbara Chapman. "High Performance Computing Using MPI and OpenMP on Multi-Core Parallel Systems." *Parallel Computing*. Elsevier BV, September (2011). https://doi.org/10.1016/j.parco.2011.02.002.
8. Dalcin, Lisandro D., Rodrigo R. Paz, Pablo A. Kler, and Alejandro Cosimo. "Parallel Distributed Computing Using Python." *Advances in Water Resources*. Elsevier BV, September (2011). https://doi.org/10.1016/j.advwatres.2011.04.013.
9. Kamp, Michael, Mario Boley, Olana Missura, and Thomas Gärtner. "Effective Parallelisation for Machine Learning." *arXiv*, (2018). https://doi.org/10.48550/ARXIV.1810.03530.
10. Shashank, Ojha and Kylee, Santos. Parallelizing Gradient Descent on Different Architectures. https://shashank-ojha.github.io/ParallelGradientDescent/ (accessed December 10, 2023)
11. Maleki, Saeed, Madanlal Musuvathi, and Todd Mytkowicz. "Parallel Stochastic Gradient Descent with Sound Combiners." *arXiv*, (2017). https://doi.org/10.48550/ARXIV.1705.08030.

12. Mochurad, Lesia. “Optimization of Regression Analysis by Conducting Parallel Calculations.” *COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems*, April 22–23, 2021, Kharkiv, Ukraine

13. Lin, Chieh-Yen, Cheng-Hao Tsai, Ching-Pei Lee, and Chih-Jen Lin. “Large-Scale Logistic Regression and Linear Support Vector Machines Using Spark.” *2014 IEEE International Conference on Big Data (Big Data)*. IEEE, October 2014. https://doi.org/10.1109/bigdata.2014.7004269.

14. Woodsend, K., and Gondzio, J. “Hybrid MPI/OpenMP parallel linear support vector machine training.” *Journal of Machine Learning Research*, 10, Article 1937-1953.

15. Mitchell, Lawrence, Terence M. Sloan, Muriel Mewissen, Peter Ghazal, Thorsten Forster, Michal Piotrowski, and Arthur S. Trew. “A Parallel Random Forest Classifier for R.” *Proceedings of the Second International Workshop on Emerging Computational Methods for the Life Sciences*. ACM, June 8, 2011. https://doi.org/10.1145/1996023.1996024.

16. Jie Hu. Parallelized Random Forests Learning, CSE 633 Course Project. https://cse.buffalo.edu/faculty/miller/Courses/CSE633/Hu-Fall-2012-CSE633.pdf (accessed December 10, 2023)

17. Kenzhebek, Yerzhan, Timur Imankulov, Darkhan Akhmed-Zaki, and Beimbet Daribayev. “Implementation of Regression Algorithms for Oil Recovery Prediction.” *Eastern-European Journal of Enterprise Technologies*. Private Company Technology Center, April 30, 2022. https://doi.org/10.15587/1729-4061.2022.253886.

IRSTI 28.23.29 https://doi.org/10.26577/jpcsit2023v1i4a3

A.M. Madiev*, D.K. Sissenov, R.Y. Shein, N.M. Kaziyeva

L.N. Gumilyov Eurasian National University, Astana, Kazakhstan
*e-mail: ablaikhanmadiev@gmail.com

REVIEW OF EXISTING BARCODE SOLUTIONS IN BIOMETRICS AND ITS APPLICATIONS BASED ON QR CODES

Abstract. In this study, a comprehensive analysis of existing solutions for encoding barcodes in the field of biometrics is presented, with a special emphasis on integration into QR codes. The research begins with an introduction to the analysis topic, highlighting the critical intersection of biometrics and QR code technology. The main objective is to assess potential areas of application and the advantages of using QR codes as a means of encoding biometric data. The study underscores the importance of this integration in strengthening security measures, optimizing identification processes, and expanding the scope of application in various domains. Additionally, the study describes the methodology employed, including a detailed review of relevant literature and a general analysis of existing solutions. The results of this research shed light on the effectiveness and adaptability of biometric QR codes, providing valuable insights into their practical application. This study makes a significant contribution to the field of biometric technology, illuminating the innovative use of QR codes. The research findings serve as a valuable resource for both academic circles and practical implementation, offering a foundation for further refinement of secure identification methods.

Key words: Biometric Methods, Barcodes, QRCodes, Personal Identification, Biometric Authentication, Biometric Technology.

1 Introduction

Relevance of the Topic: Since the beginning of the 21st century, we have witnessed a rapid development of technologies that affect all aspects of modern life. A crucial aspect of this progress is the question of personal authentication in the digital environment. In this context, the topic addressed in this article – the interaction of biometric authentication methods and QR codes – represents an extremely relevant and promising research direction. Research Objective: Our main goal is to survey existing biometric authentication methods and QR codes, and then analyze the possibilities of their combined application. This analysis will allow us to assess the prospects for improving authentication methods and enhancing security in the digital environment. With the growth of online transactions, electronic documents, and data storage, security and authentication issues become critically important. Traditional authentication methods, such as passwords, are increasingly vulnerable to new forms of cyber threats. In this context, the development and analysis of biometric methods and QR codes appear as promising avenues to enhance security in the digital space. Despite the active interest in the authentication topic, the integration of biometric methods with QR codes remains a relatively underexplored area. Existing works focus on partial aspects of this issue, suggesting the presence of untapped potentials in this field [1-2].

2 Overview of Existing Biometric Methods

Facial recognition is a widely used technology in the modern world, as it can be applied in all areas of society's activities. Facial recognition utilizes facial characteristics, unlike other types of biometric authentication (which we will discuss below). This technology records the spatial geometry of distinctive facial features for further recognition. Facial recognition can be divided into four major groups based on the method of processing: 2D facial recognition, 3D facial recognition, facial recognition based on skin texture, and facial recognition based on thermal imaging. In 2D facial recognition, the person's im- age is used for identification [3]. It goes through several stages before yielding an identification result with the answer to who exactly is depicted on it. The identification process consists of the following stages:

– Face recognition;
– Face detection;
– Face alignment;

– Feature extraction;

– Matching features with other features from the database consisting of people's features for identification.

Some of the most common biometric characteristics applied in modern technologies worldwide [4], including in Kazakhstan, include:

Fingerprint recognition: One of the most common and widely used types of biometric identification. Fingerprints are unique to each individual and extremely difficult to forge.

Facial recognition: A technology that uses geometric features of the face for identification and authentication. Applied in surveillance systems, mobile devices, and other fields.

Iris recognition: This method uses the unique features of the iris of the eye for identification.

Voice biometrics: Analysis of unique characteristics of a person's voice, such as tone, speech features, and other parameters, for authenticity verification.

Retina scan: Examination of the unique retina pattern for establishing identity.

Electrocardiogram (ECG): Measurement and analysis of unique electrical signals of the heart for biometric authentication.

In Kazakhstan, biometric technologies are also actively applied in various fields, including government identification management, security, access control, financial services, and many others. They are used for creating passports, driver's licenses, electronic identity documents, as well as in security and access control systems in enterprises and organizations. These technologies contribute to enhancing security levels, improving efficiency, and convenience in various areas of activity.

3 Barcode Technology

Barcode technology plays a crucial role in modern business by providing efficient tools for the identification and tracking of goods, as well as optimizing logistical and managerial processes. Barcodes, as an established technology, have several significant advantages that are important to consider when analyzing biometric QR codes.

Barcodes stand out for their ease of use. Generating and reading them requires minimal resources and time. This is particularly important in the context of widespread adoption – even untrained users can operate this technology.

The effectiveness of barcodes is not limited to a single industry. They are successfully used in retail, logistics, healthcare, and many other sectors. This versatility confirms their importance as a fundamental tool for tracking and identification.

The use of barcodes significantly reduces the likelihood of errors compared to manual data entry. This is especially critical in fields where data accuracy is of paramount importance. Additionally, they serve as a means of protection against counterfeiting, as they are difficult to replicate.

Barcodes, as a technology with a long history, have seamlessly integrated with new technological developments. They easily integrate with modern mobile devices, enabling their use in mobile applications and services.

The use of barcodes allows companies to gather detailed information about sales, inventory, and consumer behavior. This data can be analyzed to optimize processes and develop business growth strategies.

Overall, barcode technology represents a reliable and versatile tool for data identification and tracking. Its effectiveness and ease of use make it indispensable in various industries, making it a significant milestone in the development of modern technologies [5].

4 Biometric Barcodes: Technologies and Applications

In the modern world, identification plays a pivotal role in various spheres. Biometrics, the science of recognition based on unique physiological and behavioral characteristics, has become a reliable tool for security, access control, and authentication.

Biometric barcodes have garnered special attention. They are unique symbolic sequences that store information about an individual in a specific format. They combine biometric technologies with the convenience and reliability of encoding.

The fundamental principle behind the operation of a biometric barcode lies in the fact that the unique sequence of bars and spaces in the barcode can be associated with specific characteristics of a person. These characteristics may include various parameters such as anthropometric data, biochemical indicators, medical history, and more. Each barcode represents a unique identifier for the corresponding set of data [6].

Biometric barcodes are unique encoded sequences capable of storing information about an individual's biological characteristics. There are several different types of biometric barcodes:

Anthropometric Barcodes are used to encode data about measurements and body parameters of a person. They include information about the sizes and shapes of various body parts, such as finger lengths, wrist circumference, and other anthropometric characteristics. This data can be used for identifying a person based on their physical parameters.

Biochemical Barcodes contain information about biological indicators of a person, such as levels of specific substances in the body. This data may include information about the concentration of biochemical markers like glucose, cholesterol, proteins, and other substances that can be used for unique identification.

Genetic Barcodes utilize information about a person's DNA to create a unique identifier. They may contain data about specific genetic markers or sequences that can be unique to each individual.

Geometric Barcodes use geometric parameters, such as face shape, dimensions of facial features, distances between key points, to create a unique identifier. These parameters can be used for facial recognition.

Behavioral Barcodes encode information about a person's behavior, such as walking style, typing manner, gestures, and other behavioral characteristics. This data can be used to create a unique behavior profile of an individual [7].

Biometric barcodes represent an innovative method of identification that has found application in various countries worldwide. Here are several examples of their use:

In the USA, biometric barcodes are employed in healthcare systems for precise patient identification. This improves the quality of medical services and helps prevent errors in treatment.

In India, biometric barcodes are used to authenticate citizens when availing various government services, such as issuing identity cards and voting.

Japan is integrating biometric barcodes into security and access control systems. They are applied in offices and businesses to ensure the safety of personnel and visitors.

In the UK, biometric barcodes are utilized in the financial sector and banking industry. They provide secure customer authentication for financial transactions.

China is actively implementing biometric barcodes across various sectors including education, healthcare, and government services. This helps enhance the efficiency and accuracy of the services provided.

In Brazil, biometric barcodes are employed in access control systems for government institutions and offices. They offer robust protection against unauthorized access.

South Korea uses biometric barcodes in public transport systems. This simplifies fare payment and enhances passenger safety.

Global experience demonstrates diverse applications of biometric barcodes in different areas of life and public activities. Their reliability, effectiveness, and convenience make this technology increasingly popular and in demand [8-9].

Biometric barcode technology represents a promising direction in the field of identification and control in Kazakhstan. It is actively applied in various sectors, covering both governmental structures and the private sector.

Healthcare and Medicine: One of the primary applications of biometric barcodes in Kazakhstan is in healthcare. They are used for patient identification, ensuring accurate and swift medical assistance. Biometric barcodes allow linking medical history, analyses, and other data to specific patients, ensuring high-quality healthcare services.

Government Institutions: In Kazakhstan, biometric barcodes are used for citizen identification in government institutions. For instance, they may be applied when obtaining passports, driver's licenses, or other documents requiring precise identity verification.

Financial Sector: The banking sector in Kazakhstan is also actively incorporating biometric barcodes. They are used to authenticate customers during financial transactions, thereby enhancing security levels and preventing fraud.

Security Systems and Access Control: In the realm of security, biometric barcodes find application in access control systems. They help prevent unauthorized access to restricted areas, ensuring high levels of security in offices, enterprises, and other facilities.

Trade and Retail: In retail, biometric barcodes can be employed to manage customer loyalty, as well as to ensure accurate identification during payment for goods and services.

Transport Systems: In public transport systems and transportation organizations, biometric barcodes can be used for passenger tracking and ensuring travel safety.

Education: In educational institutions, biometric barcodes can be used for student identification and ensuring security within educational facilities [10].

Biometric technologies, including the use of barcodes, are in a constant state of development with the emergence of new innovations and the refinement of existing approaches. This leads to expanded capabilities and areas of application for these technologies.

Integration with Artificial Intelligence (AI) improves the data processing algorithms of biometric barcodes, increasing system accuracy and reliability. Multimodal biometric systems combine multiple biometric methods, enhancing the level of security and accuracy in the identification process. Embedded sensors in mobile devices such as smartphones and tablets enable the use of biometric barcodes for various tasks, including authentication and conducting payments. In addition to physiological parameters such as fingerprints, attention is being given to parameters related to individual behavior (e.g., voice analysis or gait analysis) [11]. The application of deep learning and neural networks significantly enhances the efficiency of face recognition. The implementation of biometric barcodes in medical institutions contributes to improved patient identification and the prevention of medical errors. Protecting user privacy is becoming a priority. Methods for anonymizing data are being developed to strike a balance between security and privacy. Biometric technologies utilizing barcodes can be integrated into smart devices, enhancing security levels and providing convenience of use. In the tourism and hospitality industry, biometric barcodes can be applied to expedite the guest check-in process in hotels and airports. Furthermore, in the field of logistics management and inventory, the use of biometric barcodes enables the improvement of inventory management processes and control over the movement of goods [12].

5 Representing Facial Information in QR Codes

As part of this experiment, work was conducted to investigate the transformation of facial encodings obtained using the face_recognition library into QR codes. The experiment aimed to develop an efficient method for representing facial encodings using QR code technology. The photograph used for the transformation is shown in Figure 1.

Figure 1 – Face Photograph

The encoding of the face generated using the computer vision library face_recognition, used in the experiment, is shown in Figure 2.

```
[array([-9.97023135e-02, 4.94740456e-02, 7.98235759e-02, -7.43214488e-02,
    -1.74680203e-01, 2.79045738e-02, -1.27507180e-01, -1.43855691e-01,
     9.56659392e-02, -1.17645845e-01, 2.61246145e-01, -4.02291156e-02,
    -1.64547428e-01, -6.32798672e-02, -9.90153030e-02, 1.91734418e-01,
    -1.89899921e-01, -1.40684173e-01, 4.19026706e-03, 6.11237958e-02,
     9.44161266e-02, -2.53463201e-02, 2.35297140e-02, 9.19905305e-02,
    -1.68697655e-01, -3.25366795e-01, -1.21952787e-01, -6.61166087e-02,
    -5.02436869e-02, -1.31008953e-01, -8.16686377e-02, -6.62299246e-03,
    -2.20089406e-01, -4.14269194e-02, -2.67850384e-02, 4.34266329e-02,
    -1.65863633e-02, -6.26730323e-02, 1.69974893e-01, 6.15422018e-02,
    -2.48007476e-01, 1.44100130e-01, 3.63545120e-02, 2.17177197e-01,
     1.54172152e-01, 3.73906642e-02, 4.68519032e-02, -1.43617228e-01,
     1.15307607e-01, -1.93156064e-01, 5.25930226e-02, 1.44928217e-01,
     7.74639547e-02, 6.38333708e-03, -4.41481099e-02, -8.97464156e-02,
     1.52576864e-02, 1.51003569e-01, -1.32537097e-01, 1.12094909e-01,
     1.00682348e-01, -3.61202210e-02, 3.09028402e-02, -6.22204319e-02,
     1.40536264e-01, 1.15435317e-01, -6.55267537e-02, -2.16769978e-01,
     8.74410942e-02, -9.69708562e-02, -1.31602004e-01, 6.40264750e-02,
    -1.47282735e-01, -2.15343893e-01, -2.94206232e-01, 1.74608827e-03,
     4.26398218e-01, 9.73839611e-02, -2.45120198e-01, 4.33005765e-03,
    -8.52620453e-02, -4.55107987e-02, 8.27133283e-02, 9.23452899e-02,
     2.27138102e-02, -1.54741108e-04, -1.18523955e-01, 2.08501369e-02,
     2.84547150e-01, -3.15525308e-02, 2.27908157e-02, 2.60421723e-01,
     5.82121126e-02, 5.17934412e-02, 5.40908352e-02, 6.96922839e-02,
    -1.01050869e-01, 6.62658662e-02, -1.35656029e-01, -1.46889240e-02,
    -5.61088547e-02, -1.68944784e-02, -1.85205936e-02, 8.92843828e-02,
    -1.85447246e-01, 1.84819877e-01, -1.52932107e-02, -3.85973640e-02,
    -4.33367640e-02, 3.24743390e-02, -1.36015892e-01, -3.40196043e-02,
     1.40036732e-01, -2.04911560e-01, 1.80357859e-01, 2.03965828e-01,
     1.44400179e-01, 8.23330805e-02, 1.48437679e-01, 5.47322854e-02,
    -2.67138630e-02, 8.28737020e-03, -1.76903620e-01, -3.75504792e-02,
     9.47215557e-02, -8.78797621e-02, 1.04836330e-01, 2.59825587e-02])]
```

Figure 2 – Face Encoding

The Python code for converting face encoding into a QR code with a size of 177x177 modules is shown in Figure 3.

```python
qr = qrcode.QRCode(
    version=1,
    error_correction=qrcode.constants.ERROR_CORRECT_L,
    box_size=10,
    border=4,
)

# Добавляем данные в QR-код
qr.add_data(face_encoding_str)
qr.make(fit=True)

# Создаем изображение QR-кода
img = qr.make_image(fill_color="black", back_color="white")

# Изменяем размер изображения до 177x177 на LANCZOS
img = img.resize((177, 177), Image.Resampling.LANCZOS)
```

Figure 3 – Part of the Python code

The experiment successfully resulted in converting face encodings into QR codes with module sizes of 177x177 pixels. The generated QR codes serve as an efficient means for storing and transmitting information about the face encodings. A detailed block diagram is shown in Figure 4.

Figure 4 – QR Code Generation Flowchart

6 Conclusions

The conducted experiment has demonstrated the unique application of biometric barcodes in the modern technological environment. The specially developed Python algorithm efficiently transformed the facial image into a QR code. This barcode, containing information about a person's biological characteristics, represents an innovative approach to identity verification.

The results of the experiment confirmed the high efficiency and accuracy of converting biometric data into a barcode. This approach can be applied in various fields, ranging from medical institutions and government structures to the financial sector and security systems.

However, despite all the advantages, it is important to consider issues related to data protection and privacy. The implementation of biometric technologies requires careful analysis and compliance with relevant regulations.

Overall, the experiment's results indicate the promising and significant development of biometric barcodes in the modern world. These technologies offer new opportunities to enhance the efficiency and security of various aspects of human activity.

Funding

This study was funded by the Ministry of Science and higher education of the Republic of Kazakhstan, Grant No. AP19678000.

References

1. G. Kukharev, N. Kazieva and N. Shchegoleva, «Animation Qr-Codes for Facial Biometry Tasks and Its Applications*,» 2021 10th Mediterranean Conference on Embedded Computing (MECO)*, Budva, Montenegro, 2021, pp. 1-4, https://doi.org/10.1109/MECO52532.2021.9460179.
2. Kukharev G.A., Kaziyeva N. QR codes with animation for digital passes. Scientific and Technical Journal of Information Technologies, Mechanics and Optics, 2020, vol. 20, no. 3, pp. 461–464 (in Russian) https://doi.org/10.17586/2226-1494-2020-20-3-461-464
3. C Metody obrabotki i opredeleniya izobrazheniy lits v zadachakh biometrii / G. A. Kukharev, Ye. I. Kamenskaya, YU. N. Matveyev, N. L. Shchegoleva; pod red. M. V. Khitrova. – SPb.: Politekhnika, 2013. – 288s.: il. (in Russian)
4. Woodward, John D., Christopher Horn, Julius Gatune, and Aryn Thomas, Biometrics: A Look at Facial Recognition. Santa Monica, CA: RAND Corporation, 2003. https://www.rand.org/pubs/documented_briefings/DB396.html. Also available in print form.
5. Hassan Yusuf Ahmed, Dr. Ruchi Sharma, 0, Barcodes Technology, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH \& TECHNOLOGY (IJERT) V-Impact – 2018 (Volume 06 – Issue 17)
6. Rachapalli, D.R., Kalluri, H.K. Multi-modal compound biometric feature set security and person authentication using cancelable 2D color barcode pattern generation technique. *Int. j. inf. tecnol.* **14**, 201–214 (2022) https://doi.org/10.1007/s41870-021-00819-3
7. Kaziyeva, N., Kukharev, G., Matveev, Y. (2018). Barcoding in Biometrics and Its Development. In: Chmielewski, L., Kozera, R., Orłowski, A., Wojciechowski, K., Bruckstein, A., Petkov, N. (eds) Computer Vision and Graphics. ICCVG 2018. Lecture Notes in Computer Science(), vol 11114. Springer, Cham. https://doi.org/10.1007/978-3-030-00692-1_40
8. Bashkirova O.V. Digital identification system's risk classification (based on foreign experience). *Issues of Risk Analysis*. 2023;20(1):64-77. (In Russ.) https://doi.org/10.32686/1812-5220-2023-20-1-64-77
9. Ajish, S., Anil Kumar, K.S. (2022). Secure Mobile Internet Banking System Using QR Code and Biometric Authentication. In: Pandian, A.P., Fernando, X., Haoxiang, W. (eds) Computer Networks, Big Data and IoT. Lecture Notes on Data Engineering and Communications Technologies, vol 117. Springer, Singapore. https://doi.org/10.1007/978-981-19-0898-9_60
10. Proyekt programmy sozdaniya Natsional'noy platformy tsifrovoy biometricheskoy identifikatsii na 2022-2024 gody // [Elektronnyy resurs] Rezhim dostupa: https://legalacts.egov.kz/npa/view?id=14023459, – S. 135 (data obrashcheniya: 01.10.2023). (in Russian)
11. Kukharev, G.A., Kaziyeva, N. Tsymbal, D.A. Barcoding Technologies for the Tasks of the Facial Biometrics: State of the Art and New Solutions. Pattern Recognit. Image Anal. 28, 496–509 (2018). https://doi.org/10.1134/S1054661818030124
12. D. V. Belyshev, Ya. I.–O. Guliev. Usage of Barcodes Technology in Hospital Information Systems // Proceedings of Program Systems institute scientific conference "Program systems: Theory and applications". – Pereslavl-Zalesskij, v. 2, 2009. – p. 71–96. – ISBN 978-5-901795-18-7 (in Russian)

IRSTI 28.23.37 https://doi.org/10.26577/jpcsit2023v1i4a4

Y. Kenzhebek[1*], A. Amangeldiyev[2]

[1]National Engineering Academy of the Republic of Kazakhstan, Almaty, Kazakhstan
[2]Astana IT University, Astana, Kazakhstan
*e-mail: kenzhebekyerzhan@gmail.com

PHYSICS-INFORMED NEURAL NETWORKS FOR PREDICTING PRESSURE DISTRIBUTION IN POROUS MEDIA

Abstract. In recent years, the integration of modern information technologies has become pervasive across various industries, and the oil sector is no exception. The utilization of high-performance computing technologies, artificial intelligence algorithms, and advanced methods for data collection, processing, and storage has been instrumental in addressing challenges related to enhancing oil recovery. While deep learning has demonstrated significant advancements in diverse applications, its application to solving partial differential equations has recently gained prominence. A noteworthy strategy entails substituting conventional numerical techniques with neural networks that approximate solutions to partial differential equations. Physics-informed neural networks (PINNs) represent a significant development in this domain by incorporating partial differential equations directly within the loss function of neural network through automatic differentiation. This study presents a numerical algorithm and a PINNs to solve the one-dimensional equation describing the distribution of water and oil pressure within the context of the Buckley-Leverett mathematical model. The obtained results include the numerical solution and predictions derived from the PINN neural network to solve the pressure distribution. The insights gained from the comparative analysis underscore the promising role of PINNs as a robust and competitive tool for addressing intricate problems within the realm of complex fluid dynamics.

Key words: Enhanced Oil Recovery, Physics-Informed Neural Networks (PINNs), Deep Learning, Numerical Method.

1 Introduction

In the last decade and a half, deep learning, represented by deep neural networks, has demonstrated significant effectiveness in diverse applications like computer vision and natural language processing [1]. Although its success in these areas, the widespread adoption of deep learning in scientific computing has been limited. Nevertheless, a contemporary trend is evolving, centering around the utilization of deep learning to address partial differential equations [2]. In this strategy, conventional methods of numerical discretization are substituted with neural networks that provide approximations for solving differential equations.

Achieving an approximate solution to differential equations through deep learning techniques involves a crucial step: restricting the network to minimize the residual of partial differential equations. Various methods have been suggested for this objective. In contrast to traditional grid-oriented techniques like finite difference methods and finite element methods, deep learning presents a mesh-free alternative, capitalizing on automatic differentiation [3]. Some of these approaches may be applicable only to specific categories of issues, such as input domains like images [4] or parabolic partial differential equations [5].

The first glimpses of the prospect of using structured prior information to create data-efficient and physics-aware machine learning have already been demonstrated in recent research [6]. In that work, the authors employed Gaussian process regression to develop functional representations adapted to a given linear operator, accurately deriving solutions. Additionally, they provided uncertainty assessments for diverse model scenarios in the realm of mathematical physics. Subsequent research [7, 8] has suggested expansions to nonlinear issues within the domains of logical inference and identifying systems. In the light of the versatility and mathematical sophistication of Gaussian processes in capturing prior information, dealing with nonlinear challenges imposes two significant limitations.

Physics-Informed Neural Networks are described in works [9-13], where the use of these networks adheres to the laws of physics (for problems described by differential equations). In [9], the application of PINNs to classical fluid mechanics and quantum mechanics problems is discussed.

In [10, 11], the authors introduce a deep learning approach called physics-informed neural networks for quantitative uncertainty assessment in ordinary differential equation systems. In 2020, this method started to be applied for mapping heart activations [12] and evaluating fluid conductivity governed by Darcy's law [13]. The results of these works demonstrate that applying PINNs can yield results comparable to those of physical models.

The early stages of developing Physically Informed Neural Networks primarily concentrated on comprehending and enhancing their training dynamics. Initially, there was a significant challenge related to the disparate convergence rates among various elements of the loss function, a pivotal aspect in neural network training. This discrepancy often resulted in scenarios where the network prioritized learning the physical dynamics at the expense of fitting the data or vice versa. In [14], the authors contribute to this comprehension by conducting an wide-ranging survey of literature on PINNs, elucidating their characterization, advantages, and disadvantages. Various PINN variants, including physics-constrained neural networks (PCNN), variational hp-VPINN, and conservative PINN (CPINN), are discussed, highlighting the diversity within the field. The seminal work in [15] delves deep into this challenge, providing crucial insights into the concepts of training PINNs. It underscores the necessity for a balanced approach that ensures equitable representation of both physical laws and data throughout the learning process. This understanding played a pivotal role in guiding subsequent advancements in the field.

Establishing a robust theoretical foundation, particularly concerning error analysis, is a crucial aspect of the development and utilization of physics-informed neural networks. Linear parabolic differential equations are commonly employed for modeling time-dependent phenomena such as heat transfer and diffusion processes. The error analysis outlined in [16] holds significance as it provides valuable insights into the precision and dependability of PINNs when employed in the context of these equations. This study systematically investigates various error sources, including discretization, approximation, and algorithmic errors, offering benchmarks to assess the efficacy of PINNs. A noteworthy contribution of this analysis lies in addressing the curse of dimensionality, a challenge prevalent in high-dimensional spaces where the space volume escalates exponentially with the number of dimensions. Traditionally, this curse poses computational and analytical challenges in numerical methods. The findings presented in [16] illustrate that PINNs, equipped to handle high-dimensional data while incorporating physical laws, adeptly surmount this challenge. This error analysis not only enriches the comprehension of PINNs' capabilities but also instills confidence in their application to intricate, high-dimensional issues.

2 Physics-Informed Neural Networks

Automatic differentiation. The method of automatic differentiation for calculating derivatives of network outputs relative to network inputs is considered. Considering that neural networks are compositional functions, automatic differentiation repeatedly applies the chain rule to calculate derivatives. Automatic differentiation consists of two steps: a forward propagation to compute the values of all variables and a subsequent backward propagation to calculate the derivatives.

The DeepXDE library was explored to implement a physics-informed neural network. DeepXDE is a deep learning library on top of TensorFlow that supports many features: construction of primitive and complex geometries, support for multiple boundary conditions for partial differential equations, 6 sample filling methods, ease of saving and loading the model during training.

The algorithm for solving differential equations using PINNs consists of four stages:

1. Construction of a neural network $u(x; \theta)$ with parameters θ;

2. Specify two training sets: for the partial differential equation and the boundary/initial conditions that are built into the loss function;

3. Determination of the loss function by aggregating the weighted L^2 norms as residuals from partial differential equations and boundary conditions;
4. Training a network to determine the optimal parameter θ* through the reduction of the loss function.

3 PINNs for modeling the pressure distribution in the Buckley-Leverett model

A numerical algorithm has been developed to solve the equation for distribution of pressure from the Buckley–Leverett mathematical model. To numerically solve the pressure equation, the Jacobi iterative method was used.

$$div(\vec{v}_w) + div(\vec{v}_o) = 0, \qquad (1)$$

where $\vec{v}_w, \vec{v}_o$ – fluid flow speed, which is expressed by the following Darcy's law:

$$\vec{v}_i = -K_0 \frac{f_i(s)}{\mu_i} \nabla P, i = w, o, \qquad (2)$$

$f_i(s), \mu_i$ – relative phase permeabilities and viscosities of the water and oil phases, respectively, K_0 – absolute permeability. Substituting equation (2) to equation (1), we obtain a one-dimensional equation for pressure:

$$\frac{\partial}{\partial x}\left(Mx \frac{\partial P}{\partial x}\right) = 0, \qquad (3)$$

where Mx is denoted as follows:

$$Mx = (-K_0 \frac{f_1(s)}{\mu_1}) + (-K_0 \frac{f_2(s)}{\mu_2}).$$

A neural network PINN was built to solve the one-dimensional pressure equation from the Buckley-Leverett mathematical model. Figure 1 shows the PINN architecture for solving this problem:

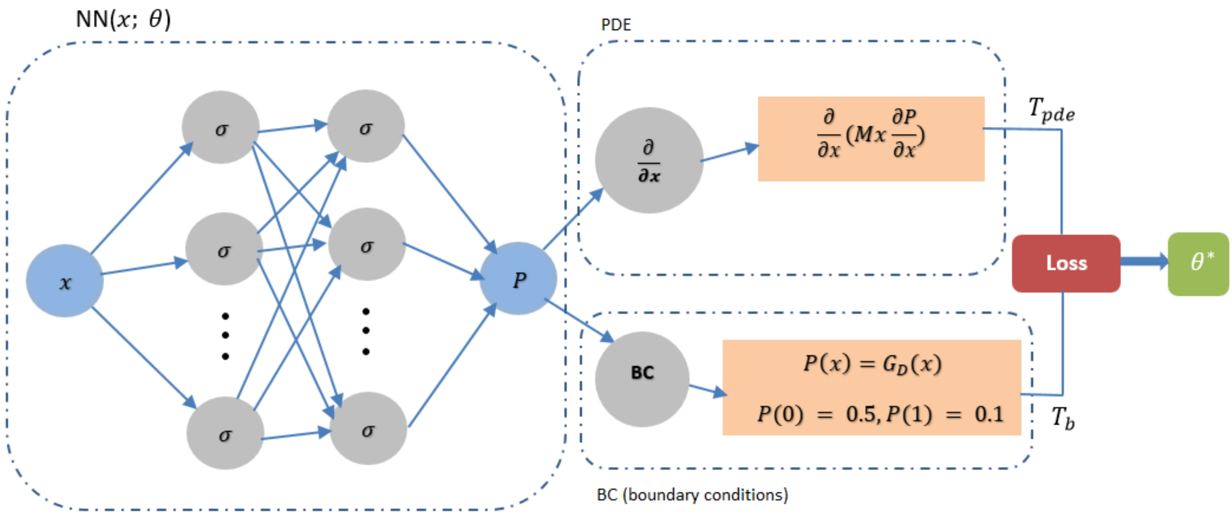

Figure 1 – PINN neural network architecture for pressure equation

In PINN, an initial step involves constructing a neural network, denoted as P(x; θ), to serve as a substitute for the solution p(x). This neural network takes an input x and produces a vector with dimensions identical to *p*. Here, θ = {W, b} represents the collection of all weight matrices and bias vectors within the neural network P. An advantageous feature of PINN, particularly in opting for neural networks to approximate *p*, lies in the ability to compute derivatives of P concerning input x. This is achieved through the chain rule for differentiating compositions of functions using automatic differentiation (AD), conveniently integrated into machine learning packages.

The loss function is considered, defined as a weighted summation of the L^2 norm of residuals from the equation and boundary conditions:

$$L(\theta;T) = w_{pde}L_{pde}(\theta;T_{pde}) + w_b L_b(\theta;T_b).$$

This loss function will be used for model training so that it fits the pressure equation for the oil displacement problem and satisfies the boundary conditions.

Here L_b is the loss function for the boundary conditions. This is used to ensure that the model satisfies the boundary conditions.

$$L_b = \frac{1}{N_b}\sum_{i=1}^{N_b}((\mathrm{p}(x_i) - p)^2|_{x_i},$$

where N_b is the number of points on the border, and x_i are the coordinates of these points.

Whereas L_{pde} is the loss function for the partial differential equation. This is used to allow the model to approximate the equation. For the pressure equation from the Buckley-Leverett model is defined as L^2 norm between the left and right sides of the equation, that is, the discrepancy of the equations. As can be seen from the loss function, labeled data is not used here, that is, the physical limitations of the equation under consideration are considered.

$$L_{pde} = \frac{1}{N_{pde}}\sum_{j=1}^{N_{pde}}\left|\frac{\partial}{\partial x}\left(Mx\frac{\partial P}{\partial x}\right)\right|^2 |_{x_j},$$

where N_{pde} is the number of points at which the partial differential equation is applied.

A fully connected neural network was used, consisting of 4 layers (3 hidden layers) and a width of 32 neurons: [1] + 32*[3] + [1]. The x-space component is taken as the input parameters of the network. Optimizers of the "Adam" type was selected as the network hyperparameter, and the learning rate was 0.001. Testing of 10,000 epochs was carried out for training a neural network, where the number of trained (collocation) points is 100 and two points are used for the Dirichlet boundary condition. PINN construction is implemented using the deepxde deep learning library on top of TensorFlow, which supports many functions for constructing geometries. In Figures 2 and 3 you can see the network training results and prediction:

For testing, 120 points of the numerical solution of the pressure equation and the predicted 120 points of the PINN neural network were used. Figure 3 shows a visualization of the numerical solution and the PINN solution. From Figure 3 it is noticeable that the values are very close, the next Figure 4 shows the absolute error between the two solutions.

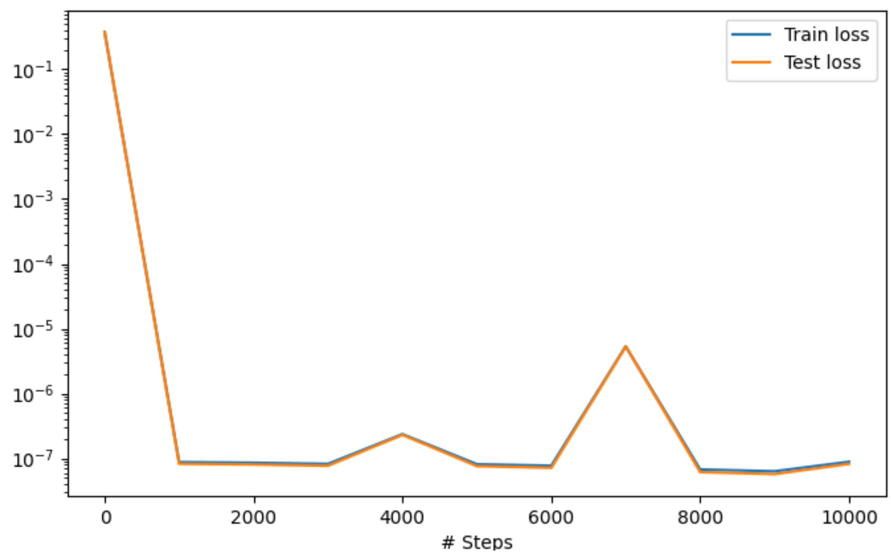

Figure 2 – PINN training loss history

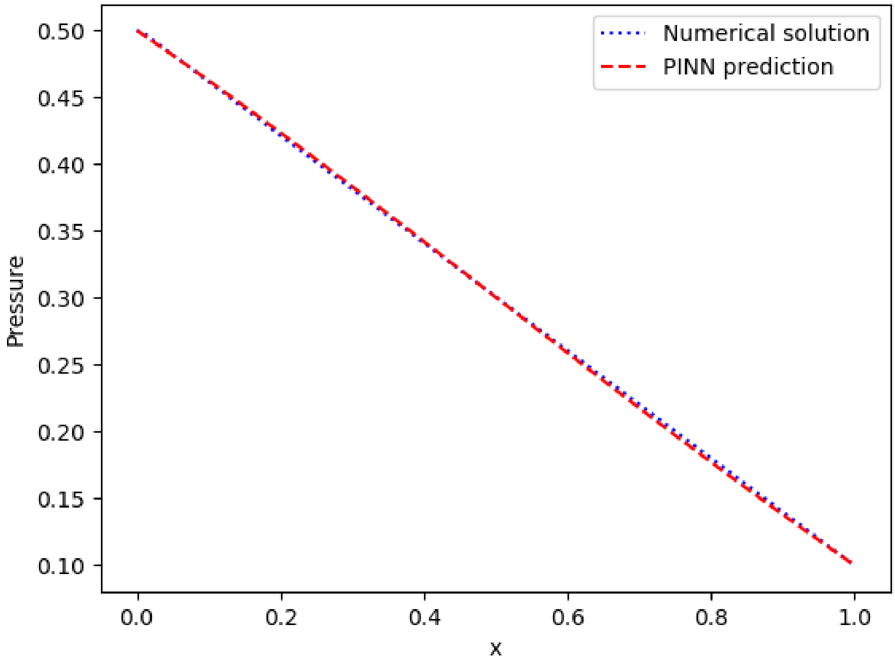

Figure 3 – Comparison of the numerical solution of the pressure equation with the PINN prediction

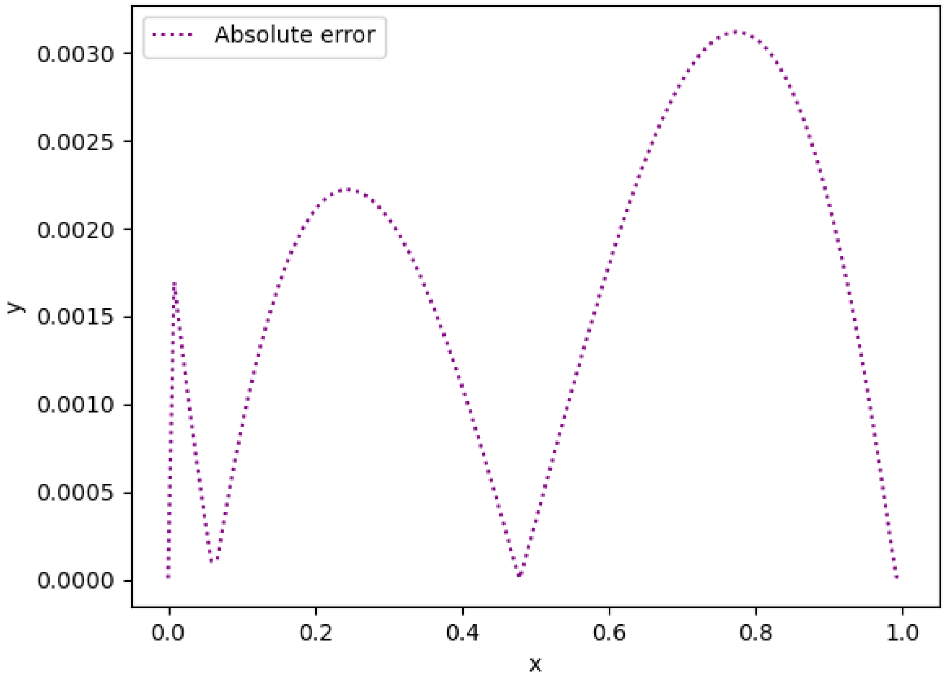

Figure 4 – Absolute error between the numerical solution and the PINN prediction

4 Conclusion

A numerical algorithm and a fully connected neural network PINN have been developed and tested to solve the pressure equation. The results of the numerical solution of the pressure equation are compared with the prediction of a physics-informed neural network. In the comparative analysis, it is revealed that the absolute error between the PINN predictions and the numerical solution is within the range of 0.0005-0.003. This quantifiable measure underscores the precision and effectiveness of PINNs in capturing the complex dynamics of the pressure equation, positioning them as a valuable alternative to traditional linear approximation methods. At a more fundamental level, physics-informed neural networks provide a nonlinear approximation of a function and its derivatives, whereas traditional methods provide a linear approximation.

Acknowledgements

This research was funded by the Science Committee of the Ministry of Science and Higher Education of the Republic of Kazakhstan (Grant No. AP14871644).

References

1 LeCun Y., Bengio Y., Hinton G. (2015) Deep learning. Nature, 521, p. 436.

2 Baker N., Alexander F., Bremer T., Hagberg A., et al. (2019) Workshop report on basic research needs for scientific machine learning: Core technologies for artificial intelligence. Tech. report, US DOE Office of Science, Washington, DC (United States).

3 Poggio T., Mhaskar H., Rosasco L., Miranda B., Liao Q. (2017) Why and when can deep-but not shallow-networks avoid the curse of dimensionality: a review. International Journal of Automation and Computing, 14, 503–519.

4 Khoo, Yuehaw & Lu, Jianfeng & Ying, Lexing. (2017). Solving parametric PDE problems with artificial neural networks. European Journal of Applied Mathematics, 32, 421-435. 10.1017/S0956792520000182.

5 Beck C., Ee W., Jentzen A. (2019) Machine learning approximation algorithms for highdimensional fully nonlinear partial differential equations and second-order backward stochastic differential equations. Journal of Nonlinear Science, 29, 1563–1619. https://doi.org/10.1007/s00332-018-9525-3

6 Raissi M., Perdikaris P., Karniadakis G.E. (2017) Inferring solutions of differential equations using noisy multi-fidelity data. J. Comput. Phys., 335, 736–746.

7 Raissi M., Perdikaris P., Karniadakis G.E. (2017) Numerical Gaussian processes for time-dependent and non-linear partial differential equations. SIAM J. Sci. Comput., 40(1), 172–198.

8 Raissi M., Karniadakis G.E. (2018) Hidden physics models: machine learning of nonlinear partial differential equations. Journal of Computational Physics, 357, 125-141.

9. Raissi M., Perdikaris P., Karniadakis G.E. (2019) Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics, 378, 686-707.

10 Zhu Y., Zabaras N., Koutsourelakis P.S., Perdikaris P. (2019) Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data. Journal of Computational Physics, 394, 56-81.

11 Yang Y., Perdikaris P. (2019) Adversarial uncertainty quantification in physics-informed neural networks. Journal of Computational Physics, 394, 136-152.

12 Sahli F., Yang Y., Perdikaris P., Hurtado D.E., Kuhl E. (2020) Physics-Informed Neural Networks for Cardiac Activation Mapping. Frontiers in Physics, 8. 10.3389/fphy.2020.00042.

13 Tartakovsky A.M., Marrero C.O., Perdikaris P., Tartakovsky G.D., Barajas-Solano D. (2020) Physics-Informed Deep Neural Networks for Learning Parameters and Constitutive Relationships in Subsurface Flow Problems. Water Resources Research, 56 (5). https://doi.org/10.1029/2019WR026731

14 S. Cuomo, V. S. Di Cola, F. Giampaolo, G. Rozza, M. Raissi, and F. Piccialli, “Scientific Machine Learning Through Physics–Informed Neural Networks: Where we are and What’s Next,” Journal of Scientific Computing, vol. 92, no. 3, 2022, doi: 10.1007/s10915-022-01939-z.

15 S. Wang, X. Yu, and P. Perdikaris, “When and why PINNs fail to train: A neural tangent kernel perspective,” Journal of Computational Physics, vol. 449, p. 110768, Jan. 2022, doi: 10.1016/j.jcp.2021.110768.

16 T. De Ryck and S. Mishra, “Error analysis for physics-informed neural networks (PINNs) approximating Kolmogorov PDEs,” Adv Comput Math, vol. 48, no. 6, p. 79, Nov. 2022, doi: 10.1007/s10444-022-09985-9.

ISSN 2958-0846 eISSN 2958-0854 Journal of Problems in Computer Science and Information Technologies №4 (1) 2023 https://jpcsip.kaznu.kz

IRSTI 28.23.01 https://doi.org/10.26577/jpcsit2023v1i4a5

M.B. Mustafin[1*], Zh.S. Muratay[2], B.K. Orazov[3]

[1]U.A. Joldasbekov Institute of Mechanics and Engineering, Almaty, Kazakhstan
[2]Astana IT University, Astana, Kazakhstan
[3]Al-Farabi Kazakh National University, Almaty, Kazakhstan
*e-mail: mustafin.mb@gmail.com

HARNESSING GPU POWER FOR MACHINE LEARNING

Abstract. In the rapidly evolving landscape of data science and machine learning, the need for high-speed, efficient data processing is more critical than ever. This article delves into the pivotal role of graphics processing units (GPUs) in transforming the realm of data analytics and machine learning. GPUs, originally designed for rendering graphics in video games, have emerged as powerhouse tools in scientific computing due to their ability to perform parallel computations swiftly and effectively.

The work conducted a study and comparison of the performance of machine learning algorithms using the Scikit-Learn and RAPIDS cuML libraries on a GPU. Testing was carried out on various data volumes and the results confirmed significant execution speedup when using RAPIDS cuML. This highlights the practical importance of GPU acceleration for processing large data sets. In addition, the developed algorithms were successfully applied to predict the oil recovery factor based on the Buckley-Leverett mathematical model, demonstrating their effectiveness in the oil and gas industry. Overall, this article serves as a comprehensive overview of the current state and future prospects of GPU utilization in data processing and machine learning, providing valuable insights for both practitioners and researchers in the field.

Key words: GPU, Machine Learning, Linear Regression, Support Vector Machine Regression.

1 Introduction

In the modern world, the field of data and machine learning is becoming a key element in various areas, including science, industry, and technology. With the continuous increase in data volumes, there is a need for efficient methods and tools for processing and analyzing them. The use of graphics processing units (GPUs) has become an integral part of the data processing process, enabling parallel execution of complex computations.

The relevance of this topic is due to the rapid development of research in the field of optimization of machine learning algorithms using graphics processors. The authors of paper [1] describe modifications of the GPGPU-Sim simulator to support machine learning using PyTorch and cuDNN. This modification provided high accuracy of execution results and revealed new opportunities for optimizing the microarchitecture of GPUs for deep neural networks. In article [2], innovative methods are presented that accelerate feature generation and reduce model training time by 15.6 times. Using the RAPIDS.AI cuDF library and optimizations in PyTorch, the authors significantly reduced the prototyping time of deep learning-based recommendation systems. In work [3], a study is presented on the use of the cuML library, developed by NVIDIA to accelerate machine learning on GPUs, with particular attention to support for cluster systems with multiple GPUs. The authors propose a Python API for using MPI for communication in cuML, conduct analysis and benchmarking of algorithms.

This article presents an analysis of the effectiveness of various methods, including the use of CUDA technology for parallelizing matrix operations, the RAPIDS library set (cuML and cuDF) for data analysis and implementation of machine learning algorithms on GPUs [4, 5, 6].

This research also considers alternative machine learning methods, such as Linear Regression, SVM, implemented using the RAPIDS and Scikit Learn libraries [7, 8, 9]. The presented performance analysis highlights the advantages of using GPUs in training machine learning models, especially when processing extensive data volumes.

The developed algorithms were also tested on a dataset generated from the Buckley-Leverett mathematical model for predicting the oil recovery factor [10].

Based on the presented results, this article substantiates the importance and prospects of using graphics processors in the field of machine learning and highlights fundamental methods leading to improved performance and efficiency of algorithms.

2 Materials and Methods

Machine learning is a field of computer science that focuses on the development of algorithms and models that allow computers to learn from data and make predictions or make decisions without explicit programming. This field finds applications in various fields, including pattern recognition, data analysis, classification and forecasting.

Regression algorithms are used to solve the prediction problem when it is necessary to predict a continuous value based on input data. These algorithms build a model that describes the relationship between the input features and the output value. Examples of regression algorithms are linear regression, decision trees, support vector machine (SVM) and gradient boosting, each of which is suitable for different types of data and predictive tasks.

Scikit-learn, a widely used machine learning library for Python, provides powerful tools for applying regression algorithms. Among them is linear regression based on a model of linear dependence between the features and the target variable. Linear regression is often used in simple cases where a linear relationship between variables is assumed. Scikit-learn also provides an implementation of the Support vector Machine (SVM) for regression. In the case of SVR, the algorithm builds a hyperplane as far away from the data points as possible, and thus makes predictions. This method is effective when working with data, where it is difficult to identify linear patterns, and allows you to take into account nonlinear relationships.

Thus, Scikit-learn becomes an essential tool for machine learning specialists and researchers, providing user-friendly interfaces for learning, evaluating and selecting the best regression models depending on the specific requirements of forecasting tasks.

Scikit-Learn: LR and SVR algorithms

Linear regression in Scikit-learn is a model designed to describe a linear relationship between one or more features (independent variables) and a target variable (dependent variable) representing a continuous value.

Linear regression model

A linear regression model is a linear function that describes the relationship between input and output data. The visualization's example of linear regression model can be seen in Figure 1.

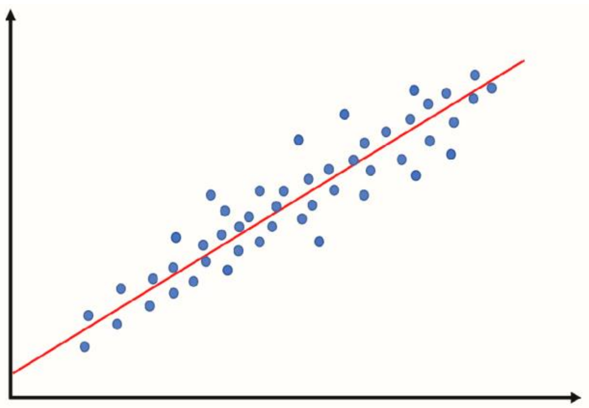

Figure 1 – Linear Regression (LR).

In general, the linear model looks like this:

$$y = w_0 + w_1 x_1 + w_2 x_2 + \ldots + w_n x_n,$$

where y – output value, w_0 – free term, w_i – weighting factor for the i-th feature, x_i – i-th attribute.

Weights in linear regression are used to account for the effect of each feature on the output feature. The weights of the linear regression model are determined during model training. The model learning algorithm minimizes the model error by selecting the optimal values of the weighting coefficients.

Support Vector Machine Regression (SVM) is a machine learning method that is used to predict continuous values based on a dataset with numerical features. SVR is an extension of the support vector machine, which is used for classification tasks.

The SVR model

The SVR model is a hyperplane that divides the feature space into two parts. The values of the output feature for points that are on one side of the hyperplane will be the same. The visualization's example of SVR model can be seen in Figure 2.

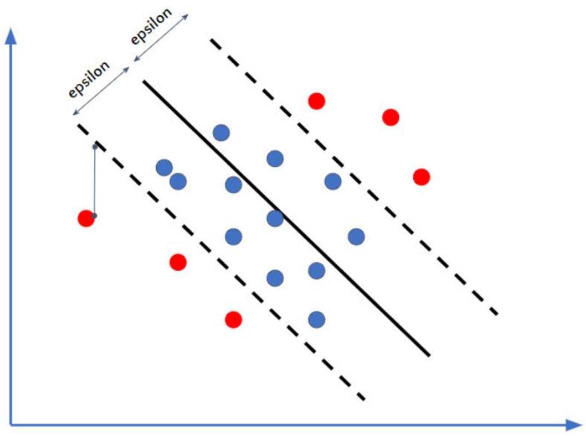

Figure 2 – Support vector regression (SVR).

The SVR model can be written as follows:

$$y = f(x) + \varepsilon,$$

where y – output value, x – input data, $f(x)$ – a function that describes the relationship between input and output data, ε – model error.

The function $f(x)$ can have different forms, depending on the method used to train the SVR model. The main goal here is to determine the boundary of the solution at a distance ε from the original hyperplane so that data points close to the hyperplane or being reference vectors are located within this boundary line.

RAPIDS: LR and SVR algorithms

RAPIDS is a set of open source libraries that accelerate machine learning and data analysis on the GPU. RAPIDS is based on the CUDA and cuda libraries and uses GPU capabilities to improve performance.

RAPIDS has the following libraries for use in machine learning:

- cuML – a library for machine learning on the GPU. It provides implementations of basic machine learning algorithms such as linear regression, logistic regression, k-means, k-nearest neighbors, etc.
- cuGraph – a library for processing graphs on the GPU. It provides tools for working with graphs, such as graph calculations, graph algorithms, etc.
- cuDF – a library for processing data on the GPU. It provides tools for working with data, such as data downloads, data transformations, calculations, etc.

RAPIDS cuML provides implementations of two main regression algorithms: linear regression (LR) and support vector machine regression (SVR).

Linear Regression is a function for training a linear regression model. This function takes as input a set of data with numeric attributes and a target attribute. The output result of the function is an object of the linear regression model

SVR is a function for training a regression model using the support vector machine. This function takes as input a set of data with numeric attributes and a target attribute. The output result of the function is an object of the regression model using the support vector machine.

The Linear Regression function and SVR use CUDA libraries to speed up GPU computing. CUDA is a set of libraries and tools that provide programmers with access to GPU hardware. CUDA libraries provide functions for performing various operations on the GPU, such as scalar operations, vector operations, and matrix operations. The linear regression model learning algorithm is also optimized for GPU computing. Special data structures are used that are optimized for accessing data from the GPU. In addition, special algorithms are used to speed up calculations.

predict is a function for predicting the values of a target feature for new data. This function takes as input an object of the linear regression model and a dataset with new data. The output result of the function is a set of predicted values of the target feature.

Implementation and Testing of Training Models using cuML

In this paper, two regression models based on the linear regression algorithm and the support vector machine (SVM) regression have been successfully implemented using the cuML library, specially designed for optimal performance on graphics processors (GPUs).

Implementations of LR and SVR using the cuML library in the RAPIDS environment on a graphics processor (GPU).

To begin with, random data was generated for the regression task using the make_regression function from the cuML library. The data was then converted to udf format for efficient processing on the GPU.

```
import cudf
from cuml.datasets import make_regression
from cuml.model_selection import train_test_split
n_samples = 80000
n_features = 20

X, y = make_regression(n_samples=n_samples, n_features=n_features, random_state=42)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
X_train = cudf.DataFrame(X_train)
y_train = cudf.Series(y_train)
X_test = cudf.DataFrame(X_test)
y_test = cudf.Series(y_test)
```

Then, instances of the LR and SVM model were created using the Linear Regression and SVM class from the cuML library, and the model was trained on the training data.

```
from cuml import SVR
from cuml.linear_model import LinearRegression as LR
modelLR = LR( fit_intercept = True, normalize = True, algorithm = 'eig')
modelLR.fit(X_train, y_train)
modelSVR = SVR(C=1.0, kernel='rbf', gamma=0.1)
modelSVR.fit(X_train, y_train)
```

After training the model, its performance is evaluated. Predictions were made on the test data, and metrics were calculated for an objective assessment of the quality of the trained model. The developed algorithms have been successfully tested on a dataset based on the Buckley-Leverett mathematical model designed to predict the oil recovery coefficient. Input parameters include porosity, absolute permeability, phase viscosity, and time iterations.

3 Results and Discussion

The experiments were conducted on a personal computer with a high-performance Intel Core i9-10900KF processor, 32 gb of RAM and a discrete NVIDIA RTX 3070 GPU with 8 gb of video memory. This configuration provides powerful computing resources and allows efficient use of libraries optimized for working with the GPU. The results of Linear Regression testing using the scikit-learn and cuML libraries on the RAPIDS platform are presented in Table 1 and Table 2:

Table 1 – Linear Regression Test Results: scikit-learn vs cuML

Number of instances	Scikit-Learn	RAPIDS cuML
300000	0.10	0.004
600000	0.22	0.007
1200000	0.568	0.012
2000000	0.933	0.012

There is a significant acceleration of the Linear Regression model when using cuML on the RAPIDS platform compared to Scikit-Learn. The effectiveness of cuML is especially evident when the amount of data increases. It is noticeable that cuML on RAPIDS is successfully scaling with an increase in the number of data instances. This indicates the high efficiency of processing large amounts of data using a GPU. With the time difference, especially at 2,097,120 instances, cuML confirms its outstanding performance in processing large data. The graph of the results presented in Table 1 is shown in Figure 3.

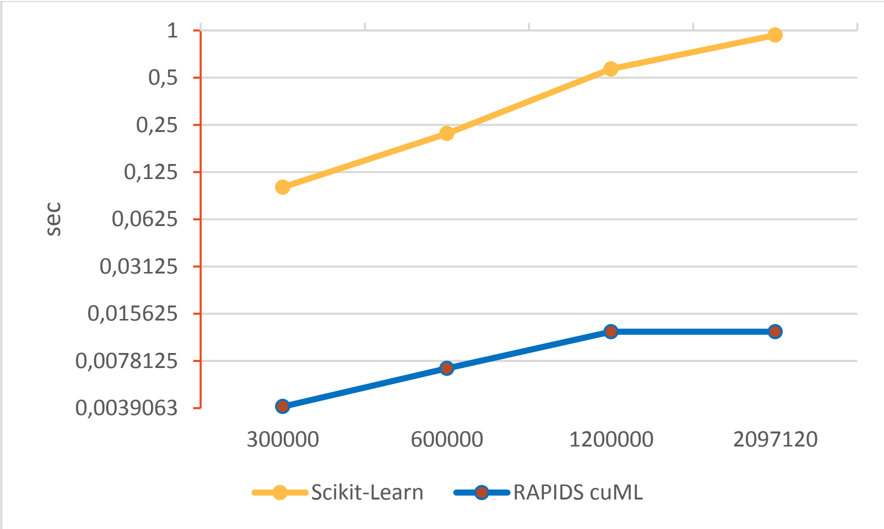

Figure 3 – The Effectiveness Of Linear Regression: Scikit-Learn vs RAPIDS cuML.

Table 2 – Test results of the Support Vector Machine (SVM): scikit-learn vs cuML

Number of instances	Scikit-Learn	RAPIDS cuML
10000	1.424	0.35
20000	5.795	0.62
40000	22.56	1.077
80000	83.9	3.562

The results clearly demonstrate a significant acceleration in the performance of the support vector machine (SVM) when using cuML on the RAPIDS platform compared to the scikit-learn library. The use of cuML is particularly advantageous when working with large amounts of data. With 80,000 instances, cuML demonstrates more than 20 times faster calculations compared to scikit-learn. The graph of the results presented in Table 2 is shown in Figure 4.

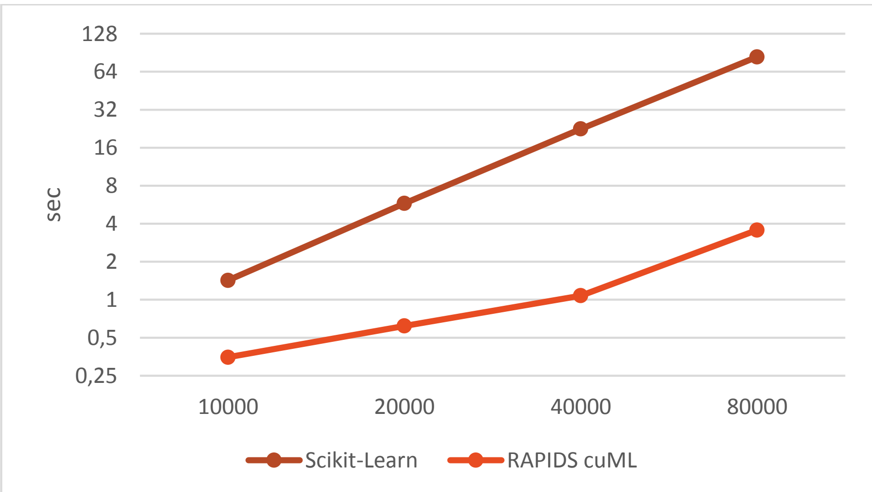

Figure 4 – The Effectiveness of the Support Vector Machine (SVM): Scikit-Learn vs RAPIDS cuML.

4 Conclusion

During the research, Linear Regression and Support Vector Machine (SVM) algorithms were developed and tested using the Scikit-Learn and cuML libraries on the RAPIDS platform. Experiments have been successfully conducted on data sets, including those generated from the Buckley-Leverett mathematical model for oil recovery factor prediction. Test results confirmed a significant acceleration of the algorithms when using cuML on the RAPIDS platform compared to the Scikit-Learn library. This is especially noticeable when processing large volumes of data, which demonstrates the outstanding performance of GPU computing.

Introducing GPU compute capabilities using RAPIDS and cuML opens new horizons in machine learning, delivering high performance and efficiency when processing big data. The results highlight the potential of these technologies to solve complex problems and provide a basis for further research and industrial applications.

Acknowledgments

This research was funded by the Science Committee of the Ministry of Science and Higher Education of the Republic of Kazakhstan, grant number BR18574136.

References

1. J. Lew et al., "Analyzing Machine Learning Workloads Using a Detailed GPU Simulator," 2019 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), Madison, WI, USA, 2019, pp. 151-152, doi: 10.1109/ISPASS.2019.00028.

2. Sara Rabhi, Wenbo Sun, Julio Perez, Mads R. B. Kristensen, Jiwei Liu, and Even Oldridge. 2019. Accelerating recommender system training 15x with RAPIDS. In Proceedings of the Workshop on ACM Recommender Systems Challenge (RecSys Challenge '19). Association for Computing Machinery, New York, NY, USA, Article 8, 1–5. https://doi.org/10.1145/3359555.3359564

3. S. M. Ghazimirsaeed, Q. Anthony, A. Shafi, H. Subramoni and D. K. D. Panda, "Accelerating GPU-based Machine Learning in Python using MPI Library: A Case Study with MVAPICH2-GDR," 2020 IEEE/ACM Workshop on Machine Learning in High Performance Computing Environments (MLHPC) and Workshop on Artificial Intelligence and Machine Learning for Scientific Applications (AI4S), GA, USA, 2020, pp. 1-12, doi: 10.1109/MLHPCAI4S51975.2020.00010.

4. RAPIDS.AI. 2019. RAPIDS.AI cuDF repository. https://github.com/rapidsai/cuDF

5. T. Hricik, D. Bader and O. Green, "Using RAPIDS AI to Accelerate Graph Data Science Workflows," 2020 IEEE High Performance Extreme Computing Conference (HPEC), Waltham, MA, USA, 2020, pp. 1-4, doi: 10.1109/HPEC43674.2020.9286224.

6. N. Becker et al., "Streamlined and Accelerated Cyber Analyst Workflows with CLX and RAPIDS," 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 2019, pp. 2011-2015, doi: 10.1109/BigData47090.2019.9006035.

7. G. Varoquaux, L. Buitinck, G. Louppe, O. Grisel, F. Pedregosa, and A. Mueller. 2015. Scikit-learn: Machine Learning Without Learning the Machinery. GetMobile: Mobile Comp. and Comm. 19, 1 (January 2015), 29–33. https://doi.org/10.1145/2786984.2786995.

8. Hao, J., & Ho, T. K. (2019). Machine Learning Made Easy: A Review of Scikit-learn Package in Python Programming Language. Journal of Educational and Behavioral Statistics, 44(3), 348-361. https://doi.org/10.3102/1076998619832248

9. Pedregosa, Fabian & Varoquaux, Gael & Gramfort, Alexandre & Michel, Vincent & Thirion, Bertrand & Grisel, Olivier & Blondel, Mathieu & Prettenhofer, Peter & Weiss, Ron & Dubourg, Vincent & Vanderplas, Jake & Passos, Alexandre & Cournapeau, David & Brucher, Matthieu & Perrot, Matthieu & Duchesnay, Edouard & Louppe, Gilles. (2012). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 12.

10.Kenzhebek, Yerzhan, Timur Imankulov, Darkhan Akhmed-Zaki, and Beimbet Daribayev. “Implementation of Regression Algorithms for Oil Recovery Prediction.” Eastern-European Journal of Enterprise Technologies. Private Company Technology Center, April 30, 2022. https://doi.org/10.15587/1729-4061.2022.253886

ISSN 2958-0846 eISSN 2958-0854 Journal of Problems in Computer Science and Information Technologies №4 (1) 2023 https://jpcsip.kaznu.kz

IRSTI 28.23.29 https://doi.org/10.26577/jpcsit2023v1i4a6

G.H. Aimal Rasa[1], Z.M. Abdiakhmetova[2*]

[1]Kabul Education University, Kabul, Afghanistan
[2]Al-Farabi Kazakh National University, Almaty, Kazakhstan
*e-mail: zukhra.abdiakhmetova@gmail.com

STUDY OF SIGNS OF IMPACT ON THE QUALITY OF EDUCATION BY ML

Abstract. Use of machine learning (ML) algorithms to analyze and identify signs that affect the quality of education opens up new opportunities for individualization of education, optimization of educational processes and improvement of educational results of students.

The personalization of education stands as a paramount trend in contemporary learning. Each student possesses distinct requirements, passions, and talents. By scrutinizing the factors impacting educational excellence, we can pinpoint individual elements that wield substantial influence over each student's success. Consequently, this enables the crafting of customized educational programs and techniques finely tuned to the unique needs and aptitudes of every student.

The objective of this endeavor is to construct a system employing machine learning algorithms that can discern the factors influencing the assessment of students' educational quality. These facets render the research pertinent and noteworthy within the landscape of modern education. It offers a gateway to a deeper comprehension of the learning processes, streamlining educational procedures, and ultimately yielding improved outcomes in student education.

Key words: Machine Learning Algorithm, Support Vector Method, Random Forest, Dataset, Linear Regression.

1. Introduction

Currently, the process of digitalization of society is developing very quickly. This, in turn, simplifies many tasks encountered in everyday life. Of course, this phenomenon finds its place in education area and affects the development of this field. Lydia Sandra, Ford Lambangaol, and Tokuro Matsuo [1] reviewed the factors that may affect student performance and technologies that help predict student performance.

The study delved into several inquiries, one of which aimed to identify key attributes for forecasting student achievement. The findings revealed that both internal assessment and summative assessment emerged as the most consistent indicators for predicting academic performance. Additionally, the study highlighted other significant factors such as personal and self-assessment, prior academic records, extracurricular involvement, and social characteristics.

One of the primary objectives of this research was to explore the application of machine learning algorithms in forecasting student performance. The study was conducted in May 2021, and the data was sourced from the IEEE Access and Science Direct databases, adhering to standard database search protocols, including exclusion and inclusion criteria and search result analysis. The study employed a range of algorithms, including logistic regression, Bayesian methods, K-nearest neighbor (KNN), regression trees, random forest, decision trees, long-term and short-term memory (LSTM), support vector machines (SVM), multi-layer perceptron neural networks (MLP), artificial neural networks (ANN), and reference vector methods, with all of them demonstrating promising results [2].

Another research study on this topic [3] highlights the challenge of predicting student performance due to the vast volume of data in educational databases. The primary objective of this research is to review the use of artificial intelligence systems for predicting academic learning.

The research paper presents an automated system for evaluating students' progress and analyzing their achievements. Here, the author uses a tree algorithm to accurately predict student performance. The data clustering method was used to analyze a large set of student databases. This method speeds up the search process and provides accurate classification results. A new model of teaching is proposed using information about a student obtained during college registration. The final data sets are entered so that Machine Learning algorithms can use them and predict student performance. During the research, 13 algorithms were selected from Machine learning algorithms in 5 categories: Bayesian, SVM, MLP, IBK, linear regression and tree-type algorithms. As a result of the study, the use of binomial logistic regression gave 97.06% accuracy, decision tree 88.24%, entropy 91.18%, K-nearest neighbor showed 93.72% accuracy. Among the machine learning algorithms, binomial logistic regression performed best.

The research paper in question [4] emphasizes that a nation's economic prosperity hinges on the accessibility of higher education, a concern at the forefront of any government's agenda. Hussein Altabrawi's insights further underscore that the surge in student loan debt in America is partly attributable to delayed graduation rates. To construct the machine learning model, a fully connected artificial neural network, Bayesian algorithm, decision tree, and logistic regression were employed.

The dataset used to develop these models was compiled from students at the College of Humanities during the 2015 and 2016 academic years, sourced from a combination of student surveys and test books. This dataset encompasses information pertaining to 161 students. The research activities encompassed the creation of a student dataset, data collection, data preprocessing, the construction and evaluation of four machine learning models, identification of the best-performing model, and a thorough analysis of the results.

The dataset is comprehensive in nature, encompassing factors ranging from the students' age and gender to the extent of family involvement in their education. The effectiveness of these four models in relation to the dataset was assessed using the ROC index, with the artificial neural network achieving a value of 0.807, Bayesian at 0.697, decision tree at 0.762, and logistic regression at 0.767. Notably, the artificial neural network emerged as the most effective model [5].

2. Literature review

The next work [6] was conducted in order to develop and compare different Machine Learning algorithms to predict students' academic achievement. The work was published in the journal Computers & Education and is widely recognized in the field of education and information technology.

The work used the data of students with the following characteristics:

- Data on student demographics such as age, gender, mother tongue and nationality;
- Information about the activity of students on the learning platform, for example, the number of visits, the duration of sessions and the time spent on tasks;
- Results of tests and assignments, including scores, quizzes, work grades, and number of correctly solved assignments.

Another work in this direction [7] describes the use of various machine learning methods to predict student achievement. The paper describes the various indicators that can be used to predict student performance, as well as the advantages and disadvantages of using machine learning in this field.

In their article [8], authors Muneer Ahmad Al-Radaideh, Ghassan Issa, and Mohammed Al-Zyoud employed their dataset to forecast the academic performance of students enrolled at a Jordanian university. To achieve this objective, they applied various machine learning algorithms and determined the most efficient one for this specific task.

The outcomes of their research demonstrated that Machine Learning techniques can indeed be highly effective in predicting students' academic success. The researchers employed several Machine Learning algorithms, including decision trees, Bayesian classifiers, and the support vector method (SVM), and conducted a comparative analysis of their performance. The results revealed that the SVM algorithm outperformed the others, achieving a prediction accuracy rate of over 90%. Moreover, the researchers identified key factors such as prior exam GPAs, student age, the number of subjects studied, and the presence of family support as the most influential in determining academic achievement. Consequently, this study provides further evidence of the feasibility of employing machine learning methods to predict students' academic success and pinpoint significant factors that impact their educational accomplishments.

Researchers in their work [9] used machine learning methods to predict students' final grades. To do this, they used data on the academic performance of 600 students at a Portuguese university. Information on socio-economic characteristics such as Student's study schedule, number of absences, number of study materials available to the student, age and gender of the student, as well as parents' education and access to the Internet at home were used as indicators of analysis. The researchers used several machine learning algorithms, including logistic regression, random forest, and gradient boosting, to predict students' final grades. They also compared the effectiveness of machine learning techniques with traditional methods such as linear regression and multiple regression.

The results showed that machine learning methods are more effective in predicting students' final grades than traditional regression methods. The researchers also found that the most important predictors of student achievement were internet access at home, the student's study schedule, and absenteeism.

In a research paper [10], scholars conducted a comparative investigation of various machine learning algorithms for the purpose of forecasting student performance. The study employed a dataset comprising the academic records of 649 engineering students at an Indian university. The dataset included information regarding students' online learning activities, encompassing metrics such as the number of lectures viewed, tasks completed, responses to test questions, and the average time taken to answer those questions. Additionally, the study considered background characteristics such as students' age, gender, and geographic location as factors for analysis. The researchers assessed multiple machine learning algorithms, including logistic regression, multiple linear regression, decision trees, random forest, and gradient boosting, to predict student performance. They also compared the efficacy of these algorithms and identified the most crucial attributes for forecasting student success. The findings revealed that gradient boosting emerged as the most effective algorithm for predicting student performance, exhibiting robust performance across diverse data configurations.

In a separate study [11], researchers evaluated the effectiveness of different machine learning algorithms in predicting student performance using the «student performance dataset,» which contains data on student achievements in mathematics and Portuguese language courses. This dataset incorporates various aspects, including demographic information, academic background details, family-related data, and student behavior patterns.

Furthermore, the work presented in [12] constitutes a case study aimed at predicting the progress of engineering students through the application of Machine Learning algorithms. The authors gathered data from 235 engineering students attending a private university in India. This dataset encompassed a wide array of features, such as student demographic data, progress records, attendance records, and assignment grades. The researchers employed various Machine Learning algorithms, including decision trees, random forests, and support vector machines, to forecast student scores.

In the research conducted by M. Abdullahi and S. K. Abdullah in their work [13], they explored the use of different machine learning algorithms. Their study was based on data collected from the University's information system in Iran and aimed to predict student performance in higher education. Various indicators were utilized in their predictive models:

- Average number of hours spent studying per week.
- Average score in mathematics, physics, chemistry, English subjects at school.
- Student course at the university.
- The size of the student study group.
- Student status (for example, whether the student is an athlete or not).
- Status of the student in relation to military duty.
- The student's employment status (for example, whether the student is employed or not).
- The average score of the student in the previous semesters at the university.

The authors employed the following Machine Learning algorithms to forecast student performance: Random Forest, Gradient Boosting, K-Nearest Neighbors (KNN), Support Vector Machines (SVM), and Decision Trees.

One of the primary strengths of this research lies in its utilization of real data extracted from the University's information system in Iran. This ensures that the findings have practical applicability and can be instrumental in enhancing the higher education process. An article by V. Nair, S. Bhatia, and D. Dey in 2021 [14] offers an overview of numerous studies utilizing machine learning techniques for predicting student performance. In their publication, the authors scrutinized 66 articles published from 2010 to 2020 and identified common indicators utilized across these studies to forecast student performance.

However, despite the significance of this research, it does have some limitations, notably a relatively modest feature set and a lack of comparisons between machine learning methods.

In another research paper [15], the authors conducted an extensive literature review to assess existing machine learning approaches for predicting student performance in educational institutions. They compared various machine learning techniques employed in research. The authors incorporated diverse indicators for forecasting student performance, encompassing demographic information, educational data, test scores, student attendance and engagement data, among others. The incorporation of such indicators enables more accurate predictions of student progress and the identification of factors influencing their academic accomplishments.

In [16], the authors proposed a machine learning model for predicting student performance using student data. They explored various machine learning techniques, including decision trees, random forest, logistic regression, and the support vector method. The analysis utilized data from 183 students, encompassing various course-related characteristics such as age, gender, ethnicity, prior education's grade point average, school type, number of subjects, completion and duration of standardized math tests, and the number of lectures and exercises. The results demonstrated the effectiveness of machine learning models in predicting students' performance in higher education, particularly when employing algorithms like random forest and the support vector method.

Overall, this paper constitutes a valuable contribution to the field of utilizing machine learning models to predict student performance in higher education. However, it is essential to acknowledge some limitations and drawbacks of this work.

In an article by S. N. Tiwari and A. K. Misra [17], the authors explored the potential of machine learning techniques in predicting student performance in online courses (MOOCs). They considered four distinct machine learning algorithms: decision tree, random forest, nearest neighbor method, and logistic regression. Indicators derived from student activity on the MOOC platform, such as time spent in the course, assignments completed, forum participation, etc., were employed. Additional indicators, such as word count in task responses and video lecture views, were also utilized. The research utilized data from the course «Introduction to Computer Science and Programming Using Python» on the edX platform, involving a total of 1972 students. The results indicated that the random forest method achieved the highest accuracy with an F1 score of 0.85, while logistic regression yielded the lowest precision with an F1-measure of 0.72. The authors concluded that Machine Learning techniques can serve as an effective tool for predicting student performance in MOOCs and recommended the random forest approach for this task.

In the subsequent article [18], B. Kabir and H. Ali conducted a systematic review of literature concerning the utilization of machine learning algorithms for predicting student performance in higher education. Their review encompassed 65 articles published between 2015 and 2020. The authors utilized the following indicators to forecast student performance:

Demographic data (gender, age, race, etc.).

Academic data (grade point average, exam grades, credits, etc.).

Behavioral data (frequency of visits, time spent in the course, number of materials visited, etc.).

Social data (activity in social networks, communication with peers and teachers, etc.).

The authors compared various Machine Learning algorithms used for predicting student performance, including logistic regression, decision trees, support vector methods, random forests, and neural networks. They also assessed prediction quality using metrics such as accuracy, completeness, F-measure, and ROC curve.

3. Material and Methods

Collecting Data

In total, the data set contained 21 characters. Labels in the data set:

- Gender (binary: 1-male, 0-female)
- Age characteristics (number: from 15 to 17 years old)
- Is your school near your home? (binary: 1-yes, 0-no)
- Is your family complete? (binary: 1-yes, 0-no)
- Your mother's education?
- Your father's education? (numerical value: 1-medium, 2-unfinished higher, 3-higher, 4-higher post-graduate professional)
- Your mother's job? (binary: 1-employed, 0-unemployed)
- Do your parents do work that requires physical effort? (binary: 1-yes, 0-no)
- Your father's job? (binary: 1-employed, 0-unemployed)
- Is there internet connection at home? (binary: 1-yes, 0-no)
- What do you do in your spare time? (binary: 1-useful, 0-useless)
- Do you meet your friends often? (binary: 1-yes, 0-no)
- Do you eat well? (binary: 1-yes, 0-no)
- What is your phone's operating system? (binary: 1-Ios, 0-Android)
- Do you have a close relationship with students? (binary: 1-yes, 0-no)
- Are you satisfied with your social situation? (binary: 1-yes, 0-no)
- Can the teacher interest you in his subject? (binary: 1-yes, 0-no)
- Does the teacher evaluate fairly (competently)? (binary: 1-yes, 0-no)
- G1- previous term grades (number: 3 to 5 years)
- G2-prior term grades (numerical: 3 to 5 years)
- G3-estimated value (number: 3 to 5 years)

On the basis of the selected signs, a survey was conducted among the upper classes of the specialized lyceum named after Al-Farabi. Data attributes include student grades, demographic, social, and school characteristics. These data were collected through school reports and questionnaires. Indicators include the age and gender of the students, the occupation of the parents, and a healthy daily diet. A total of 505 students participated in the survey. 234 of them are men, 271 are girls. In the dataset, the textual data were converted to binary and numeric data for use in calculations. Attributes with a binary value (gender, marital status, completeness of the family, parental work, frequent meetings with friends, internet access, healthy nutrition, teacher's interest in his subject and competent assessment) were changed to 0 and 1 according to the response values. Also, the education of the student's parents was numerically divided into 4 different values. They are as follows: 1-secondary, 2-incomplete higher, 3-higher, 4-higher post-graduate professional.

After transforming the data, we need to classify it. In the following program code, we divide the data into two parts: the labels and the target variable.

4. Processing of data collected during pedagogical practice using machine learning algorithms

After converting, classifying and grouping our data, we proceeded to our calculations using 3 selected algorithms. We used the Random Forest method for the first time. The next method used in the calculation is the linear regression method. The linear regression method is widely used in forecasting and classification. The advantage of the linear regression method over other methods is its simplicity and the results can be easily interpreted.

The last method used in forecasting is the method of reference vectors or the method of support vectors. This code is used to train a model using support vectors (SVM) on the training data (x_train and y_train).from sklearn.svm import SVC: This line of code imports the SVC (Support Vector Classifier) class from the SVM module into the scikit-learn (sklearn) library. SVC is used for classification problems using the reference vector method.

5. Analysis of the results obtained from the experiment

The accuracy of each algorithm was calculated using the above MSE (Mean Squared Error), MAE (mean Absolute Error) and R2 (R-squared) indicators. According to the results of these evaluation indicators, the mean square error of the linear regression method is 0.708, the mean absolute error is 0.722, R-squared is 0.025, the mean square error of the reference vector method is 0.712, the mean absolute error is 0.701, the R-squared is 0.014, the random forest method mean square error was 0.813, mean absolute error was 0.781, R-squared was 0.176. Accordingly, among the methods, the method of reference vectors showed the best results (Figure 1).

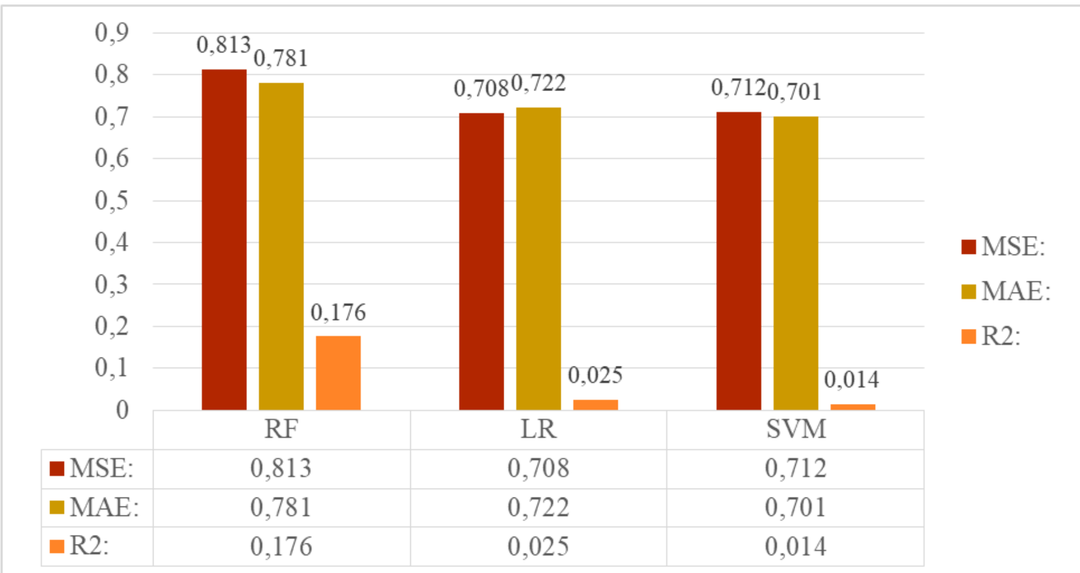

Figure 1 – Comparing of methods

Also, among the symptoms relevant to the student, the prediction and the magnitude of the symptoms that affected the

Among the characteristics, characteristics such as the student's age, parents' education, free time and grades of the previous term were found to be more influential than others.

The main features were found to be the student's age, parents' education, free time and grades of the previous term (Figure 3).

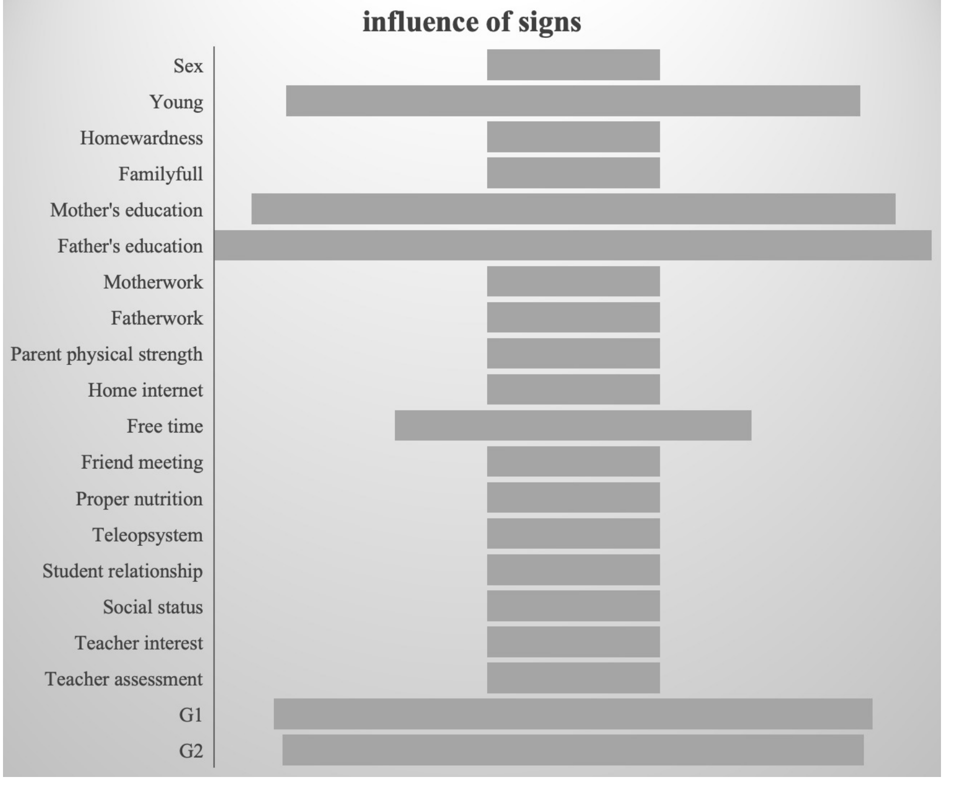

Figure 2 – Influence of Signs

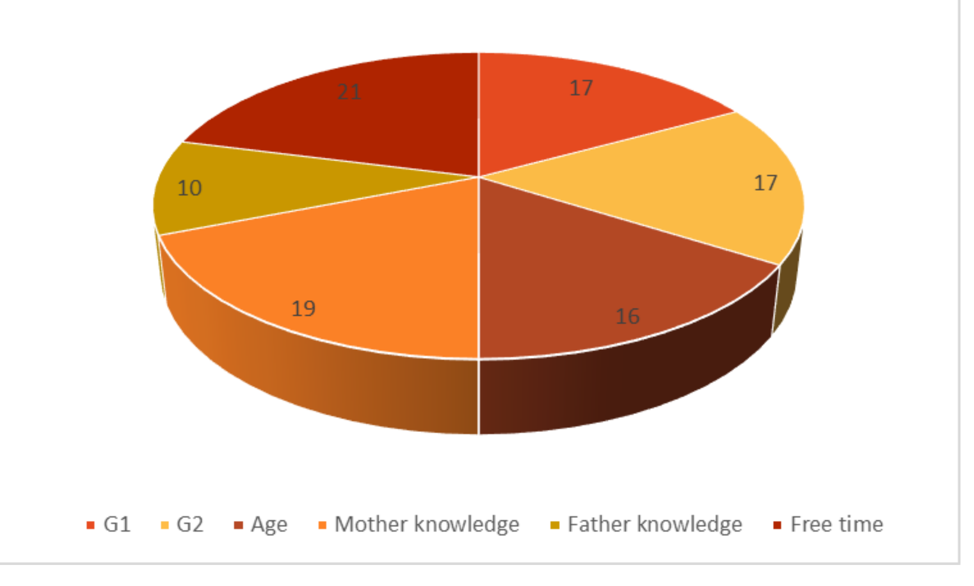

Figure 3 – Main Features of dataset

The selection of these signs has its own reasons and features.

6. Conclusion

Throughout the course of our investigation, we examined the factors influencing the quality of students' education. Our initial objective involved identifying the characteristics relevant to the utilization of machine learning algorithms within the realm of education. We conducted comprehensive literature reviews to explore existing studies that had employed machine learning algorithms in educational contexts. Subsequently, we engaged in research and analysis of these machine learning algorithms.

In the practical part of the research work, a survey was conducted on 18 signs in Al-Farabi specialized lyceum. The second task was to use symbols and machine learning algorithms in calculations. Calculations were made on the data set collected on the basis of the survey by the method of linear regression, random forest, and support vectors. Students' grades were estimated. We used MSE (Mean Squared Error), MAE (mean Absolute Error) and R2 (R-squared) indicators to assess the accuracy of the methods.

According to the evaluation, the mean square error of the linear regression method is 0.708, the mean absolute error is 0.722, R-squared is 0.025, the mean square error of the reference vector method is 0.712, the mean absolute error is 0.701, R-squared is 0.014, the mean square error of the random forest method is 0.813, average absolute error – 0.781, R-squared – 0.176. Accordingly, among the methods, the method of support vectors showed the best result by 6%. Among the signs, the signs that had the greatest influence on the prediction were determined to be the student's age, parents' education, free time and grades of the previous term.

In general, it can be concluded that the conducted research Machine learning algorithms allow to effectively analyze the signs that affect the quality of education of learners and to predict grades. We believe that it will be a useful tool for educational institutions and teachers in developing individual learning approaches and improving educational outcomes.

References

1. Lidia Sandra, Ford Lumbangaol, Tokuro Matsuo Machine Learning Algorithm to Predict Student's Performance: A Systematic Literature Review, Vol. 10, Issue 4, – 2021. – P. 1919-1927.
2. J. Dhilipan, N. Vijayalakshmi, S. Suriya, Arockiya Christopher Prediction of Students Performance Using Machine learning, – 2021 IOP Conf. Ser.: Mater. Sci. Eng. 1055 012122 DOI 10.1088/1757-899X/1055/1/012122
3. Hussein Altabrawee, Osama Ali, Amir Qaisar Predicting Students' Performance Using Machine Learning Techniques, Journal of University of Babylon for Pure and Applied Sciences 27(1):194-205 DOI:10.29196/jubpas.v27i1.2108
4. Y. Meier, J. Xu, O. Atan and M. Van Der Schaar, "Predicting grades," IEEE Trans. Signal Process, – 2016. – Vol. 64, – P. 959-972.
5. P. Guleria, N. Thakur, and M. Sood, "Predicting student performance using decision tree classifiers and information gain," Proc. 2014 3rd Int. Conf. Parallel, Distrib. Grid Comput. PDGC, – 2015. – P. 126-129.
6. Yıldız, M.B., Börekçi, C. (2020). Predicting Academic Achievement with Machine Learning Algorithms. Journal of Educational Technology & Online Learning, 3(3), 372-392.
7. P. Kaur et al. A review of machine learning techniques for student performance prediction. Conference: 2021 International Conference on Computer & Information Sciences (ICCOINS) – 2018. – 7p.
8. M. Al-Radaideh et al. Predicting Student Academic Performance Using Machine Learning Algorithms. –TEM Journal. Volume 10, Issue 4, Pages 1919-1927, ISSN 2217-8309, DOI: 10.18421/TEM104-56
9. J. M. Banda et al. Using Machine Learning Algorithms to Predict Final Grades of Students. Bakolori Journal of General Studies Vol. 12 No. 2 – 2019.
10. A. K. Sharma and A. K. Chaturvedi. A Comparative Study of Machine Learning Algorithms for Predicting Student Performance. International Journal of Intelligent Systems and Applications 11(12):34-45 DOI:10.5815/ijisa.2019.12.04
11. A. Mittal and N. Jindal. An Analysis of Machine Learning Techniques for Student Performance Prediction. *Educ. Sci.* 2021, *11*(9), 552; https://doi.org/10.3390/educsci11090552.
12. Al-Alawi, L., Al Shaqsi, J., Tarhini, A. et al. Using machine learning to predict factors affecting academic performance: the case of college students on academic probation. Educ Inf Technol (2023). https://doi.org/10.1007/s10639-023-11700-0
13. M. Abdollahi and S. K. Abdullah. Comparison of Machine Learning Algorithms for Student Performance Prediction in Higher Education. International Journal of Computer Sciences And Engineering 7(4):721-725

14. V. Nair et al. Prediction of Student Performance Using Machine Learning Algorithms: A review. – Proceedings of the International Conference on Applications of Machine Intelligence and Data Analytics (ICAMIDA 2022) (pp.735-741)

15. Albreiki, B.; Zaki, N.; Alashwal, H. A Systematic Literature Review of Student' Performance Prediction Using Machine Learning Techniques. Educ. Sci. 2021, 11, 552. https://doi.org/ 10.3390/educsci11090552

16. R. Rasheed et al. Predicting Student Performance in Higher Education using Machine Learning Algorithms. Journal of University of Babylon, Pure and Applied Sciences, Vol.(27), No.(1): 2019.

17. S. N. Tiwari and A. K. Misra. Predicting Student Performance in MOOCs using Machine Learning Techniques. – Conference: 11th International Conference on Data Mining, Computers, Communication and Industrial Applications (DMCCIA-2017) Kuala Lumpur (Malaysia)

18. Alyahyan, Eyman & Dustegor, Dilek. (2020). Predicting Academic Success in Higher Education Literature Review and Best Practices. International Journal of Educational Technology in Higher Education. 17. 10.1186/s41239-020-0177-7.

IRSTI 28.23.29 https://doi.org/10.26577/jpcsit2023v1i4a7

N. Azatbekuly*, A. Mukhanbet

Al-Farabi Kazakh National University, Almaty, Kazakhstan
*e-mail: nurtugang17@gmail.com

OPTIMIZED INTELLIGENT MODULES FOR VIDEO SURVEILLANCE AND MONITORING

Abstract. Platform of optimized intelligent modules, designed for intelligent video surveillance and control systems, is an outstanding tool for implementing effective security measures and improving monitoring processes. This paper examines the development and optimization of several important modules, including a facial recognition system, line intersection detection algorithms, methods for detecting the presence of a person in specified zones and an algorithm for searching for people. The most productive algorithms, such as Faster R-CNN (Faster Region-based Convolutional Neural Network), YOLO (You Only Look Once), SSD (Single-Shot Detector), were researched, thereby work was carried out to improve and optimize the YOLO algorithm. In addition to the use of this algorithm, optimizations were carried out, including image processing methods (including scaling) and a frame skipping mechanism using parallel computing, which significantly reduced the computational load. The resulting platform provides users with the ability to effectively monitor and analyze video streams, automatically identify potential threats and events, which makes it the optimal solution for ensuring security in a variety of applications, including public places, enterprises and critical infrastructure facilities. The results of this paper provide new prospects for improving video surveillance and control systems, contributing to an increase in the level of security and efficiency of actions.

Key words: Intelligent Surveillance, CNN, YOLO, Optimization, Multithreading.

1. Introduction

The development of an intelligent video surveillance and control system that has the ability to function effectively with limited computing resources is an urgent paper in modern information technology. The present time is characterized by an increase in the volume of video data and growing requirements for the performance of video surveillance systems, which is dictated by both the sphere of public security and corporate interests. It was found that the creation of a system that does not resort to excessive computing resources, considering modern technological requirements, is a challenge. It is important to take into account that such a system should ensure stable operation, free from failures and errors.

There are several papers on a similar topic. For example, in [1] good results have been achieved using the high-level TensorFlow library, in [2] there is good information about main issues related to the intelligent monitoring systems used in public spaces.

Based on the complexity of the task, in this paper we pay attention to the optimization of algorithms and models of computer vision by manipulating images. This technique is designed to reduce the load on computing resources while maintaining the accuracy of the system. In addition, an important stage in the development of such systems is the use of frame skipping techniques, which contributes to the rational use of computing power. This paper offers a detailed analysis of the process of developing and optimizing an intelligent video surveillance and control system, highlighting the key aspects, methodology and achievements that allowed us to solve complex problems and create a system that combines high efficiency and optimal use of resources. The problems discussed in this paper are partially described in [3].

2. Methodology

This part will show our research and opinions on our topic in the context of the developed optimized modules, coupled with the theory and justification of the methods themselves.

2.1. Justification of the choice of the YOLO algorithm

YOLO (You Only Look Once), being a super-high-level tool, is not deprived of poor quality of work or inefficient use of resources [4]. Surface studies and tests of models built on PyTorch or Tensorflow frameworks were carried out, the results and quality of which were much inferior to the ready-made YOLO object detection models [5]. Thus, this cluster of ready-made models is suitable for using them in production. But the existing problems have not gone away, its customized solutions are given in this paper.

The task was mainly based on the optimization of human detection, so other algorithms such as Faster R-CNN (Faster Region-based Convolutional Neural Network) [6] and SSD (Single-Shot Detector) were not used, due to a significantly lower detection rate than the YOLO algorithm (an algorithm with a single-stage approach in architecture). For detection in almost all cases in the environment of our developed modules, the accuracy of the fast YOLO algorithm of the last eighth version turned out to be quite enough. As befits any convolutional neural network, YOLO has the same 2 main stages: feature extraction and detecting, in a single-stage form [7]. Here are the main points highlighted in the course of the research in the YOLO architecture (see Figure 1):

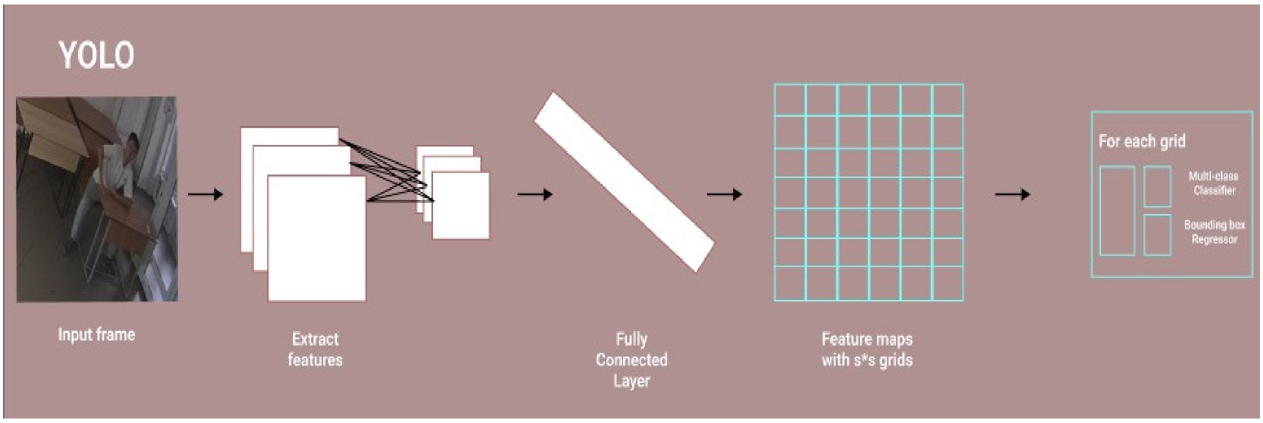

Figure 1 – Progress of post-traumatic stress disorder of personality

2.2. Optimization with multithreading

Complex and resource-intensive tasks of working with a video stream can also be solved live due to multithreading. The methodology for optimizing deep-learning algorithms using a parallel approach and potential directions for parallelism in deep learning are described in [8]. Work in this paper describes a solution that is a simple implementation for the capture and processing of video frames in a separate stream, with callback functions and blocking and non-blocking frame extraction options. In particular, the following aspects of the implementation and application of multithreading in working with video streams and processing it with the YOLO framework should be considered (see Figure 2).

For efficient frame processing in the YOLO algorithm, the "FreshestFrame" class was developed using multithreading. Consider the key elements of its implementation include:

1. Initialization. When creating an instance of the class, the variables necessary for working with video capture and multithreading are set. A blocking condition ("self.cond") is supported, which ensures waiting for a new frame to appear.

2. Starting A Stream. When the stream starts ("self.start()"), an infinite loop of reading frames from the video capture begins. Each frame is published using a lock condition, which allows other threads to wait for a fresh frame to appear.

3. Reading and Processing Frames. In the "run()" method, frames are read from the video capture, and their state variables are updated. If there is a callback ("self.callback"), the corresponding frame processing is performed.

4. Reading Frames from the Main Stream. The "read()" method allows the main thread to read frames. When using the lock condition, it is blocked until a new frame appears. You can set a sequential frame number ("seqnumber') to lock until the specified number is reached.

5. Stop and Release. When the "release()" method is called, the stream is stopped and video capture resources are released.

6. Main Stream. The main thread continues to perform other tasks without being blocked while waiting for new frames.

The use of multithreading in the "FreshestFrame" class ensures efficient and continuous processing of the video stream, which is especially important for the YOLO algorithm, which requires fast analysis of multiple frames. The above optimization using multithreading gives the following results in the development of intelligent modules for video surveillance (see Figure 3):

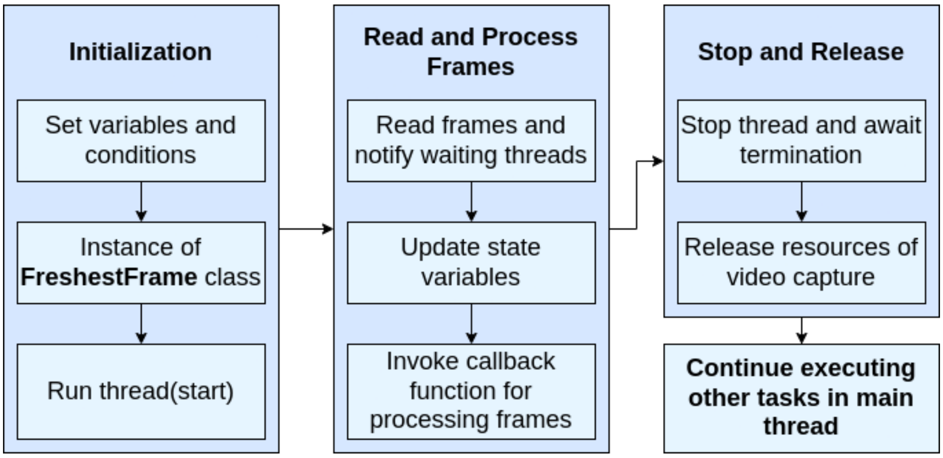

Figure 2 – Schematic representation of multithreading routine on reading and processing frames

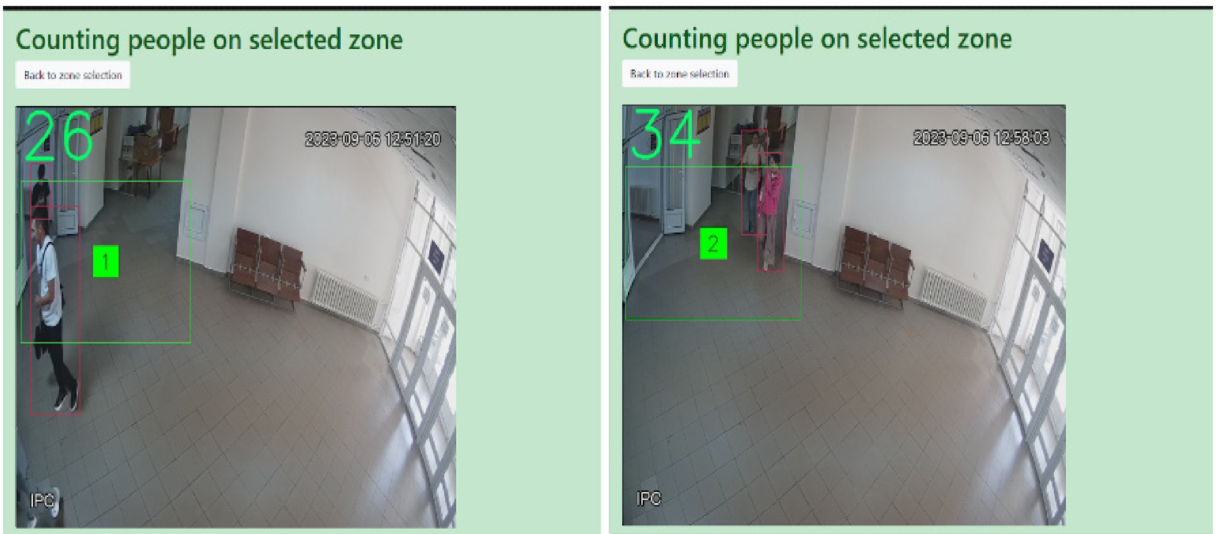

Figure 3 – FPS improvement after optimization with multithreading

2.3. Optimization with image manipulation

Image manipulation techniques play a critical role in optimizing and improving the performance of an integrated video surveillance and control system [9] [10]. In this article, image manipulation techniques have been developed and applied, aimed at improving the accuracy and reliability of video stream analysis in a variety of conditions. The use of these methods has had a significant impact on the effectiveness of the system. Let's look at a few key methods of image manipulation:

Increasing the size of detected objects: In cases when the objects on the video stream are small or their image is not clear enough, a method of increasing the size of detected objects has been implemented. This method allows you to more accurately extract features and provide more stable recognition, even with limited visibility.

Changing contrast and brightness: Analyzing a video stream in different light conditions can be a challenge. To solve this problem, methods of changing the contrast and brightness of images were applied. This helped to better highlight objects in the background and improve overall clarity.

Noise Filtering: Video data may contain noise and artifacts that may affect detection accuracy. Filtering techniques such as median filter or Gaussian filter have been applied to reduce the effect of noise to improve image quality.

One example of the use of frame manipulation methods is the face recognition module, in which all the listed techniques were applied. Before direct face recognition, a person is detected by one of the YOLOv8 detection models, then the frame with the person is enlarged and the listed filters are applied to restore the enlarged and fuzzy image. After all these manipulations, a face will be recognized using face_recognition (see Figure 4).

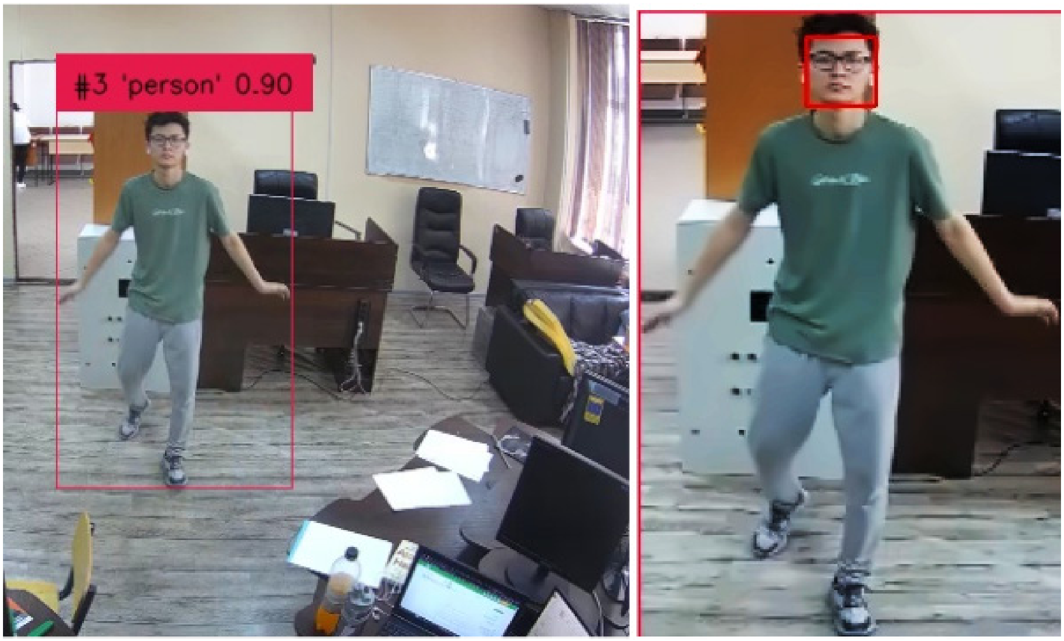

Figure 4 – Face detecting improvement after frame optimization

3. Performance evaluation

In this part, in order to visualize the optimized results obtained, FPS indicators were compared, as well as the accuracy of the modules was visually checked.

3.1. Performance comparison

Coupled with the included local Django server and other manipulations with the video stream that was captured by the open-cv module, which gradually but not significantly reduce the frames per second, several tests were conducted on the performance of the models (see Figures 4, 5). But it should also be noted that although the provided models from the YOLO – versions tiny, small, medium, large, extra-large, promise to be more accurate as the model becomes more complex (increasing the number of layers and, accordingly, parameters), these tests show slightly unexpected results, at least for human recognition. This applies to medium and large models, which did not justify the promised high-performance indicator in practice, being approximately on par with the small model. And of course, the fact remains true that the value of frames per second varies depending on the workload of processing the video stream (specifically on the number of people). The following tests were made on the basis of the NVIDIA GeForce GTX 1660 SUPER GPU with 6GB dedicated RAM, with 16 GB RAM. The video stream from the camera with a resolution of 640 by 544 (348160 pixels) was processed.

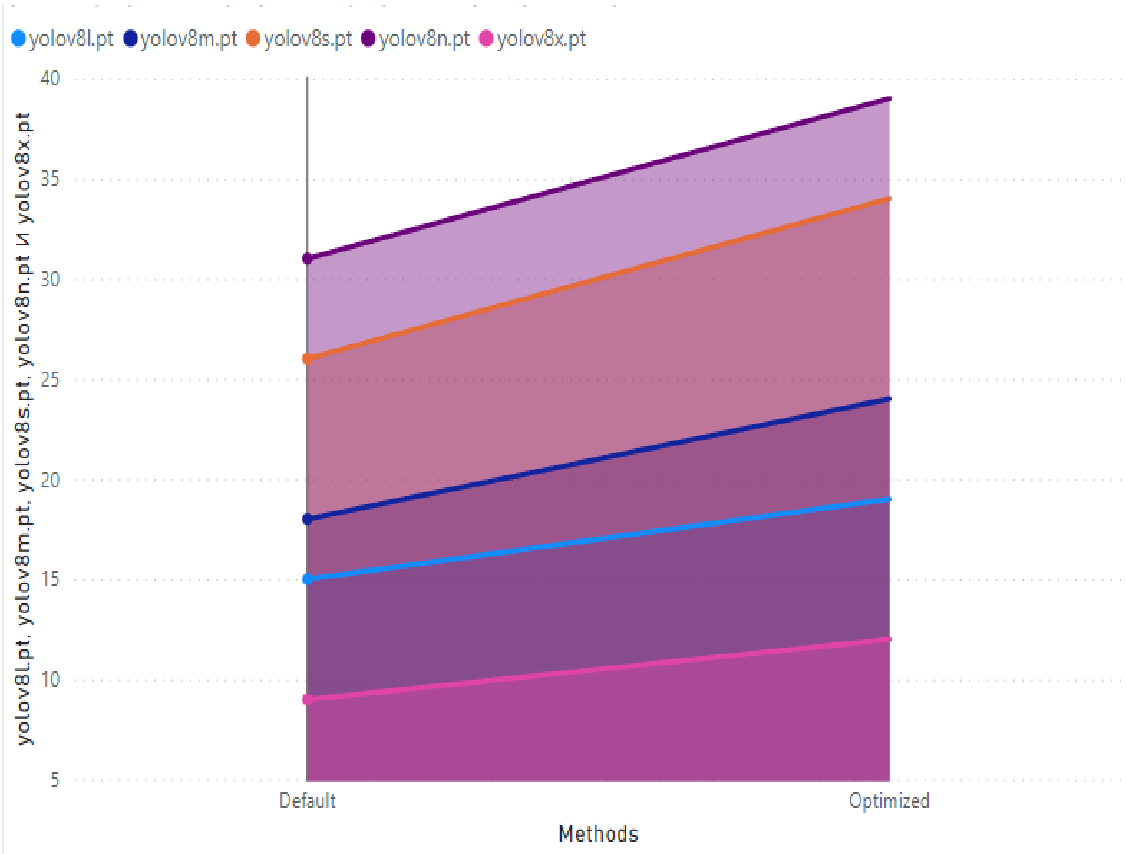

Figure 5 – Comparison of FPS rate of default and optimized versions

3.2. Comparison of detection quality

The face_recognition library does not allow to recognize, even detect faces on the camera at decent distances, say a few meters. But with the help of image manipulation methods, as well as playing with hyperparameters, in particular the threshold value, a good result was achieved in face recognition. The previous images (see Figure 3) show an enlarged, filtered, optimized frame with a person where a person's face is clearly detected. At the level of our development, all these points are registered in the backend and are not visible to the user of the developed software.

4. Results

As a result, this article applies the results of the work of the existing development, previously optimized by the methods described in part of the methodology. In connection with the results achieved in the development of optimized modules for detecting and recognizing faces, monitoring people in the zone, monitoring the intersection of the line, detecting abandoned objects and searching for a person, the possibility of adding triggers and viewing analytics were also added to the existing developments. The correct operation of powerful YOLO models makes it possible to add secondary features of the developed system, such as sending messages to messengers or to e-mail according to a given trigger (an event created by the software user). A smart video surveillance and monitoring system was implemented using computer vision algorithms, and accordingly, the methods proposed in the article were used to optimize them. Figure 6 shows screenshots of the operation of the module for monitoring people in the zone.

The human line crossing module is also a resource-intensive task, since in the context of frame processing using YOLO for the human detection subtask, an additional task of calculating coordinates in each frame is added, these tasks have been optimized using the proposed methodology. Screenshots of the implementation of this module are presented in Figure 7.

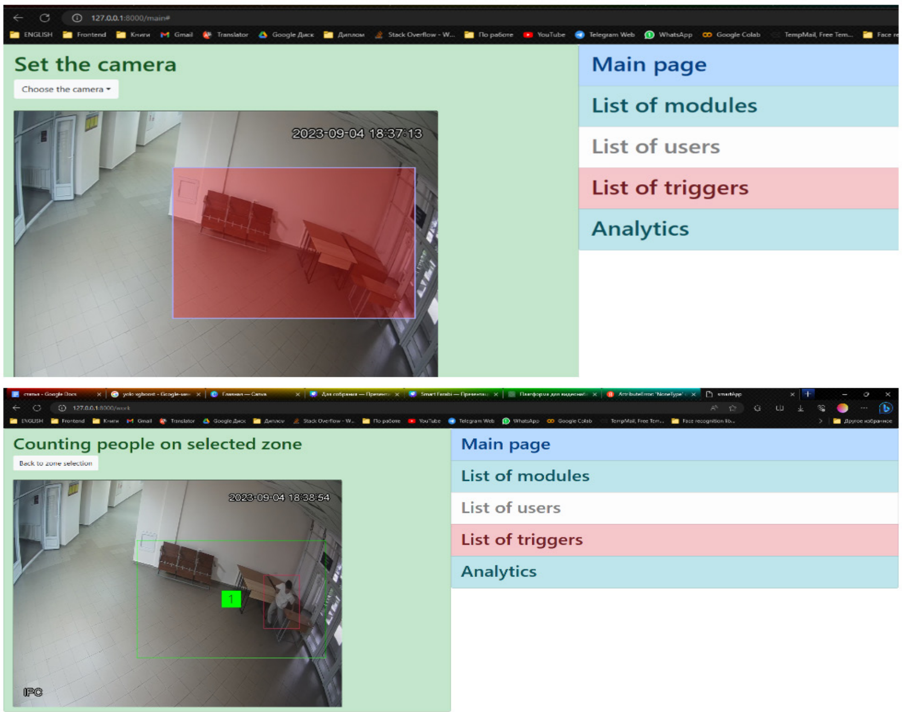

Figure 6 – Screenshot of the results of the development of the zone detection module

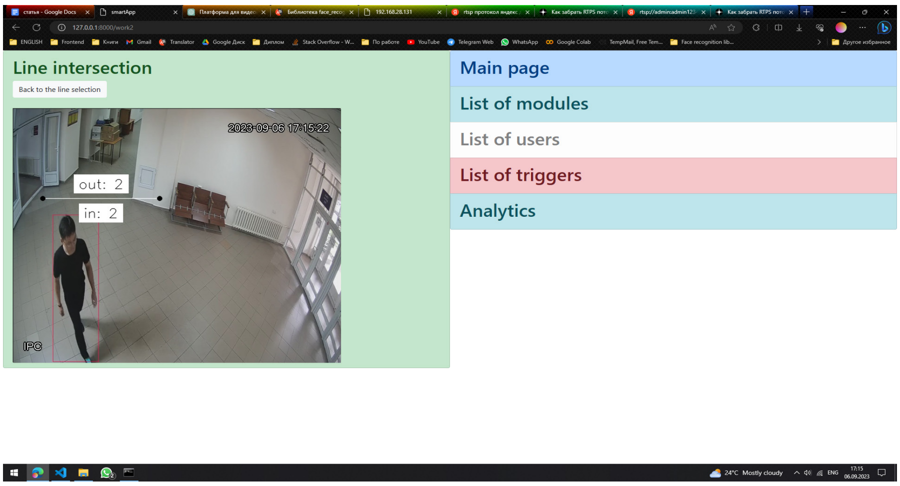

Figure 7 – Screenshot of the results of the development of the line intersection module

People searching module (see Figure 8) is the most high-performance in the existing development, because it includes the tasks of human detection as well as facial recognition. During its implementation, before detecting a face, the process of detecting a person and subsequent image manipulations, such as zooming, contrast enhancement, and noise removal, are pre-launched. Additionally, in addition to the proposed multithreading technique, the yolo ono model was used to further improve performance.

Figure 8 – Screenshot of the results of the development of the human search module

5 Conclusion

In this research paper, the results of the development and optimization of modules for an integrated video surveillance and control system based on YOLO algorithm were presented. Based on a number of methods and techniques, including multithreading and image manipulation, there has been a significant improvement in the performance and accuracy of video stream analysis.

The system presented in this paper has the ability to detect and classify objects in real time, which significantly increases its potential application in the field of security, monitoring and automation of processes. Key modules, including face recognition, object detection, and counting and motion tracking, have demonstrated outstanding performance and accuracy under various operating conditions.

The results of the comparisons indicate a significant increase in performance, as well as maintaining high accuracy of object detection. It is worth noting the increase in frame rate (FPS), which is a key indicator for real-time system.

In conclusion, this paper highlights the importance of a combined approach, including both technical optimizations and image manipulation methods, to create highly efficient integrated video surveillance and control systems. The results obtained have the potential for wide application in various fields, including ensuring the safety of society and industrial applications.

References

1. Kadikonda Mounika, Vancha Vaishnavi Reddy, Asma Begum (2021). “Intelligent video surveillance using deep learning”. International Journal for Innovative Engineering and Management Research. Volume 11 Issue no.06, (2021): 566-579.

2. Wang, Xiaogang. 2013. “Intelligent Multi-Camera Video Surveillance: A Review.” Pattern Recognition Letters. Elsevier BV. https://doi.org/10.1016/j.patrec.2012.07.005.

3. Liu, Li, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jie Chen, Xinwang Liu, and Matti Pietikäinen. 2019. “Deep Learning for Generic Object Detection: A Survey.” International Journal of Computer Vision. Springer Science and Business Media LLC. https://doi.org/10.1007/s11263-019-01247-4.

4. Rekha B. S., Athiya Marium, Dr. G. N. Srinivasan, Supreetha A. Shetty. 2020. “Literature Survey on Object Detection using YOLO”. International Research Journal of Engineering and Technology (IRJET). Volume 07, Issue no. 06 (2020): 3082-3088.

5. Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. “You Only Look Once: Unified, Real-Time Object Detection.” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE. https://doi.org/10.1109/cvpr.2016.91.

6. Girshick, Ross. 2015. “Fast R-CNN.” arXiv. https://doi.org/10.48550/ARXIV.1504.08083.

7. Jiang, Peiyuan, Daji Ergu, Fangyao Liu, Ying Cai, and Bo Ma. 2022. “A Review of Yolo Algorithm Developments.” Procedia Computer Science. Elsevier BV. https://doi.org/10.1016/j.procs.2022.01.135

8. Ben-Nun, Tal, and Torsten Hoefler. 2019. “Demystifying Parallel and Distributed Deep Learning.” ACM Computing Surveys. Association for Computing Machinery (ACM). https://doi.org/10.1145/3320060.

9. Lou, Haitong, Xuehu Duan, Junmei Guo, Haiying Liu, Jason Gu, Lingyun Bi, and Haonan Chen. 2023. “DC-YOLOv8: Small Size Object Detection Algorithm Based on Camera Sensor.” MDPI AG. https://doi.org/10.20944/preprints202304.0124.v1

10. Wei, Xiaoyan, Yirong Wu, Fangmin Dong, Jun Zhang, and Shuifa Sun. 2019. “Developing an Image Manipulation Detection Algorithm Based on Edge Detection and Faster R-CNN.” Symmetry. MDPI AG. https://doi.org/10.3390/sym11101223.

IRSTI 28.23.29 https://doi.org/10.26577/jpcsit2023v1i4a8

Ye.A. Amanbay , N. Azatbekuly*, A.A. Mukhanbet

Al-Farabi Kazakh National University, Almaty, Kazakhstan
*e-mail: nurtugang17@gmail.com

IMPLEMENTATION AND COMPARISON OF REINFORCEMENT LEARNING ALGORITHMS FOR SOLVING THE PROBLEM OF FINDING A PATH IN A 2D MATRIX

Abstract. The topic of this paper is the study of two reinforcement learning algorithms, SARSA and Q-Learning. Reinforcement learning is generating significant interest due to its potential applications in various domains such as robotics, gaming, optimization, etc. In addition, reinforcement learning is an interesting object of research from the perspective of theory and practice, as it is related to concepts such as exploration and use, learning and planning, consistency, and stabilization, etc. SARSA and Q-Learning are two of the most well-known and widely used reinforcement learning algorithms, which are based on the evaluation of the value function of states and actions. The aim of this paper is to study the learning characteristics of these algorithms in different scenarios of agent's interaction with the environment. To this end, experiments were conducted in which the agent had to find an optimal path in a 2D matrix containing walls to reach the final position safely. The results showed that SARSA was on average 28.3% faster than Q-Learning.

Key words: SARSA, Q-Leaning, Reinforcement Learning.

1 Introduction

Reinforcement learning (RL) [1] is a task faced by an agent that must learn behavior through trial-and-error interactions with the environment to achieve a goal. There are three main classes of methods for solving the OP problem: dynamic programming, Monte Carlo methods, and temporal difference (TD) learning. Reinforcement learning, which is a field of machine learning, is becoming one of the main tools of computational intelligence as a technique in which computers make their own choices in each environment without historical or labeled data [2]. By interacting with the environment, the agent predicts the optimal decision and continuously develops and learns the optimal policy based on the value function [3]. In [4], TD methods update estimates partly based on other estimates. They are trained to build hypotheses on hypotheses. Since TD-learning is a combination of Monte Carlo and dynamic programming ideas, much attention has recently been paid to the study of TD-learning [5,6]. It is precisely TD-learning that formed the basis of this research to implement SARSA and Q-Learning algorithms. It is stated in [4] that one of the early breakthroughs in reinforcement learning was the development of a TD algorithm for split strategy control, known as Q-Learning. And a classic on-policy TD algorithm called SARSA (State Action Reward State Action) [7] is considered critical to the success of OD [8]. In [9], it is found that SARSA and Q-learning have different performance depending on the learning environment. From this, it can be hypothesized that in the task of path finding in 2D matrix, SARSA may be a better algorithm option.

The paper primarily seeks for an agent to maximize total rewards within a virtual environment through sequential actions. Specifically focusing on SARSA and Q-Learning algorithms in a 2D matrix, the objective is for the agent to navigate to the target while navigating obstacles. This scenario is an illustrative example of the application of reinforcement learning in solving the problem of optimal behavior under resource constraints and possible complications. Illustrating the practical use of reinforcement learning in addressing decision-making challenges, the paper analyzes algorithm performance, aiming to highlight their distinct strengths in solving this problem.

2 Materials and methods of research

2.1 Time-Difference Forecasting

In time-difference (TD) based forecasting, as in the Monte Carlo method, the value of states is sought. However, in the Monte Carlo method, the value function is estimated by simple mean reversion, whereas in TD-learning, the value of the current state is updated by the current state. In TD-learning, the so-called time-difference based update rule is used to update the state value, it is represented in formula (1):

$$V(s) = V(s) + \alpha(r + \gamma V(s') - V(s)), \quad (1)$$

where, $V(s)$ – value of the previous state, α – learning rate, r – reward, γ – correction factor, $V(s')$ – value of the current state.

2.2 SARSA algorithm

Rummery and Niranjan [10] proposed a modified Q-learning algorithm called SARSA. Unlike the traditional Q-learning algorithm, SARSA is a policy enabled TD algorithm whose updates are policy dependent.

SARSA is one of the most popular reinforcement learning algorithms used to train agents to interact with the environment. It gets its name from the sequence of actions and states in a task: State-Action-Reward-State-Action.

SARSA updates Q-value estimates for state-action pairs based on the rewards received and the agent's experience. The updates of Q-values are calculated using formula (2):

$$Q(s,a) = Q(s,a) + \alpha(r + \gamma Q(s',a') - \\ -Q(s,a)), \quad (2)$$

where, α is the learning rate, a'- the action chosen by the epsilon greedy strategy (ε> 0), γ – correction factor.

The process of updating Q-values in SARSA is as follows:

1. The agent starts in state $'s'$ and chooses action $'a'$ according to its strategy;
2. The agent performs action $'a'$ and moves to a new state $'s'$ while receiving reward $'r'$;
3. The agent chooses a new action $'a'$, again according to its strategy.

A block diagram of the SARSA algorithm is shown in (Fig.1):

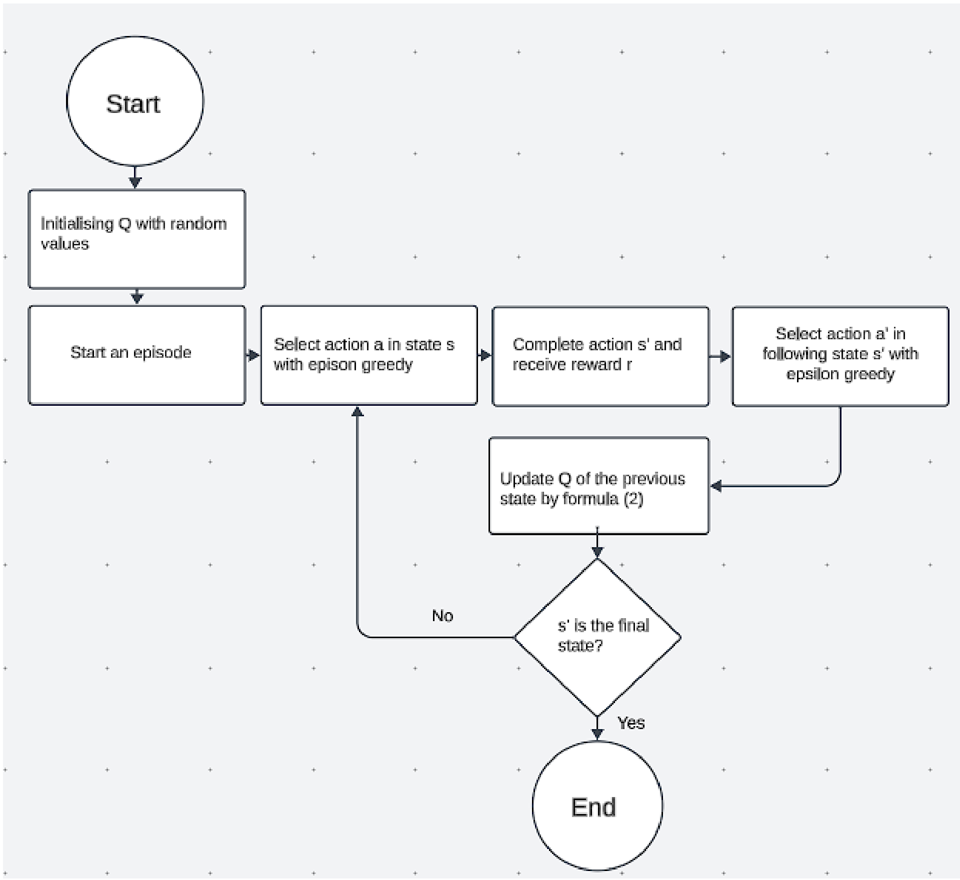

Figure 1 – Block diagram of the SARSA algorithm

1.3 Q-learning algorithm

Q-learning is another important reinforcement learning algorithm that is also used to find the optimal strategy in a path planning task. Unlike SARSA, Q-learning estimates the maximum expected Q-value for each state-action pair.

The process of updating Q-values in Q-learning is as follows:

1. The agent starts in state 's' and selects action 'a' according to its strategy.

2. The agent performs action 'a' and moves to a new state 's'' while receiving reward 'r'.

The value of Q is updated according to formula (3):

$$Q(s,a) = Q(s,a) + \alpha(r + \gamma maxQ(s',a') - Q(s,a)), \quad (3)$$

where, α is the learning rate, a'- the action chosen by the epsilon greedy strategy (ε> 0), γ – correction factor.

Q-Learning evaluates the optimal strategy by maximizing Q-values for each state and action. This allows the agent to converge to the optimal strategy faster but may require more computational power and may be more sensitive to noise in the data.

A block diagram of the Q-Learning algorithm is shown in (Figure 2).

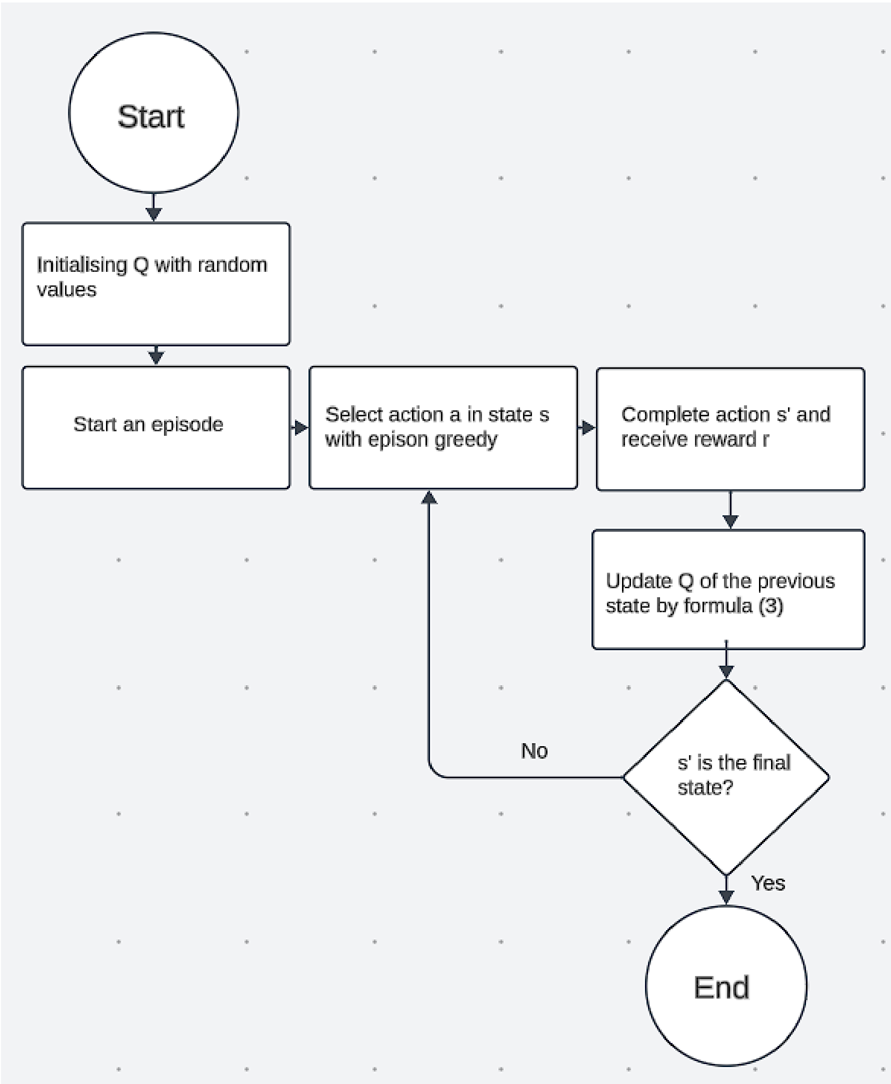

Figure 2 – Block diagram of the Q-Learning algorithm

3 Results of research

We run SARSA and Q-Learning in a 2D matrix with dimensions 10 by 10 and 20 by 20 (Figure 3). A small negative reward, -1, is given for each move, and a large reward, 100, is given for reaching the end point.

The number of episodes is equal to 1000, learning rate $\alpha = 0.1$, $\gamma = 0.9$, $\varepsilon = 0.01$

In a comparison of plots of the number of steps per episode on a 10 by 10 matrix, SARSA is faster and finds the optimal path in the least number of episodes than Q – Learning (Fig.4).

When increasing the size of the 2D matrix to 20 by 20, both algorithms needed more steps to reach the target point, however, the SARSA algorithm was able to reach the target point faster by minimizing the number of steps with each episode (Figure 5).

As can be seen in (Fig.6), the SARSA algorithm minimizes the number of collisions to zero with each episode. This is due to the fact that SARSA is an on-policy algorithm, i.e., the agent updates Q-values given its own actions, according to the current strategy. This makes the agent more cautious by choosing actions that are already known to be safe. While Q-Learning is an off-policy algorithm, based on this the agent updates Q-values based on an optimal strategy, which may be more exploratory and lead to the selection of less known but potentially dangerous actions.

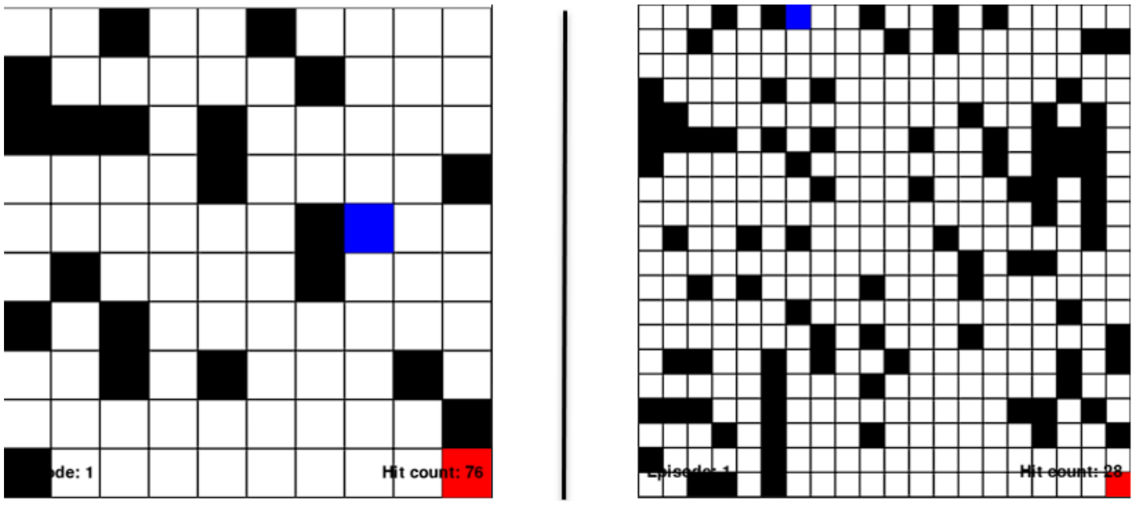

Figure 3 – 10 by 10 and 20 by 20 matrices, where (Blue square is agent, red square is target position)

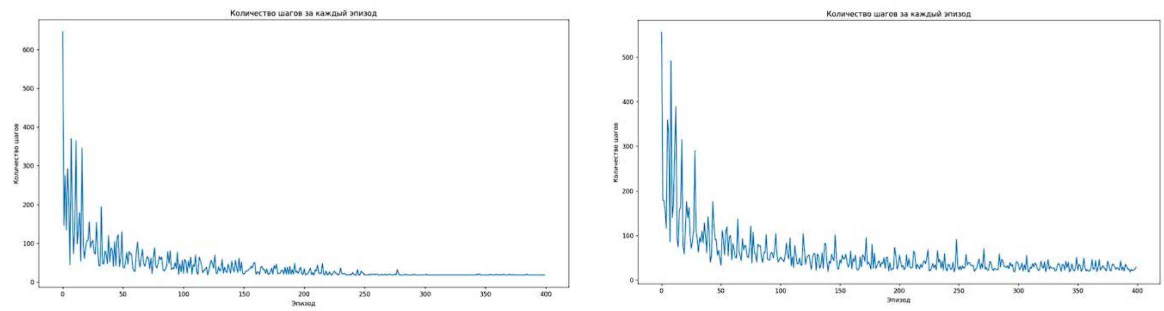

Figure 4 – Graph of the number of steps per episode of the SARSA (left) and Q-Learning (right) algorithms

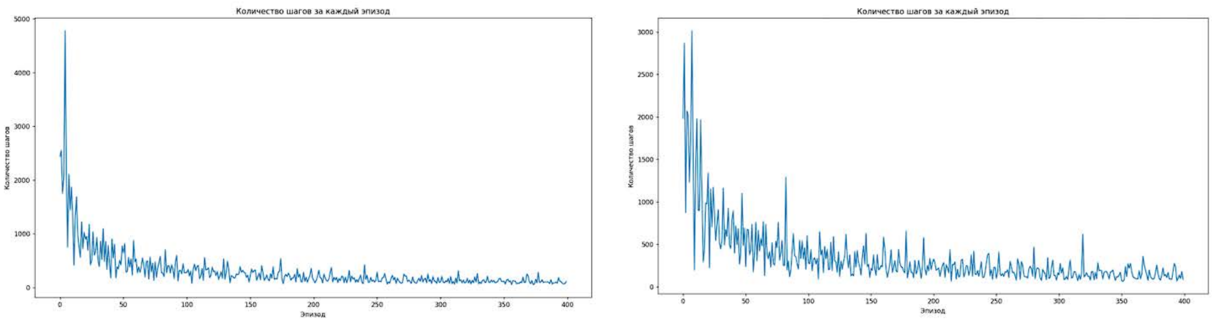

Figure 5 – Graph of the number of steps per episode of the SARSA (left) and Q-Learning (right) algorithms

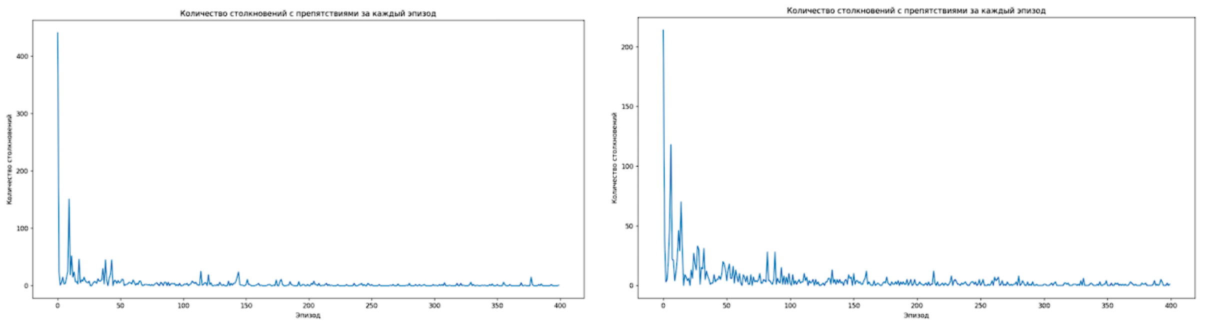

Figure 6 – Graph of the number of obstacle collisions per episode of the SARSA (left) and Q-Learning (right) algorithms, in a 20 by 20 environment

From the results of these experiments, it is evident that the SARSA algorithm demonstrated better performance in several key aspects compared to the Q-learning algorithm. Specifically, SARSA showed significant advantages in the following parameters:

1. Number of collisions: SARSA also demonstrated a lower number of collisions with obstacles. This is due to the fact that SARSA considers recent actions and avoids repeated errors, which makes it more robust in collision avoidance.

2. Number of steps per episode: SARSA often required fewer steps to reach a goal compared to Q-learning. This indicates its ability to more efficiently select actions that lead to successful task completion.

In case the environment has a high noise level or is unstable, SARSA may be preferable, as it updates its strategy based on the actual actions taken by the agent. In tasks with large state and action space, Q-learning may be more effective as it can learn based on the value of the best action. However, when emphasizing the criteria of optimal path length, number of collisions, and number of steps per episode, SARSA appears to be a more appropriate algorithm.

4 Conclusion

In this paper, two popular reinforcement learning algorithms, SARSA and Q-Learning, were implemented in the context of the 2D pathfinding task. Experimental results showed that SARSA was more efficient in reaching the optimal path, converging to it 28.3% faster compared to Q-Learning.

This improvement in the convergence speed of SARSA represents a significant finding that emphasizes the advantages of this algorithm in solving a particular problem. The success of SARSA can be attributed to its ability to learn more efficiently in environments where obstacles are present.

The SARSA algorithm showed significant advantages over Q-Learning in several important characteristics such as the length of the optimal path, the number of collisions, and the number of steps per episode. SARSA proved to be more efficient in finding optimal strategies for moving the agent, which is of practical importance in robotics and autonomous systems. The results suggest that SARSA is the preferred choice for the 2D matrix pathfinding problem, collision avoidance and step count optimization play an important role.

This analysis allowed us to gain a deep understanding of how different algorithms affect the robot's navigation performance.

This study emphasizes the importance of selecting the most appropriate algorithm for a particular task. In the experiments, SARSA showed advantages in environments where minimizing the length of the optimal path and avoiding collisions are critical factors. Despite the advantages of SARSA, Q-Learning remains an important tool, especially in partial observability environments where it can adapt to unexpected situations and find optimal solutions.

References

1. Sutton, Richard S., and Andrew G. Barto. Reinforcement Learning: An Introduction. Cambridge, MA: MIT Press, 1998.

2. Jordan, M. I., and T. M. Mitchell. 2015. “Machine Learning: Trends, Perspectives, and Prospects.” Science. American Association for the Advancement of Science (AAAS). https://doi.org/10.1126/science.aaa8415.

3. Tesauro, Gerald. 1995. “Temporal Difference Learning and TD-Gammon.” Communications of the ACM. Association for Computing Machinery (ACM). https://doi.org/10.1145/203330.203343.

4. Sutton, Richard S., and Barto, Andrew G. Reinforcement Learning: An Introduction. MIT Press, 2018

5. Sutton, Richard S. "Temporal Credit Assignment in Reinforcement Learning." Ph.D. thesis, University of Massachusetts, 1984.

6. Sutton, Richard S. 1988. “Learning to Predict by the Methods of Temporal Differences.” Machine Learning. Springer Science and Business Media LLC. https://doi.org/10.1007/bf00115009.

7. Sutton, Richard. “Generalization in reinforcement learning: Successful examples using sparse coarse coding Advances in Neural Information Processing Systems, edited by David S. Touretzky, Michael C. Mozer, and Michael E. Hasselmo, pp. 1038-1044. The MIT Press, 1996.

8.Wang, Yin-Hao, Tzuu-Hseng S. Li, and Chih-Jui Lin. 2013. “Backward Q-Learning: The Combination of Sarsa Algorithm and Q-Learning.” Engineering Applications of Artificial Intelligence. Elsevier BV. https://doi.org/10.1016/j.engappai.2013.06.016.

9. Sutton, Richard. “Open theoretical questions in reinforcement learning”. Proceedings of EuroCOLT’99, 1999

10. Rummery, G.A., and Niranjan, M. On-line q-learning using connectionist systems. Technical Report CUED/F-INFENG/TR 166. Cambridge: Cambridge University Engineering Department, 1994.

Contents