ИНСТИТУТ СИСТЕМНОГО
АНАЛИЗА И УПРАВЛЕНИЯ

# COLLECTION OF SCIENTIFIC AND PROJECT ACTIVITY REPORTS OF THE GRADUATES OF THE INTERNATIONAL SCHOOL OF INFORMATION TECHNOLOGIES “DATA SCIENCE”

DATA SCIENCE
IT SCHOOL

Ministry of Education of the Moscow Region

Dubna State University

Joint Institute for Nuclear Research

# COLLECTION OF SCIENTIFIC AND PROJECT ACTIVITY REPORTS OF THE GRADUATES OF THE INTERNATIONAL SCHOOL OF INFORMATION TECHNOLOGIES “DATA SCIENCE”

Collection of works

VOLUME 1

Edited by V. Korenkov, Eu. Cheremisina, O. Streltsova, D. Priakhina

DUBNA 2020

UDC 004.42
LBC 32.97я43
C 232

C 232 **Collection of scientific and project activity reports of the graduates of the International School of Information Technologies "Data Science"** : collection of works. Volume 1 / edited by V. Korenkov et al. – Dubna : Dubna State University, 2020. – 50 p.
Translators: Julia Meshcherskaya, Ekaterina Voytishina.

ISBN 978-5-89847-609-0

The publication presents a collection of brief summary reports of the State Budgetary Educational Institution of Higher Professional Education of the Moscow Region "Dubna University" students who have completed their studies at the International School of Information Technologies "Data Science".

During the studies, the students were involved in real promising projects of the Joint Institute for Nuclear Research (JINR, Dubna, Russia), the work in which was performed by the students within the discipline "Scientific projects" under the supervision of the JINR and Dubna University staff.

UDC 004.42
LBC 32.97я43

ISBN 978-5-89847-609-0

# CONTENT

# PREFACE

The International School of Information Technologies “Data Science” (hereinafter referred to as the IT School) is a joint educational project of the Laboratory of Information Technologies (LIT) of the Joint Institute for Nuclear Research (JINR) and the Institute of System Analysis and Management (ISAM) of Dubna State University. Its goal is to train highly qualified IT specialists for the development of megascience projects’ computing, Big Data analytics, the digital economy and other perspective directions.

The educational program of the IT School is formed taking into account the personnel needs of JINR and other organizations of the high-technology sector of the economy and is implemented with their participation. The program comprises such disciplines as:

- Additional chapters of mathematics;
- Programming languages for data analysis;
- Introduction to Unix operating systems;
- Tools for the collaborative development of software;
- Introduction to cloud technologies;
- Big Data analytics;
- Distributed systems;
- Multiagent systems;
- High-performance computing;
- Professional English.

Practical classes are held in computer rooms of Dubna University, involving the resources of the HybriLIT heterogeneous computing platform, which is part of the Multifunctional Information and Computing Complex of LIT JINR.

Employees of Dubna University and JINR participate in the organization of the educational process and the creation of a software and information environment.

The first competitive selection of students of Dubna University who wanted to enter the IT School took place in February 2019, and classes started on 1 March 2019. It should be noted that training is free. The educational program of the IT School is mastered by students in parallel with the main educational program.

The IT School closely collaborates with leading Russian universities that train qualified IT specialists. Therefore, during the studies, students gained knowledge and competences in the field of modern computing and Big Data analytics not only from teachers of Dubna University and the JINR staff. Lectures were also given by teachers from such Russian universities as the National Research Nuclear University MEPhI, Plekhanov Russian University of Economics, etc.

Moreover, students attended lectures and workshops from leading experts of Russian companies and foreign organizations (in English):

- lecture on the topic “Deep and Machine Learning methods for document clustering and classification” on deep and machine learning from a data analysis specialist of the SAP SE company (Germany), on the basis of LIT JINR (17 April 2019);
- workshop on the topic “Intel architectures and technologies for high-performance computing and the tasks of machine/deep learning (ML/DL)” from specialists of the Intel and RSC companies, on the basis of LIT JINR (15 November 2019);

- workshop on high-performance computing on the basis of the National Research University "Higher School of Economics" (21 January 2020).

Students of the IT School who went through a challenging competitive selection took part in student educational and scientific events, including foreign ones:

- International IT School "Machine Learning, Parallel and Hybrid Computations & Big Data Analytics" within the International Conference "Mathematical Computational Physics – MMCP'2019" (1-5 July 2019, Slovakia);
- Summer Computer School "Data Science Dubna-2019" (6-13 July 2019, Dubna University, Dubna, Russia);
- International School "Big Data mining and distributed systems" within the International Conference "Symposium on Nuclear Electronics and Computing – NEC'2019" (29 September – 3 October 2019, Montenegro);
- School of Young Scientists "High-performance platforms for the digital economy and megascience projects" (3-4 December 2019, PRUE, Moscow, Russia);
- XXVII Scientific-practical Conference of students, postgraduate students and young specialists (14-27 April 2020, Dubna University, Dubna, Russia).

The first graduation of students of the IT School took place in June 2020. 21 graduates successfully completed their studies and were awarded certificates of additional education within the program "Big Data Analytics".

One of the major principles of the IT School rests on training through research. Therefore, during the studies, students were involved in real promising projects of JINR, the work in which was performed within the discipline "Scientific projects". The given publication presents a collection of brief reports on their activities.

The Directorate of the IT School expresses its gratitude to the teachers of Dubna University and the JINR staff for their fruitful work with students. The following specialists worked with students of the IT School:

- Balashov N., software engineer of LIT JINR;
- Belov S., leading programmer of LIT JINR;
- Gertsenberger K., candidate of technical sciences, scientific and experimental department of heavy-ion collision physics at the NICA complex, head of the group of mathematical support and software of VBLHEP JINR;
- Kadochnikov I., software engineer of LIT JINR;
- Kalinovsky Yu., doctor of physics and mathematics, leading researcher of LIT JINR;
- Koshlan D., software engineer of LIT JINR;
- Kullenberg Ch., researcher of DLNP JINR;
- Meshcherskaya J., associate professor of the department of SAM ISAM of Dubna University;
- Oleynik D., leading programmer of LIT JINR;
- Ososkov G., doctor of physics and mathematics, principal researcher of LIT JINR;
- Papoyan V., software engineer of LIT JINR;
- Pelevanyuk I., software engineer of LIT JINR;
- Petrosyan A., leading programmer of LIT JINR;

- Pyatkov Yu., doctor of physics and mathematics, leading researcher of FLNR JINR;
- Stadnik A., candidate of physics and mathematics, software engineer of LIT JINR;
- Sychev P., associate professor of the department of DICS ISAM of Dubna University;
- Zrelov P., candidate of physics and mathematics, head of the scientific and technical department of software and information support of LIT JINR.

*Scientific leaders of the IT School:*
***V. Korenkov****, doctor of technical sciences, director of LIT JINR, head of the department of DICS ISAM;*
***Eu. Cheremisina****, doctor of technical sciences, professor, academician of RANS, director of ISAM.*

*Director of the IT School:*
***O. Streltsova****, candidate of physics and mathematics, senior researcher of LIT JINR, associate professor of the department of DICS ISAM.*

*Scientific secretary of the IT School:*
***D. Priakhina****, software engineer of LIT JINR, senior teacher of the department of DICS ISAM.*

*itschool.jinr.ru*

# GRADUATES OF 2020

1. Artemiev Alexey
2. Gabdrakhimov Dauren
3. Gavrilov Dmitry
4. Ilina Anna
5. Kiseeva Victoria
6. Kostopravov Anton
7. Kuzmenkov Igor
8. Makhlov Egor
9. Matveev Ivan
10. Polonskiy Denis
11. Postolov Ilia
12. Potapov Denis
13. Rezvaya Ekaterina
14. Rogozhina Elizaveta
15. Rudenko Mikhail
16. Ryabov Andrey
17. Smetanin Artem
18. Tjupin Denis
19. Yachmeneyov Andrew
20. Zhatkina Kristina
21. Zizganov Timofei

# DATA MINING

# DEVELOPMENT OF WEB-INTERFACE OF THE NEWS AGGREGATOR IN THE THEMATIC DIRECTION "CLOUD TECHNOLOGIES"

**Gabdrakhimov Dauren**[1], **Koshlan Diana**[2]

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 21;*
*The main educational program:*
*Software engineering, group 4253;*
*e-mail: daur9613@gmail.com.*

[2] *Software engineer;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Postgraduate student;*
*Department of System Analysis and Management;*
*Dubna State University.*

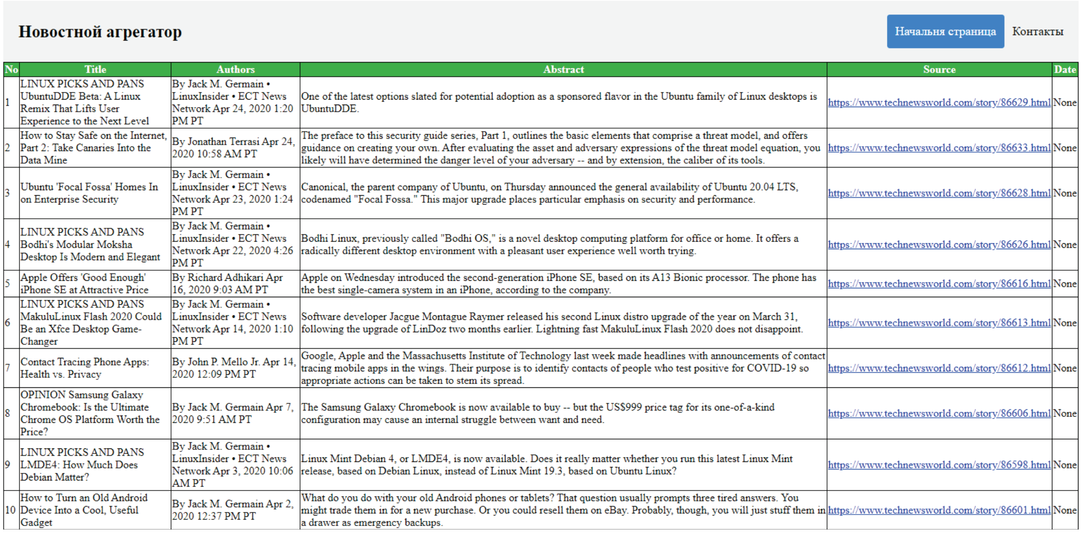
Keywords: agent technologies, information search and processing, web-interface development.

In the modern world, there is a significant increase in the number of sources of scientific and technical information on the Internet, which complicates its high-quality processing by scientists. In this regard, there is a need to create a news aggregator for the operational familiarization of the staff of the Laboratory of Information Technologies of the Joint Institute for Nuclear Research with new publications in the field of “Cloud Technologies”.

The aim of the project is to develop a web-based interface of the news aggregator in the thematic area “Cloud Technologies”.

The system works as follows: an agent collects data from more than 100 sources of authoritative publications in the field of information technology. Aggregation of news material is carried out in a centralized database [1]. To realize the goal of this project, I solved the problem of creating a website that displays news information collected by the agent.

As a result of my work, I have studied the technologies for creating sites using the Django framework [2], have developed a data storage system using Django ORM, have implemented a news aggregator interface design and have unloaded the information of interest from the database of the agent on the web page (see Fig. 1). The site was developed in the Python programming language using Django. Requests, Beautiful Soup, Xpath libraries are used to collect and process information [3].



| No | Title | Authors | Abstract | Source | Date |
|---|---|---|---|---|---|
| 1 | LINUX PICKS AND PANS UbuntuDDE Beta: A Linux Remix That Lifts User Experience to the Next Level | By Jack M. Germain • LinuxInsider • ECT News Network Apr 24, 2020 1:20 PM PT | One of the latest options slated for potential adoption as a sponsored flavor in the Ubuntu family of Linux desktops is UbuntuDDE. | https://www.technewsworld.com/story/86629.html | None |
| 2 | How to Stay Safe on the Internet, Part 2: Take Canaries Into the Data Mine | By Jonathan Terrasi Apr 24, 2020 10:58 AM PT | The preface to this security guide series, Part 1, outlines the basic elements that comprise a threat model, and offers guidance on creating your own. After evaluating the asset and adversary expressions of the threat model equation, you likely will have determined the danger level of your adversary -- and by extension, the caliber of its tools. | https://www.technewsworld.com/story/86633.html | None |
| 3 | Ubuntu 'Focal Fossa' Homes In on Enterprise Security | By Jack M. Germain • LinuxInsider • ECT News Network Apr 23, 2020 1:24 PM PT | Canonical, the parent company of Ubuntu, on Thursday announced the general availability of Ubuntu 20.04 LTS, codenamed "Focal Fossa." This major upgrade places particular emphasis on security and performance. | https://www.technewsworld.com/story/86628.html | None |
| 4 | LINUX PICKS AND PANS Bodhi's Modular Moksha Desktop Is Modern and Elegant | By Jack M. Germain • LinuxInsider • ECT News Network Apr 22, 2020 4:26 PM PT | Bodhi Linux, previously called "Bodhi OS," is a novel desktop computing platform for office or home. It offers a radically different desktop environment with a pleasant user experience well worth trying. | https://www.technewsworld.com/story/86626.html | None |
| 5 | Apple Offers 'Good Enough' iPhone SE at Attractive Price | By Richard Adhikari Apr 16, 2020 9:03 AM PT | Apple on Wednesday introduced the second-generation iPhone SE, based on its A13 Bionic processor. The phone has the best single-camera system in an iPhone, according to the company. | https://www.technewsworld.com/story/86616.html | None |
| 6 | LINUX PICKS AND PANS MakuluLinux Flash 2020 Could Be an Xfce Desktop Game-Changer | By Jack M. Germain • LinuxInsider • ECT News Network Apr 14, 2020 1:10 PM PT | Software developer Jacgue Montague Raymer released his second Linux distro upgrade of the year on March 31, following the upgrade of LinDoz two months earlier. Lightning fast MakuluLinux Flash 2020 does not disappoint. | https://www.technewsworld.com/story/86613.html | None |
| 7 | Contact Tracing Phone Apps: Health vs. Privacy | By John P. Mello Jr. Apr 14, 2020 12:09 PM PT | Google, Apple and the Massachusetts Institute of Technology last week made headlines with announcements of contact tracing mobile apps in the wings. Their purpose is to identify contacts of people who test positive for COVID-19 so appropriate actions can be taken to stem its spread. | https://www.technewsworld.com/story/86612.html | None |
| 8 | OPINION Samsung Galaxy Chromebook: Is the Ultimate Chrome OS Platform Worth the Price? | By Jack M. Germain Apr 7, 2020 9:51 AM PT | The Samsung Galaxy Chromebook is now available to buy -- but the US$999 price tag for its one-of-a-kind configuration may cause an internal struggle between want and need. | https://www.technewsworld.com/story/86606.html | None |
| 9 | LINUX PICKS AND PANS LMDE4: How Much Does Debian Matter? | By Jack M. Germain • LinuxInsider • ECT News Network Apr 3, 2020 10:06 AM PT | Linux Mint Debian 4, or LMDE4, is now available. Does it really matter whether you run this latest Linux Mint release, based on Debian Linux, instead of Linux Mint 19.3, based on Ubuntu Linux? | https://www.technewsworld.com/story/86598.html | None |
| 10 | How to Turn an Old Android Device Into a Cool, Useful Gadget | By Jack M. Germain Apr 2, 2020 12:37 PM PT | What do you do with your old Android phones or tablets? That question usually prompts three tired answers. You might trade them in for a new purchase. Or you could resell them on eBay. Probably, though, you will just stuff them in a drawer as emergency backups. | https://www.technewsworld.com/story/86601.html | None |

*Fig. 1. News Aggregator Web Interface*

Further project activities include implementing a news aggregator work schedule and publishing a news aggregator website on the Internet using an apache server for use by the cloud team of the Laboratory of Information Technology [4].

## References

1. Koshlan D., Tretyakov E., Korenkov V., Onyky B., Artamonov A. Multiagent information-analytical system in the thematic area “Cloud Technologies” // Mathematics. A computer. Education: Int. conf. (Dubna, January 27 - February 1, 2020). Izhevsk: Publishing House of ANO Izhevsk Institute for Computer Research, 2020. P. 198.
2. Web-framework Django (Python). – 2020. – [Electronic resource]. URL: https://developer.mozilla.org/ru/docs/Learn/Server-side/Django.
3. Web Scraping with python. – 2016. – [Electronic resource]. URL: https://habr.com/ru/post/280238/.
4. Apache HTTP Server. – 2010. – [Electronic resource]. URL: https://help.ubuntu.ru/wiki/apache2.

# APPLICATION OF INTELLECTUAL DATA ANALYSIS METHODS FOR DIGITAL EDUCATIONAL PLATFORM

**Zhatkina Kristina**[1], **Streltsova Oksana**[2]

*[1] Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 22;*
*The main educational program:*
*System analysis and management, group 6014;*
*e-mail: zhatkina-96@mail.ru.*

*[2] Candidate of Physics and Mathematics, senior researcher;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Associate Professor;*
*Department of Distributed Computer Systems;*
*Dubna State University.*

Keywords: agent, neural network, digital educational platform.

Currently, there are many electronic educational platforms that offer various courses on various topics. In connection with the latest events, there are more and more open free courses on each platform. Each course in turn has similar elements on most popular platforms: name, description, link to the structure of the course. Therefore, it was decided to analyze the possibilities of integrating various courses from educational platforms in one place for the convenience of their use.

One of the options for implementing a unified educational platform is the method of creating an agent [1] that collects the necessary information from sites, such as: coursera, stepik, etc., by the structure of the platform. The agent output file is a json file with the course structure inside. The agent is implemented in Python. Problems encountered during implementation: on the platform, courses are available only to registered users, resources do not allow parsing of the site again, some platforms have courses (blocks of courses) already divided into topics, not all courses are collected by the agent.

The next stage of the work was selecting and testing of the architecture of the neural network to build an individual trajectory of the (a) student. At the moment, the stage of classifying courses by topic is being implemented. The input to the neural network in the analysis of the text is a vector – a text in the form of numbers (the vectorization method is used). The tokenization method is the removal of functional words (semantically neutral words, such as conjunctions, prepositions, articles, etc.). Next, a morphological analysis is carried out (markup in parts of speech and stemmatization are performed). This can significantly reduce the dimension of space. Methods for extracting attributes from the text are N grams (sequences of words from 1 to N in length), a bag of words (bag of words, the set of all words). In neural networks, a dense vector representation of words (each token is associated with a vector, the vector dimension is lower than one hot encoding) is determined in the learning process [2]. At the first stage, the elements of the vectors are initialized with random numbers, and the vector values are changed using the back propagation method of error. As a result, the preparation of data for supply to the input of a neural network is the longest and the most labor-consuming process. To test the methods of artificial data analysis using a neural network, recur-rent neural networks (networks with cycles) were selected. To solve problems with RNN (training takes a long time, the problem of a disappearing gradient, limited «duration» of remembering previous information) more advanced architectures of recurrent networks LSTM and GRU and one-dimensional convolutional neural networks are selected [3,4].

The result of this work is an agent collecting courses from popular educational platforms, and a neural network that classifies a set of input data from a json file according to the topics of the courses. It is also planned to use a neural network to provide a user who has been registered on a single educational digital platform with blocks of courses according to templates. Currently, the agent structure for the coursera website, 3 neural network architectures (recurrent LSTM and GRU networks and a one-dimensional convolutional neural network) with testing on a marked up news dataset using the libraries tensorflow.keras, pandas, numpy, matplotlib, etc., is implemented. Further, it is possible to study the Django web framework (Python) to visualize the results of the agent and neural network. And also the refinement of the neural network for the analysis of the stack of competencies with the aim of the subsequent creation of an individual learning path.

References

1. IBM Cloud Application Performance Management. Configuring the Python agent, 2019.
2. Schollet F. Deep Learning in Python, 2018, 400 p.
3. Nikolenko S., Kadurin A., Arkhangelskaya E. Deep learning. Immersion in the world of neural networks, 2018, 480 c.
4. Chernyak E. Technology // Deep learning in text processing and analysis, 2019.

# ANALYSIS OF THE POSSIBILITIES OF USING IMAGE RECOGNITION TECHNOLOGY TO SOLVE THE PROBLEM OF AUTOMATED ATTENDANCE TRACKING

**Ilina Anna[1], Pelevanyuk Igor[2]**

*[1] Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 22;*
*The main educational program:*
*System analysis and management, group 6015;*
*e-mail: anna.ilina.1307@yandex.ru.*

*[2] Software engineer;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Assistant;*
*Department of Distributed Computer Systems;*
*Dubna State University.*

Keywords: image recognition, artificial intelligence, automation of attendance tracking.

The task of counting the number of people is relevant when conducting various events, which may be seminars, lectures, conferences, meetings, concerts, etc. Instead of monotonous manual counting participants, it is much more efficient to use some equipment with the necessary software that would allow detecting images of participants' faces and giving out the total number of all of them.

The aim of the project is to research the possibilities of using face recognition technology in images or video stream to solve the problem of automating attendance accounting. The work is primarily of a research nature, but the possibility of using the resulting system for practical purposes is not ruled out. The expected result is a rapidly implemented software and hardware complex that allows us to solve the problem and assess the feasibility of using such a solution for practical purposes. To solve the problem, the following software and hardware were used ready-made neural network with the necessary API face_recognition [1], the Python programming language v3.6, a set of extensions to the Qt graphics framework for the Python language PyQt4, single board microcomputer Raspberry Pi 3 Model B+, laptop computer ASUS X751L. As a result, a system for detecting and recognizing both images, which are already available in the database and unknown to the system, was developed. Herewith, an unknown person will be recorded in a system storage as an Unknown_Number and the image will be saved for correct renaming.

I chose to use the face_recognition neural network because of its simple and sufficient API to solve the problem, proposed a graphical interface of the system being developed, and developed the system itself, tested it on two selected sets of square images: Face Research Lab London Set [2] and Labeled Faces in the Wild [3], and analyzes of these results are also presented: the system's operating time was measured during face recognition in the video stream (the following were used: the laptop's built-in webcam, a Raspberry Pi camera, CBR CW 555M webcam), algorithm runtimes were measured on these sets of images that I resized from 1350 px to 50 px (in 10 photographs in sizes 1350, 1000, 800, 500, 400, 300, 200, 100, 90, 80, 70, 60, 50 px; on the total number of photos for each set in the size of 200, 100, 90 and 80 px). It was calculated: average encoding time per image, average time to search for a face in an image, average time for comparing the face found in the image with all available in the storage, total program runtime for image encoding, the total runtime of the program for image recognition. The number of encoded and recognized images for each of the sizes was measured. Corresponding diagrams have been constructed that display the dynamics of time changes versus image size.

The testing was implemented on the laptop and the microcomputer separately, as a result of which a noticeable difference in system operating time was found (on a microcomputer, the system runs about twice as slow). As the test results showed, the optimal image size is 100 * 100 px, which determines the smallest number of errors of the first and second kinds with the shortest system time (~ 16 min on the Raspberry Pi and ~ 6.17 min on a laptop using a set of 1020 photos). It follows from what has been said that in the work of the future system it is optimal to use images of 100 * 100 px in size.

This work has continued, as a result of which it is planned to split the system into server and client parts, as well as to get answers to an additional series of questions related to speeding up the system on other processors (including using Intel Movidius neural network acceleration technology), optimization of the algorithm, as well as with the possibilities of implementing this system in enterprises.

References

1. ageitgey/face_recognition: The world's simplest facial recognition api for Python and the command line. – [Electronic resource]. URL: https://github.com/ageitgey/face_recognition.
2. Face Research Lab London Set. – [Electronic resource]. URL: https://figshare.com/articles/Face_Research_Lab_London_Set/5047666.
3. LFW Face Database : Main. – [Electronic resource]. URL: http://vis-www.cs.umass.edu/lfw/.

# A STUDY OF THE CORRELATION FILTERS APPLICABILITY TO THE PROBLEM OF SEARCHING SIMILAR IMAGES IN THE DATABASE

**Kiseeva Viktoria**[1], **Streltsova Oksana**[2], **Stadnik Alexey**[3]

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 22;*
*The main educational program:*
*Applied mathematics and computer sciences, group 6181;*
*e-mail: vika.kiseeva@yandex.ru.*

[2] *Candidate of Physics and Mathematics, senior researcher;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Associate Professor;*
*Department of Distributed Computer Systems;*
*Dubna State University.*

[3] *Candidate of Physics and Mathematics, software engineer;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Associate Professor;*
*Department of High Mathematics;*
*Dubna State University.*

Keywords: correlation filter, computer vision, operations with images.

There are a number of tasks that require the implementation of algorithms for finding similar images in the database: systematization, cataloging, recommendation services, searching for similar people in the crowd, tracking. In general, searching for similar images implies seeking for images that are visually similar from a person's point of view.

The goal of the project was to build a simple and effective descriptor for evaluating the visual proximity of images using a correlation filter, as well as to perform a comparative analysis with the operation of the perceptual hash algorithm [1].

Using a correlation filter implies applying the Fourier transform and the convolution theorem [2][3]. The algorithm is implemented in the Visual Studio environment in the C++ programming language.

Creating an algorithm with appropriate characteristics means implementing an optimal method, or rather the simplest methods (such as perceptual hash algorithm), with a wider range of applicability than complicated methods (such as Siamese neural networks).

The algorithms efficiency is estimated by comparative analysis on open data and data collected specifically for the task, precisely: the Holiday dataset, Fashion MNIST, and storyboard of video fragments.

At this stage, the perceptual hash algorithm was implemented. The OpenCV computer vision library was used for this purpose [4]. The algorithm was carried out using the Phyton programming language. The stages of the perceptual hash construction algorithm are presented in figure 1:

- reducing the image without taking into account the aspect ratio;
- conversion of the image to grayscale.

Building a hash using information about the average brightness index for the entire image: each pixel is assigned 0 or 1, depending on whether it is greater or less than the average [1].

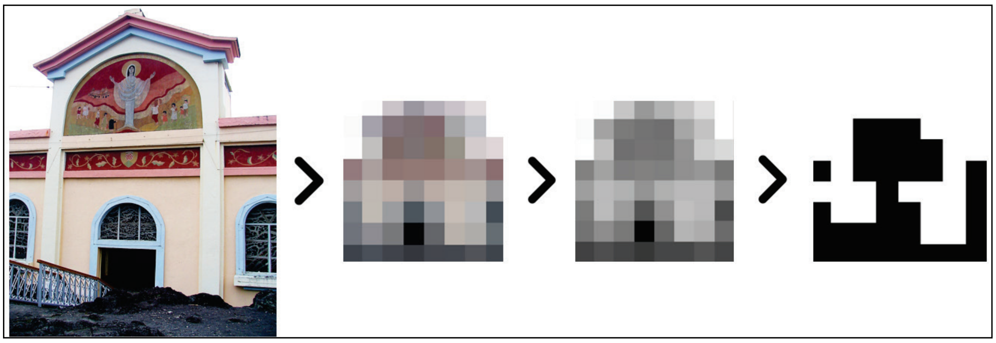


*Fig. 1. The result of the perceptual hash algorithm operation*

In the future, it is planned to complete the implementation of the algorithm using a correlation filter and conduct a comparative analysis of the algorithms operation.

## References

1. Looks Like It. – [Electronic resource]. URL: https://www.hackerfactor.com/blog/?/archives/432-Looks-Like-It.html.
2. Sadovnikov P. Optical trackers: ASEF and MOSSE. – [Electronic resource]. URL: https://m.habr.com/ru/post/421285/.
3. David S. Bolme, J. Ross Beveridge, Bruce A. Draper, Yui Man Lui. Visual Object Tracking using Adaptive Correlation Filters.
4. OpenCV documentation. – [Electronic resource]. URL: https://docs.opencv.org/.

# SOFTWARE COMPONENTS FOR MRI IMAGE ANALYSIS SERVICE

**Kostopravov Anton**[1], **Streltsova Oksana**[2]

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 21;*
*The main educational program:*
*Automation of technological processes and production, group 2231;*
*e-mail: akstud451@gmail.com.*

[2] *Candidate of Physics and Mathematics, senior researcher;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Associate Professor;*
*Department of Distributed Computer Systems;*
*Dubna State University.*

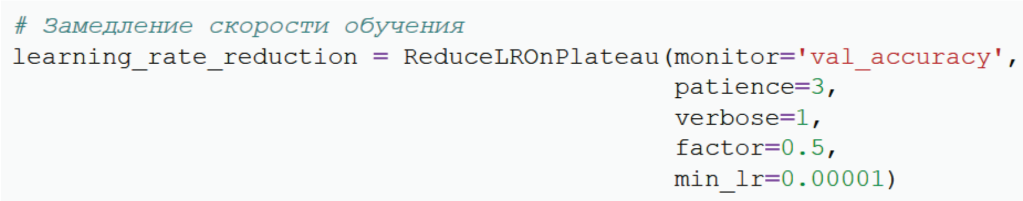
Keywords: classification, MRI, image processing.

The goal of this project is to create software components for the MRI image analysis service of the brain. Declared software components must analyze digital tomographic medical images to detect pathologies.

The software product was implemented in the Jupiter Notebook development service. The high-level programming language Python version 3.6.8 and the TensorFlow version 2.0 open source machine learning software library were selected for the implementation [1].

The first step was to search for a dataset, as well as searching and studying the already created solutions for its classification. To complete this task, a labeled Brain MRI Images for Brain Tumor Detection data set was selected on the Kaggle website, which consists of 253 images: 88 without tumors and 145 with tumors [2]. For training the neural network, the images were divided into a training sample consisting of 202 images and a validation sample having 51 images. As a result of studying ready-made solutions, it was found that the greatest classification accuracy was achieved using the pre-trained neural networks provided in Keras [3]. The highest accuracy of 93% on the validation dataset was achieved using the VGG19 neural network pre-trained on the ImageNet dataset.

The next step was to improve the learning efficiency of the neural network. For this, a callback function was created that slows down the learning speed if the proportion of correct answers on the validation sample does not change during the three training epochs (see Fig. 1). To increase the classification accuracy, the first five layers of a pre-trained neural network were frozen as well. This resulted in 98% classification accuracy on a validation sample.



```
# Замедление скорости обучения
learning_rate_reduction = ReduceLROnPlateau(monitor='val_accuracy',
                                            patience=3,
                                            verbose=1,
                                            factor=0.5,
                                            min_lr=0.00001)
```

*Fig. 1. Callback function*

In the future, the trained neural network can be used in the brain image analysis service or retrained for the classification of other tomographic images.

References

1. TensorFlow // Open software library for machine learning. – 2020. – [Electronic resource]. URL: https://www.tensorflow.org/.
2. Kaggle // A dataset with MRI images of the brain to detect tumors. – 2020. – [Electronic source]. URL: https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumor-detection.
3. Keras // Module applications deep learning library Keras. – 2020. – [Electronic resource]. URL: https://keras.io/applications/.

# BUILDING A HEAT MAP OF OBJECT MOVEMENT

**Polonskiy Denis**[1], **Streltsova Oksana**[2]

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 21;*
*The main educational program:*
*Software engineering, group 4252;*
*e-mail: denuha@mail.ru.*

[2] *Candidate of Physics and Mathematics, senior researcher;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Associate Professor;*
*Department of Distributed Computer Systems;*
*Dubna State University.*

Keywords: motion detection, heat map, OpenCV, video file.

In this project, the building of heat map of object movement has been realized. The heat map is a summation of time during which an object was moving in a specific spot within camera's FOV. To display the frequency of object's movement, a scale of three colors (green, yellow and red) which are applied to the image like a transparent layer is used [1].

This solution allows to determine the popularity of specific spots in a store (aisles, racks and counters); calculate optimized paths for people or vehicles in a specified area; analyze the visiting frequency of various objects [1].

The "in" of this solution is a video file, recorded by a stationary camera. It has been broken into frames (see Fig. 1). Each frame has been compressed (to lower the size of each image) and Gaussian's blur was applied in order to lower the amount of noise on the image.

To find and determine the movement on a frame, a difference between the current and the next frame is found. If movement was found, it is cut from the background (see Fig. 2) [2, 3, 4].

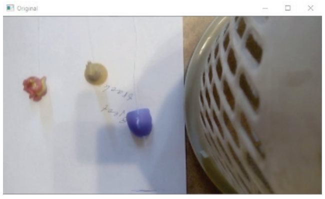
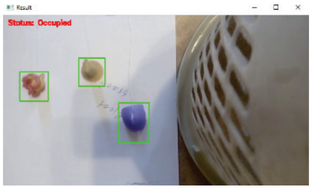


*Fig. 1. A frame from the video*



Status: Occupied

*Fig. 2. Motion detection*

Each instance of movement in each frame which has been found is saved and summed up as a result. The end result is a picture: a background from the first frame, with movement intensity – marked with colors for convenience – on it. Green color indicates low movement intensity, yellow – average intensity, and red – high intensity (see Fig. 3).

To work with the pictures, I have used the OpenCV library.

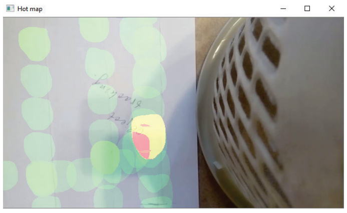


*Fig. 3. End result*

The development perspective is the implementation of a friendly graphical user interface with the ability to control image compression, Gaussian blur, the number of skipped frames to improve performance and to support work on various types of architectures.

References

1. Heat map of object movement (in Russian). – [Electronic resource]. URL: https://macroscop.com/assets/documentation/macroscop-2-6/win-client/analytics/hotmap.htm.
2. WebCam Motion Detector in Python – GeeksforGeeks. – [Electronic resource]. URL: https://www.geeksforgeeks.org/webcam-motion-detector-python/.
3. High-Performance Noise-tolerant Motion Detector in Python, OpenCV, and Numba. – [Electronic resource]. URL: https://bitworks.software/high-speed-movement-detector-opencv-numba-numpy-python.html.
4. Basic motion detection and tracking with Python and OpenCV – PyImageSearch. – [Electronic resource]. URL: https://www.pyimagesearch.com/2015/05/25/basic-motion-detection-and-tracking-with-python-and-opencv/.

# MIRROR SURFACES DETECTION IN THE IMAGE

**Postolov Ilia[1], Streltsova Oksana[2], Stadnik Alexey[3]**

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 22;*
*The main educational program:*
*Information systems and technologies, group 4281;*
*e-mail: innpot@uni-dubna.ru.*

[2] *Candidate of Physics and Mathematics, senior researcher;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Associate Professor;*
*Department of Distributed Computer Systems;*
*Dubna State University.*

[3] *Candidate of Physics and Mathematics, software engineer;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Associate Professor;*
*Department of High Mathematics;*
*Dubna State University.*

Keywords: deep learning, machine learning, machine vision, image, mirrors.

Mirrors and mirror surfaces are quite common in our daily lives. Existing computer vision systems do not consider mirror reflections in image processing and, therefore, can incorrectly interpret data due to the reflected content inside the mirror. However, it is extremely difficult to determine whether there is a mirror or other reflective surface in the image. The main problem is that mirror surfaces are usually reflect contents similar to their surroundings, which makes it difficult to find them [1].

This project explores the capabilities of the neural network approach for solving the problem of detecting mirror surfaces in images. Within the framework of this project, the following tasks will be solved: creation of a training dataset (Fig. 1), research of the applicability of the neural network approach, by testing various neural network architectures such as: U-net, CNN, Siamese neural networks, MirrorNet [2]. The current stage of the project implements U-net and CNN architectures in order to solve the task. In the future, the project plans to implement the Siamese neural network, MirrorNet architecture and training on a test dataset will be done, as well as analyzing and comparing of the results obtained from the usage of implemented neural network architectures.

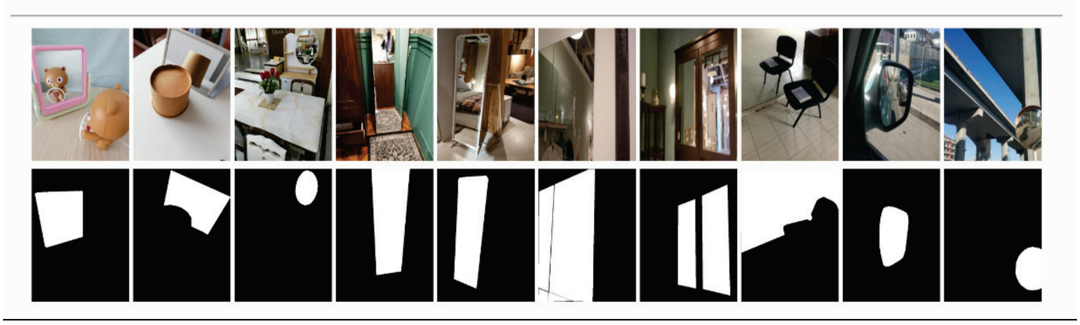


*Fig. 1. Training dataset from MirrorNet*

References

1. Yang Xin, Mei Haiyang, Xu Ke, Wei Xiaopeng, Yin Baocai, Lau Rynson W.H. Where Is My Mirror // The IEEE International Conference on Computer Vision (ICCV), October, 2019.
2. Schollet F. Deep Learning in Python, 2018, 400 p.

# USING DEEP DOMAIN ADAPTATION FOR IMAGE-BASED PLANT DISEASE DETECTION

**Rezvaya Ekaterina**[1], **Ososkov Gennady**[2], **Goncharov Pavel**[3]

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 22;*
*The main educational program:*
*Computer science and engineering, group 4013;*
*e-mail: rezvaya2016@gmail.com.*

[2] *Doctor of Physics and Mathematics, leading researcher;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Professor;*
*Department of System Analysis and Management;*
*Dubna State University.*

[3] *Postgraduate student;*
*Department of System Analysis and Management;*
*Dubna State University.*

Keywords: plant disease classification, neural networks, methods of domain adaptation.

Crop losses due to plant diseases are a serious problem for the farming sector and the economy. The ability to determine the type of disease at the early stage will help one to take necessary actions and prevent the infection spread. Therefore, a multi-functional Plant Disease Detection Platform (PDDP) was developed in the LIT JINR [1, 2].

Deep learning techniques are successfully used in PDDP to solve the problem of recognizing plant diseases from photographs of their leaves [2]. However, such methods require a large training dataset. Collecting and marking up an appropriate image database is a very complex and costly problem, while sufficient training data may not always be available and a relatively small one could be collected only. At the same time, there are number of methods used to solve classification problems in cases of a small training dataset [3]. For example, transfer learning [4], Deep Siamese Networks [4], zero-shot learning [4], domain adaptation [5].

There is a large database of Plant Village images in the open access. It consisting of 50,000 images of plant leaves with labels. However, all the photos are taken on a white background with strict positioning of the sheet in the center, which will prevent you from getting a good result if the image of the sheet is loaded in real conditions (the sheet is shifted relative to the center, the bright background, and other objects in the frame). Unlike the Plant Village dataset, PDD contains the real images from the Net. It is the presence of 2 similar data sets that influenced the choice of domain adaptation methods for training. The essence of these methods is that 2 datasets are used at once – the source domain (Plant Village dataset) and the target domain (PDD). The level of resemblance between the source and target domains hereby usually determines how successful the adaptation will be. Two methods were applied to the work. Both of them use convolution neural network. Normalization and augmentation data (flipped, random rotated and color jitter images ware added) was done before train.

The first method – Domain-Adversarial Training of Neural Networks (DANN): a neural network uses two losses functions, the classification loss and the domain confusion loss. By minimizing these functions, it is possible to ensure that both samples are indistinguishable for the classifier, which will achieve high accuracy [5]. There was pre-trained network MobileNetV2. The selected method did not allow to obtain the expected result on the PDD dataset [1]. The accuracy of the classification remained within 60%.

The second method uses two-step training for the ResNet34 pre-trained Neural Network. First, the network is trained on the source domain with a fixed training step, and then, after pre-replacing the classifier and freezing several layers, it is trained on the target dataset. Moreover, during training, there are several control points (every 100 epochs) where the value of the training step decreases by two orders of magnitude. This allows to improve the accuracy of classification, avoiding overfitting. The global goal of this method is to bring the distribution of weights closer to the desired one on the target base and get a higher classification quality [3]. This method allowed to achieve 90% classification accuracy. Classification using the Unsupervised Domain Adaptation with Deep Metric Learning (M-ADDA) method is also planned [5].

## References

1. Plant Disease Detection Platform // LIT JINR. – 2019. – [Electronic resource]. URL: http://pdd.jinr.ru/.
2. Uzhinskiy A. et al. Multifunctional Platform and Mobile Application for Plant Disease Detection / A. Uzhinskiy, G. Ososkov, P. Goncharov, A.Nechaevskiy //Proceedings of the 27th Symposium on Nuclear Electronics and Computing (NEC 2019), Budva, Montenegro. – 2019. – Vol. 2507. – P. 110-114.
3. Nikolenko S., Kadurin A., Arhangelskaya E. Deep learning // Spb.: Piter, 2018. – 480 p.
4. Goncharov P. et al. Deep Siamese Networks for Plant Disease Detection / Pavel Goncharov, Alexander Uzhinskiy, Gennady Ososkov, Andrey Nechaevskiy and Julia Zudikhina // EPJ Web of Conferences. – EDP Sciences, 2020. – Vol. 226. – P. 03010.
5. Overview of the main Deep Domain Adaptation methods (Part 1) // Mail.ru Groups blog. – 2018. – [Electronic resource]. URL: https://habr.com/ru/company/mailru/blog/426803/.

# MODELING OF FINE STRUCTURES IN THE MASS DISTRIBUTION OF NUCLEAR REACTION PRODUCTS AND THEIR RECOGNITION BY MACHINE LEARNING METHODS

**Rudenko Mikhail**[1], **Ososkov Gennady**[2], **Pyatkov Yuriy**[3]

*[1] Student;*
*Dubna State University;*
*The International School of Information Technologies*
*“Data Science”, group 21;*
*The main educational program:*
*Software engineering, group 4253;*
*e-mail: michadas@yandex.ru.*

*[2] Doctor of Physics and Mathematics, leading researcher;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Professor;*
*Department of System Analysis and Management;*
*Dubna State University.*

*[3] Doctor of Physics and Mathematics, Professor;*
*National Research Nuclear University MEPhI (Moscow Engineering Physics Institute).*
*Leading researcher;*
*Flerov Laboratory of Nuclear Reactions;*
*Joint Institute for Nuclear Research.*

Keywords: decays of heavy nuclei, modeling, deep learning, neuroclassifier.

The main goal of the project is to analyze clustering manifestations in rare multibody decays of the heavy nuclei [1]. The following tasks were set:

- to develop a computer model of the fine structure found by physicists from the JINR LNR based on experiments with the transuranic element californium $^{252}Cf$ (Fig. 1);
- to verify the hypothesis that the found structure objectively exists and is not a noise artifact.

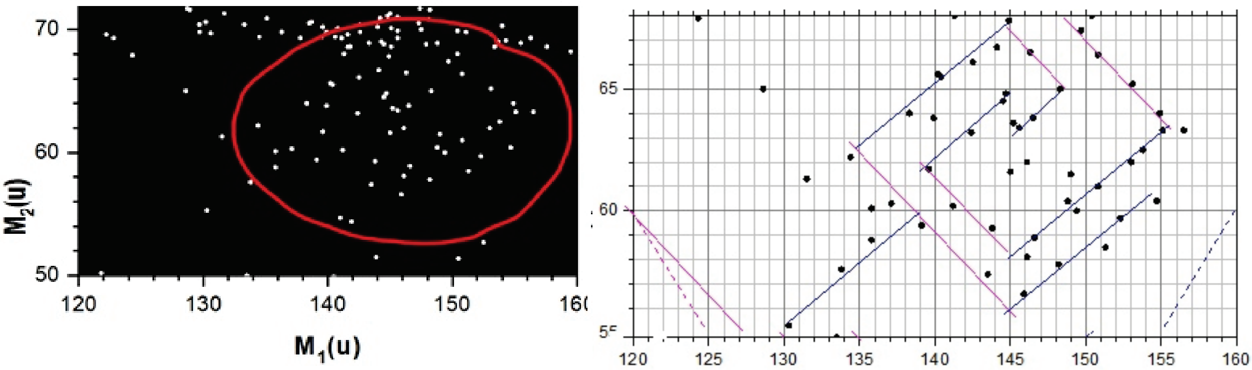


*Fig. 1. A fragment of the correlation-mass distribution of fission fragments $^{252}Cf$ from [1]. A specific rhombo-spiral structure is marked with an oval in the left picture. On the right, the same structure on a larger scale with the selection of fitted lines forming a rhombic meander*

Using the method of rotating histograms [2], 10 straight line segments forming a rhombic meander were recognized in the experimental image (Fig. 1 on the left), and their probabilistic parameters were determined based on statistical analysis. This allowed to create a numerical meander model and develop a generator program for both the thinnest structure and its alternative stochastic model in the form of the same number of random points random distributed over the meander field. A deep convolutional neural network was created as a binary classifier trained on a large sample of model and noise images obtained by the generator program. In the process of decision the Python programming language was used with connected libraries: matplotlib, karas, tensorflow, scikit-learn, numpy, pandas [3-8]. To confirm the hypothesis of non-random origin of the structure in the form of a rhombic meander, a numerical experiment was performed. As a result, a deep neuroclassifier was used to obtain a negligible probability of detecting a rhombic meander on an array of 105 statistically independent sets of random points (0.00017). The probability of a rhombo-helical structure in the original image (Fig. 1) was 99.913955%.

References

1. Yu. V. Pyatkov, et al., Eur. Phys. J. A 48, 94 (2012)
2. Nikitin V., Ososkov G., Automation of measurements and data processing of a physical experiment (monograph) (in Russian) // MSU, Moscow, 1986, 185 p.
3. Matplotlib: Visualization with Python. – [Electronic resource]. URL: https://matplotlib.org.
4. Keras. Simple. Flexible. Powerfull. – [Electronic resource]. URL: https://keras.io.
5. Tensorflow. An end-to-end open source machine learning platform. – [Electronic resource]. URL: https://www.tensorflow.org.
6. Scikit-learn. Machine Learning in Python. – [Electronic resource]. URL: https://scikit-learn.org.
7. NumPy. Fundamental package for scientific computing with Python. – [Electronic resource]. URL: https://numpy.org.
8. Pandas. Fast, powerful, flexible and easy to use open source data analysis and manipulation tool. – [Electronic resource]. URL: https://pandas.pydata.org.

# USING THE U-NET NETWORK IN IMAGE SEGMENTATION TASKS

**Ryabov Andrey[1], Streltsova Oksana[2]**

*[1] Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 22;*
*The main educational program:*
*Applied informatics, group 4071;*
*e-mail: rarjobs@mail.ru.*

*[2] Candidate of Physics and Mathematics, senior researcher;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Associate Professor;*
*Department of Distributed Computer Systems;*
*Dubna State University.*

Keywords: U-Net, architecture, convolutional networks, segmentation task.

The goal of the project is to review the architecture of U-Net network and apply it to object detection tasks. For this purpose, the task was defined, which is to detect ships and further determine the effectiveness of use U-Net.

U-Net was developed for segmentation of biomedical images. This convolutional neural network is characterized by high prediction accuracy and a small number of training data. On the descending (contracting) path, the convolution 3x3 operations with ReLu activation function and the join operations (MaxPool 2x2, stride 2) are performed sequentially, after one such iteration the property channels are doubled. On the ascending (expanding) path, the convolution 2x2 is applied that reduces the number of property channels, then the image is merged with the corresponding cropped image, after that the convolution 3x3 with the ReLu activation function takes place (Fig. 1).

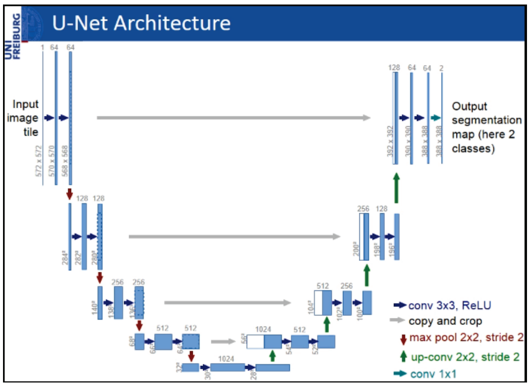


*Fig. 1. U-Net architecture [1-3]*

After studying the architecture, the task of detecting ships on aerial images was considered (Fig. 2). This network is trained on 1950 samples, and 50 samples are allocated for validation. Thus, on data of 2000 images, it turns out that the detection accuracy is 0,9888. From this we can conclude that the network efficiently segments the image in a small training sample and is suitable for accurate object detection.

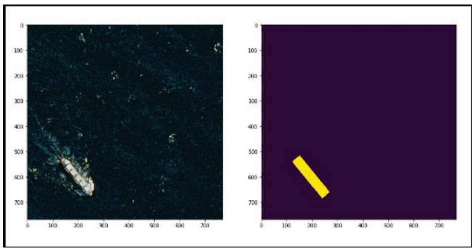


*Fig. 2. Detection of ships [4]*

In the future, it is planned to consider the neural network on the problem of detecting aircrafts on aerial images of airfields. To do this, it is supposed to find relevant data, mark them and submit to the network. The next step is to use vehicle data on aerial images.

## References

1. U-Net: Convolutional Networks for Biomedical Image Segmentation – №1. – [Electronic resource]. URL: https://lmb.informatik.uni-freiburg.de/people/ronneber/u-net.
2. U-Net: neural network for image segmentation (in Russian). – [Electronic resource]. URL: https://neurohive.io/ru/vidy-nejrosetej/u-net-image-segmentation.
3. Image segmentation with networks: U-Net (in Russian). – [Electronic resource]. URL: http://robocraft.ru/blog/machinelearning/3671.html.
4. Keras Based UNet Model Construction Tutorial – №2. – [Electronic resource]. URL: https://www.kaggle.com/krishanudb/keras-based-unet-model-construction-tutorial/notebook.

# THE CHOICE OF DEEP LEARNING METHODS FOR SOLVING THE PROBLEM OF RECOGNIZING PLANT DISEASES FOR CASES OF A SMALL TRAINING SAMPLE

**Smetanin Artem**[1], **Ososkov Gennady**[2], **Goncharov Pavel**[3]

*[1] Student;
Dubna State University;
The International School of Information Technologies
"Data Science", group 21;
The main educational program:
Computer science and engineering, group 4013;
e-mail: webstermaster777@gmail.com.*

*[2] Doctor of Physics and Mathematics, leading researcher;
Laboratory of Information Technology;
Joint Institute for Nuclear Research.
Professor;
Department of System Analysis and Management;
Dubna State University.*

*[3] Postgraduate student;
Department of System Analysis and Management;
Dubna State University.*

Keywords: recognition, artificial neural networks, deep learning, transfer learning.

Loss of yield due to plant diseases is a serious problem for rural farmers, the economy and food security, requiring timely measures to identify and prevent diseases.

This is why the loss of yield due to plant diseases is a serious problem. Recently, neural network methods of deep learning have been successfully applied to solve the problem of recognizing plant diseases from photographs of their leaves [1].

In our study, we applied deep learning methods that performed well on the extra small dataset. For the PDD data (Fig. 1) (http://pdd.jinr.ru) we use the transfer learning technique and the Siamese neural network method with a three-term error function [2]. For base net, we use architecture MobileNetV2 [3], which allows releasing a network on a mobile device. We use KNN as a classifier because this method does not require retraining when adding a new class to the dataset. The method we proposed achieves 99.5% accuracy of the classification [4].

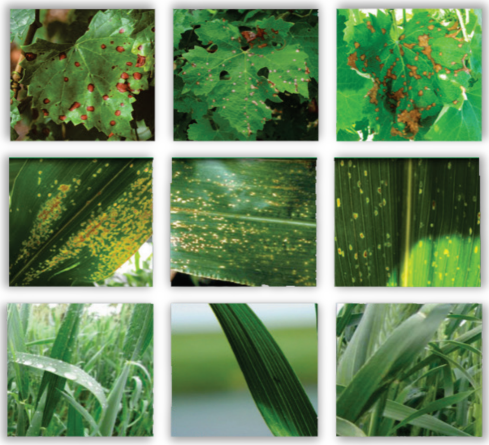


*Fig. 1. Examples from PDD dataset*

In the future, it is planned to add new cultures, as well as improve the accuracy and speed of learning of existing models.

References

1. Goncharov P. et al. Deep Siamese Networks for Plant Disease Detection / Pavel Goncharov, Alexander Uzhinskiy, Gennady Ososkov, Andrey Nechaevskiy and Julia Zudikhina // EPJ Web of Conferences. – EDP Sciences, 2020. – Vol. 226. – P. 03010.
2. Dataset PDDP. – [Electronic resource]: http://pdd.jinr.ru/crops.php.
3. Sandler M. et al. Mobilenetv2: Inverted residuals and linear bottlenecks // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2018. – P. 4510-4520.
4. Repository Plant Disease Detection. – [Electronic resource]: https://github.com/WEBSTERMASTER777/pdd.

# THE PROBLEM OF MEASURING THE PROBABILITY METHOD (LSH) AND THE SCALAR PRODUCT IN THE ANALYSIS OF A LARGE DATA SET

**Tjupin Denis**[1], **Papoyan Vladimir**[2]

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 22;*
*The main educational program:*
*Information systems and technologies, group 4281;*
*e-mail: super.denistjupin2013kraft@yandex.ru.*

[2] *Software engineer;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Postgraduate student;*
*Department of System Analysis and Management;*
*Dubna State University.*

Keywords: LSH, scalar product, big data analysis.

Matching records is a key step in many big data analysis problems, especially when working with information from disparate big data sources. Probabilistic linking methods for records provide a good basis for searching and interpreting partial matches of records. However, calculating the distance between the lines for the compared records requires combining. Direct use of probabilistic linking of records requires processing the Cartesian product of record sets. As a result, the "blocking" stage is used, when pairs of candidate records are grouped by categorical field, which significantly reduces the number of records necessary for work [1].

In this project, we considered the task of replacing the probabilistic method (Locality-sensitive hashing) with a scalar product in comparison in multidimensional data using video cards in the Python programming language [2]. Local hashing is a method of reducing the dimensionality of multidimensional data. In this approach LSH maps multiple points in high-dimensional space to multiple bins (a set of objects) in a hash table. This method has the ability to perceive location (location-dependent hash), due to which it is able to put neighboring points in the same bin [3].

The scalar product acts as a method to speed up the matching of records of multidimensional data. With an increase in data volumes, a method is built in which operations on vectors are performed, resulting in a scalar (number) [4]. This allows you to bring the result of the analysis to "similar" patterns. Within the framework of this model, each record is described by a vector in which each term (information retrieval refers to the words that make up the text) used in the document is associated with its weight value, determined on the basis of statistical information about its appearance. Both in a separate document and in the entire documentary array [5].

Participation in scientific research is completed.

References

1. Noga Alon, Joel H. Spencer 2nd Edition the Probabilistic Method. Wiley-Interscience Series in Discrete Mathematics and Optimization // Binom, 2007.

2. Holden Karau, Rachel Warren. High Performance Spark: Best practices for scaling and optimizing Apache Spark // Piter, 2018.

3. Salton, G., Buckley, C., 1988. Term-weighting approaches in automatic text retrieval. Information Processing & Management 24, 513–523. – [Electronic resource]. URL: https://doi.org/10.1016/0306-4573(88)90021-0.

4. Brown, A.P., Randall, S.M., Ferrante, A.M., Semmens, J.B., Boyd, J.H., 2017. Estimating parameters for probabilistic linkage of privacy-preserved datasets. BMC Med Res Methodol 17. – [Electronic resource]. URL: https://doi.org/10.1186/s12874-017-0370-0.

5. DuVall, S.L., Kerber, R.A., Thomas, A., 2010. Extending the Fellegi-Sunter probabilistic record linkage method for approximate field comparators. J Biomed Inform 43, 24–30. – [Electronic resource]. URL: https://doi.org/10.1016/j.jbi.

# VISUALIZATION OF A MALICIOUS NETWORK ATTACK

**Zizganov Timofei**[1], **Potapov Denis**[2], **Pelevanyuk Igor**[3]

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 21;*
*The main educational program:*
*Software engineering, group 4251;*
*e-mail: timaraylog1998@gmail.com.*

[2] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 21;*
*The main educational program:*
*Design and technology of electronic means, group 4141;*
*e-mail: potapov-deniska@inbox.ru.*

[3] *Software engineer;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Assistant;*
*Department of Distributed Computer Systems;*
*Dubna State University.*

Keywords: network attacks, visualization, server, map, distributed denial of service, attackers.

Attacks on SSH are distributed denial of service attacks, attacks on computing systems designed to make them inaccessible to users. These attacks consist of simultaneously sending a large number of requests from one or many computers towards a certain resource. If tens of thousands or millions of computers simultaneously start sending requests to a specific server, then either the server cannot stand it or the bandwidth of the communication channel to that server is not enough. In both cases, users of this network will not be able to access the attacked server, or even all the servers and other resources connected through the blocked communication channel.

The goal of the joint project is to develop a tool that allows you to visualize malicious network attacks on a specific server.

The following tasks were set.

1. Analyzing of the SSH daemon logs, extracting information from them about under which login and from which IP address the attackers are trying to log into the system.
2. Creating a daemon that can automatically analyze the SSH logs of the daemon and write the results to the ip_address, geo_coordinates, login database.
3. Creating a web page that displays a map of the world and several tables.

On the map, the dot indicates the location of the host. When the page starts, the list of attacks for the last 24 hours is taken. All the points from which attacks were made are placed on the map. Depending on the number of attacks, the line between the source of the attack and our host changes color from yellow to red.

- The first table displays the statistics for which logins are most often used by attackers.
- The second table shows which countries are mainly attacked.
- The third table shows the IP leaders by attack source.

Using Virtual Box [1], a virtual machine was created on the operating system “Ubuntu Server 19.10” [2]. Further, bash scripts were written [3]. Using the lastb command (configured to display log entries, the file /var/log/btmp, which contains records of all failed attempts to register users in the system), it received the IP addresses and login that were in the last day. Next, a GET request was generated for a third-party resource [4], to obtain information about this IP address (country, city, latitude, longitude). After receiving this data, it was entered into the SQLite database [5], which was subsequently used by my colleague Denis for visualization (see Fig. 1). The install.sh script is needed to install the necessary software on the server, and add a script run with the analysis of the log of information about malicious users in crontab (used to manage the cron daemon - a classic daemon used to periodically execute tasks a certain times) configured to run per minute, and the creation of the systemd service (the initialization and service management subsystem in Linux), which runs the Flask [6] web server in the background. The remove.sh script removes the systemd service and automatically starts from cron.

Creating a site using the Flask [6] web framework where the HTML page displays. Thanks to the OpenLayers library [7], the map is linked to the created page. This map displays malicious attacks on our server (see Fig. 1). The list of attackers for visualizing attacks is taken from the SQLite database [5] (file of unsuccessful login attempts db.sqlite). The database is located on the Virtual Box virtual machine [1], which my colleague Timofey deals with. Need to take the coordinates of the attacker (country, city, latitude, longitude) from this database. To do this, select queries are created. The database is connected to the map in the python programming language [8]. Next, the coordinates are applied to the map in latitude (lat) and longitude (lon), a marker is placed to show from which point in the world is the server being attacked from and a line is drawn between the server and the attacker. For these purposes, the JavaScript programming language is used [9]. Also, upon request ./rest/getCountryStatistic, three tables are displayed on the HTML page: Login popularity table, Country popularity table, IP popularity table (see Fig. 2).

The work done is not final. A working prototype is ready [10], all the preparatory work related to the study of individual technologies has been carried out, the repository in git is ready. It requires testing, solving problems related to the fact that the number of requests to the API is limited, a serious refinement of the UI design.

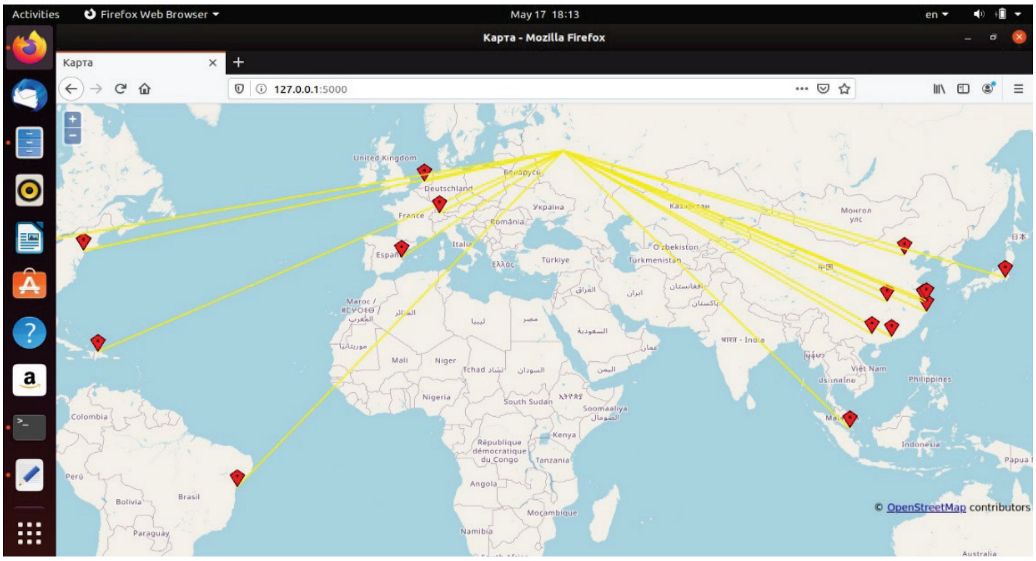


*Fig. 1. Visualization of attacks on the server*

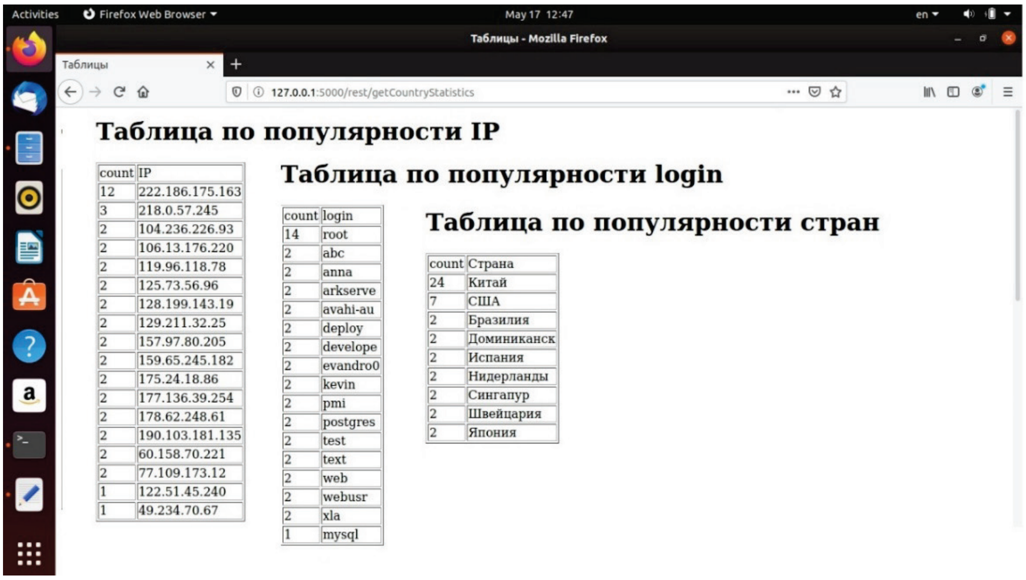

Activities
Firefox Web Browser
May 17 12:47
Таблицы - Mozilla Firefox
Таблицы
127.0.0.1:5000/rest/getCountryStatistics

**Таблица по популярности IP**

| count | IP |
|---|---|
| 12 | 222.186.175.163 |
| 3 | 218.0.57.245 |
| 2 | 104.236.226.93 |
| 2 | 106.13.176.220 |
| 2 | 119.96.118.78 |
| 2 | 125.73.56.96 |
| 2 | 128.199.143.19 |
| 2 | 129.211.32.25 |
| 2 | 157.97.80.205 |
| 2 | 159.65.245.182 |
| 2 | 175.24.18.86 |
| 2 | 177.136.39.254 |
| 2 | 178.62.248.61 |
| 2 | 190.103.181.135 |
| 2 | 60.158.70.221 |
| 2 | 77.109.173.12 |
| 1 | 122.51.45.240 |
| 1 | 49.234.70.67 |

**Таблица по популярности login**

| count | login |
|---|---|
| 14 | root |
| 2 | abc |
| 2 | anna |
| 2 | arkserve |
| 2 | avahi-au |
| 2 | deploy |
| 2 | develope |
| 2 | evandro0 |
| 2 | kevin |
| 2 | pmi |
| 2 | postgres |
| 2 | test |
| 2 | text |
| 2 | web |
| 2 | webusr |
| 2 | xla |
| 1 | mysql |

**Таблица по популярности стран**

| count | Страна |
|---|---|
| 24 | Китай |
| 7 | США |
| 2 | Бразилия |
| 2 | Доминиканск |
| 2 | Испания |
| 2 | Нидерланды |
| 2 | Сингапур |
| 2 | Швейцария |
| 2 | Япония |

*Fig. 2. Tables with statistics*

## References

1. Oracle VM Virtual Box. – [Electronic resource]. URL: https://www.virtualbox.org/.
2. Download Ubuntu Server 20.04. – [Electronic resource]. URL: https://ubuntu.com/download/server.
3. Mendel Cooper. Advanced Bash-Scripting Guide. – [Electronic resource]. URL: https://www.opennet.ru/docs/RUS/bash_scripting_guide/index.html.
4. IP Geolocation API. – [Electronic resource]. URL: https://ip-api.com/.
5. SQLite Documentation. – [Electronic resource]. URL: https://www.sqlite.org/docs.html.
6. Flask Documentation. – [Electronic resource]. URL: https://flask.palletsprojects.com/en/1.1.x/.
7. OpenLayers Documentation. – [Electronic resource]. URL: https://openlayers.org/en/latest/doc/.
8. Python Documentation. – [Electronic resource]. URL: https://docs.python.org/3/.
9. JavaScript Documentation. – [Electronic resource]. URL: https://documentation.js.org/.
10. Our project. – [Electronic resource]. URL: https://github.com/Tima-lab/DDOS-monitoring.

# COMPUTING & SOFTWARE FOR THE EXPERIMENTS AT THE NICA ACCELERATOR COMPLEX

# EXTENDING THE FUNCTIONALITY OF THE CERN ROOT PACKAGE FOR WORKING WITH DATA OF ORACLE DATABASE FOR THE TIMESTAMP FORMAT

**Artemiev Alexey[1], Gertsenberger Konstantin[2]**

*[1] Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 21;*
*The main educational program:*
*Automation of technological processes and production, group 2231;*
*e-mail: fizz4ever1337@gmail.com.*

*[2] Candidate of technical sciences,*
*Scientific and experimental department of heavy-ion collision physics at the NICA complex,*
*Head of the group of mathematical support and software;*
*Laboratory of High Energy Physics;*
*Joint Institute for Nuclear Research.*

Keywords: Oracle, TIMESTAMP, ROOT.

The goal of this project is to expand the capabilities of the latest version of the CERN ROOT 6 package for interacting with data of the TIMESTAMP temporary format from Oracle database. The current functionality of the ROOT 6 framework for working with this format has the following limitations:

1. The size of the buffer string does not allow to store the time zone and microseconds.
2. It is not possible to return TTimeStamp values from the database, so there is no functionality for getting and recording time with accuracy of fractions of seconds.
3. Writing and reading time format data with a time zone does not work correctly.

The first step was the deployment of the CentOS 8 operating system with installation and configuration of CERN ROOT package version 6.20 / 04 and the Oracle database version 19.3 on it [1].

At the next stage the capabilities of the ROOT 6 package for working with temporary data stored in the Oracle database were studied, and the capabilities of the database itself to store, read and write TIMESTAMP data were also studied, and a test database for the future verification of the functionality of the CERN ROOT package on practice was deployed too [2].

After studying the modules of the ROOT 6 package, such as TOracleStatement, which provide tools for working with Oracle database, it was revealed that the buffer string was implemented correctly (p. 1), since it originally implemented a dynamic string buffer [3].

We are currently working on the connecting to the Oracle database with the tools provided by the TOracleServer module of the CERN ROOT package for subsequent verification of the limitations described above, as well as to detect new ones [3].

Upon completion of work on expanding the capabilities of this package with the Oracle database, the new goal would proceed to the adjustment and implementation of these capabilities in the remaining database supported by ROOT 6 such as MySQL and SQLite.

References

1. Centos 8 // Operating system image and recommendation for installation. – 2020. – [Electronic resource]. URL: https://www.centos.org/.
2. Oracle database documentation // Database Installation Instructions for Linux. – 2020. – [Electronic resource]. URL: https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/.
3. Source code and documentation of the CERN ROOT // Master branch source code. – 2020. – [Electronic resource]. URL: https://root.cern.ch/doc/master/.

# ADAPTATION OF THE SERVER COMPONENT OF THE PANDA LOAD MANAGEMENT SYSTEM FOR INTEGRATION WITH THE DATA PROCESSING MANAGEMENT SYSTEM FOR THE BM@N EXPERIMENT

**Gavrilov Dmitry**[1], **Petrosyan Artem**[2], **Oleynik Danila**[3]

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 21;*
*The main educational program:*
*Software engineering, group 4254;*
*e-mail: dmitriygavrofficial@gmail.com.*

[2] *Lead programmer;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*

[3] *Lead programmer;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*

Keywords: PanDA, PandaServer, workflow.

The BM@N data processing system [1] consists of several components that interact with each other:

- workflow management system;
- data management system;
- workload management system;
- data transfer service.

The load management system is necessary for organizing work with different types of computing resources [2].

Goals of using the load management system:

- unify the access interface to various computing resources;
- optimization of resource loading tasks;
- priority task management.

PanDA (Production and Distributed Analysis System) software was chosen as the load management system (Fig. 1) [3].

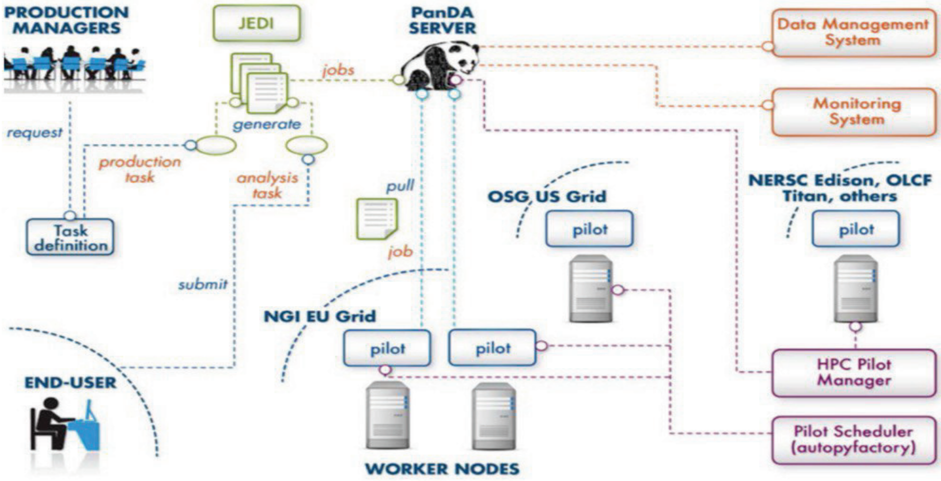


*Fig. 1. PanDA*

PandaServer (the core where the main PanDA operations are performed) was installed and configured, a database for PandaServer was created, and the necessary components such as httpd, httpd-devel, mod_ssl, and gridsite were installed. Adding tasks was tested.

References

1. Petrosyan A. Workflow Services for distributed processing BM@N data. – [Electronic resource]. URL: https://indico.jinr.ru/event/1159/contributions/9022/.
2. Oleynik D. Automation of (big) data processing for scientific research in heterogeneous distributed computing systems. Lessons of BigPanDA project. – [Electronic resource]. URL: https://indico.jinr.ru/event/738/contributions/6446/.
3. The PanDA Production and Distributed Analysis System. – [Electronic resource]. URL: https://twiki.cern.ch/twiki/bin/view/PanDA/PanDA.

# DEVELOPMENT OF MONITORING SERVICE FOR THE BM@N EXPERIMENT DATABASE USING GRAFANA PACKAGE

**Kuzmenkov Igor**[1], **Gertsenberger Konstantin**[2]

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 21;*
*The main educational program:*
*Computer science and engineering, group 4012;*
*e-mail: ivt2018tms16@gmail.com.*

[2] *Candidate of technical sciences,*
*Scientific and experimental department of heavy-ion collision physics at the NICA complex,*
*Head of the group of mathematical support and software;*
*Laboratory of High Energy Physics;*
*Joint Institute for Nuclear Research.*

Keywords: monitoring system, metrics, visualization, database, PostgreSQL, Grafana, InfluxDB, Telegraf.

The goal of this work is to implement monitoring and visualize metrics from the database and its server dedicated to BM@N experiment which is undergoing in the NICA project. This database stores information about experiment runs, detectors and their geometry, configuration, calibration and other parametric data, which is used for offline data processing of the experiment. The database management system chosen for this project is PostgreSQL [1].

Necessity of such web-monitoring system is dictated by the reliability demands placed upon the software used in the experiment for the purpose of real time analysis of system's performance indicators by means of a user interface which is constructed using Grafana software package [2], which, in case of emergency situations, automatically alerts people who are responsible for the management of technical issues. The journal of system's metrics being saved in the dedicated time series database InfluxDB [3] after being gathered with metrics collector service Telegraf [4] will provide the data for system's performance and condition analysis throughout the gathering procedure, allowing to detect the cause of exceptional situation, which is used for further improvement and development of the informational system of the experiment built on the aforementioned database.

Metrics gatherer service Telegraf allows retrieval of information about both the server where the experiment database is deployed, as well as statistics on the operation of the database itself. Acquired metrics are those which we decided are being indicative of the system's performance and workflow, such constraint is being achieved through the SQL queries in the Telegraf configuration file.

Throughout this work a virtual machine has been instantiated running CentOS 7, on which the software solutions from InfluxData were installed, such as InfluxDB and Telegraf, that take hold of gathering and storage of time series array of metrics for the purpose of usage of such data in the monitoring system. Also, on the same virtual machine was installed service for time series data visualisation — Grafana. Such combination of software proved its dependability and relative ease of deployment.

As a result of our work acquired metrics from the experiment's database and its server are being visualized with Grafana web-service. This solution allows for evaluation of the working load the system is undergoing which includes CPU usage of the server, information about data storage including amount and speed of read-write operations, as well as evaluation of index efficiency. Moreover, such a monitoring system provides history of database's performance indicators and their change by means of SQL queries and visualizes retrieved metrics.

This means, that the system being developed will resolve the monitoring and evaluation of working load issues allowing for the possibility of choosing the interval at which the data is being gathered to achieve desired detalization, also suggest the possibility of analysis of database's inner workings, statistics of index usage, access, read and write loads on the storage element for the database as well as the server it is located on. The solution developed allows for optimization and debugging of database's functioning, resolution of arising issues in almost real time. Further work is consisting of analysis of provided by default metrics from PostgreSQL database’s metrics collector, from those we'll pick ones that are required for monitoring and visualize them in Grafana web-service of Joint Institute for Nuclear Research.

## References

1. PostreSQL // 27.2. The Statistics Collector. – [Electronic resource]. URL: https://www.postgresql.org/docs/current/monitoring-stats.html.
2. Grafana // Install on RPM-based Linux (CentOS, Fedora, OpenSuse, Red Hat). – [Electronic resource]. URL: https://grafana.com/docs/grafana/latest/installation/rpm/.
3. InfluxData // Installing InfluxDB. Installing, starting, and configuring InfluxDB open source (OSS). – [Electronic resource]. URL: https://docs.influxdata.com/influxdb/v1.8/introduction/install/.
4. InfluxData // Installing Telegraf. Installing, starting, and configuring Telegraf. – [Electronic resource]. URL: https://docs.influxdata.com/telegraf/v1.14/introduction/installation.

# ADAPTATION/DEVELOPMENT OF AN INFORMATION SYSTEM AS PART OF A DISTRIBUTED DATA PROCESSING SYSTEM OF THE BM@N EXPERIMENT

**Matveev Ivan**[1], **Oleynik Danila**[2], **Petrosyan Artem**[3]

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 22;*
*The main educational program:*
*Applied informatics, group 4071;*
*e-mail: matveev.a.ivan@yandex.ru.*

[2] *Lead programmer;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*

[3] *Lead programmer;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*

Keywords: unified, resource, management, system, baryonic, matter, nuclotron, computing resource, information, catalogue.

A unified resource management system is an IT ecosystem consisting of a set of subsystems and services that should unify access to data and computing resources in a heterogeneous distributed environment, automate most of the operations associated with mass data processing, and avoid duplication of basic functions by sharing different systems by users, as a result - reduce operating costs, increase the efficiency of resource use, ensure transparent accounting of resources (see Fig. 1) [1].

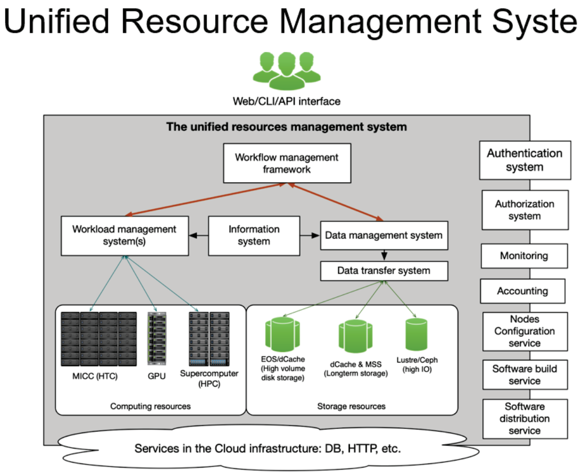

Unified Resource Management System
Web/CLI/API interface
The unified resources management system
Workflow management framework
Workload management system(s)
Information system
Data management system
Data transfer system
MICC (HTC)
GPU
Supercomputer (HPC)
Computing resources
EOS/dCache (High volume disk storage)
dCache & MSS (Longterm storage)
Lustre/Ceph (high IO)
Storage resources
Services in the Cloud infrastructure: DB, HTTP, etc.
Authentication system
Authorization system
Monitoring
Accounting
Nodes Configuration service
Software build service
Software distribution service

*Fig. 1. Unified Resource Management System schema*

The main objective of the project is the selection and development of an information system within the framework of a unified resource management system for experiments at the Nuclotron-based Ion Collider fAcility (NICA) installation, which includes the Baryonic Matter at Nuclotron (BM@N) experiment [2, 3]. To solve this problem, the Computing Resource Information Catalog (CRIC) [4] information system was chosen (see Fig. 2).

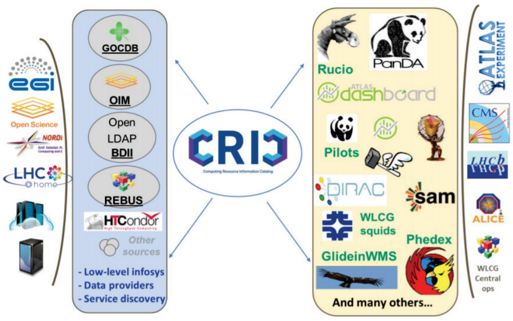


*Fig. 2. Computing Resource Information Catalogue*

A test version of the CRIC system was deployed on the cloud infrastructure resources of LIT JINR. The work on the project is going to continue. The next step is to determine the formats for exporting data to other systems.

## References

1. Oleynik D. Automation of (big) data processing for scientific research in heterogeneous distributed computing systems. – [Electronic resource]. URL: https://indico.jinr.ru/event/738/contributions/6446/attachments/4959/6533/NEC2019.pdf.
2. Nuclotron-based Ion Collider fAcility. – [Electronic resource]. URL: https://nica.jinr.ru/ru/.
3. BM@N experiment. – [Electronic resource]. URL: https://bmn.jinr.ru/about/.
4. Anisenkov A. CRIC: a unified information system for WLCG and beyond. – [Electronic resource]. URL: http://ceur-ws.org/Vol-2023/1-5-paper-1.pdf.

# NEURAL NETWORK APPROACH APPLICATION FOR TRACK RECONSTRUCTION TASKS OF THE NICA PROJECT MPD EXPERIMENT

**Makhlov Egor**[1], **Streltsova Oksana**[2]

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 21;*
*The main educational program:*
*Software engineering, group 4251;*
*e-mail: fictioncentralacc@gmail.com.*

[2] *Candidate of Physics and Mathematics, senior researcher;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Associate Professor;*
*Department of Distributed Computer Systems;*
*Dubna State University.*

Keywords: neural networks, training of neural networks, inference, neural networks, computing architectures, reconstruction of particle tracks, NICA, MPD, HEP.

One of the most important problems in processing data from experiments in high-energy physics is the reconstruction of particle tracks, which consists in constructing particle tracks (trajectories) based on the set of spatial hits, called hits, of these particles in the sensitive layers of the detector. Modern experimental conditions for high-energy physics dictate the need to find new methods of data processing, since there can be a significant gap between the classical methods that have been used for decades and the optimal ones. To solve this problem, various research groups are created [1].

This work was carried out as part of the JINR research group, which is engaged in the search, research and creation of data processing methods for the MPIC NICA experiment in collaboration with RSK Technologies.

The aim of this work is to test computational architectures for the problems of recognition of particle tracks using the neural network approach. The first step to achieve this goal was to identify and describe neural network approaches used for track reconstruction developed by other research groups. Recursive architectures are fundamental in reconstruction problems, since they are able to process sequences [1] [2]. There are applications of convolutional architectures with some restrictions [3]. Solutions are also being developed on less well-known [1].

The second step, for direct testing of computational architectures, was to use models of convolutional and recurrent architecture. The study was conducted on the computing devices: Intel Xeon 7680, Intel Xeon Phi 6048 and Nvidia Tesla K100. Only one device was used for computing learning and output, and all the models were written using the Keras open neural network library. Below are the results of training and networks inference.

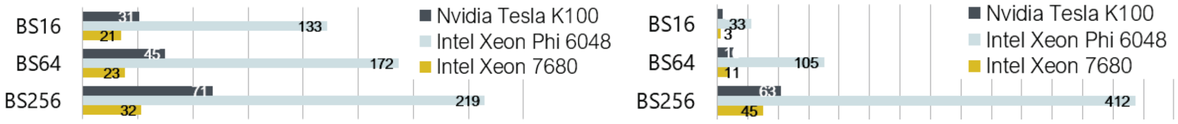


*Fig. 1. On the left figure is the median ms per step for teaching LSTM architecture; on the right figure is the median ms per step for learning CNN architecture. It is important to note that during training, the samples_per_epoch parameter was used, in which the number of samples processed for each epoch is equal to batch_size multiplied by steps_per_epoch, therefore increasing batch_size increases the calculation time, but not reduces it.*

For the inference of the LSTM network, the best result was obtained on the Xeon 7680 (139 µs), followed by the Tesla K100 (221 µs). The worst result with the Xeon Phi 6048 (1 ms). For the logical output of the CNN network, the gradation is identical: Xeon 7680 (36 µs), Tesla K100 (63 µs), Xeon Phi 6048 (281 µs).

The results are not final, since the study of acceleration from the use of various computational architectures was carried out on a simulation of real neural network models that are not used in experiments, and data volumes that are remote from the volumes of modern experiments. In the future, it is planned to study the logical inference and training of neural network models used to process large amounts of data from a particular experiment.

## References

1. Dan Guest, Kyle Cranmer, Daniel Whiteson. Deep Learning and Its Application to LHC Physics // arXiv:1806.11484v1 [hep-ex] 29 Jun 2018.
2. Steven Farrell, Dustin Anderson, Paolo Calafiura, Giuseppe Cerati. The HEP.TrkX Project: deep neural networks for HL-LHC online and offline tracking // Connecting the Dots/Intelligent Trackers 2017.
3. Dmitriy Baranov, Sergey Mitsyn, Pavel Goncharov, Gennady Ososkov. The particle track reconstruction based on deep neural networks // JINR 2018.

# NUMERICAL ANALYSIS OF THE SCATTERING PROCESSES AT FINITE TEMPERATURE AND DENSITY

**Rogozhina Elizaveta**[1], **Kalinovsky Yuri**[2], **Golyatkina Lubov**[3]

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 22;*
*The main educational program:*
*Information systems and technologies, group 4281;*
*e-mail: liorinoff@mail.ru.*

[2] *Doctor of Physics and Mathematics, leading researcher;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*
*Associate Professor;*
*Head of the Department of Higher Mathematics;*
*Dubna State University.*

[3] *Student;*
*Department of Information Technologies;*
*Dubna State University.*

Keywords: nuclear physics, NICA, Feynman diagrams, particle scattering, pion, meson.

The aim of this work is to perform calculations of cross-sections of particle scattering at the final temperature and density for the NICA project.

The main task is to create analytical code in Wolfram Mathematica, convert this result to C++ code, and place this code on HybriLIT to help calculate the NICA project experiments.

The main scattering processes are determined by Feynman diagrams of two types: boxes and diagrams of the birth of an intermediate meson, or, as they are also called, triangles [1].

To calculate the pion-pion scattering, we used "Box" and "Triangle" diagrams [1] (see Fig. 1).

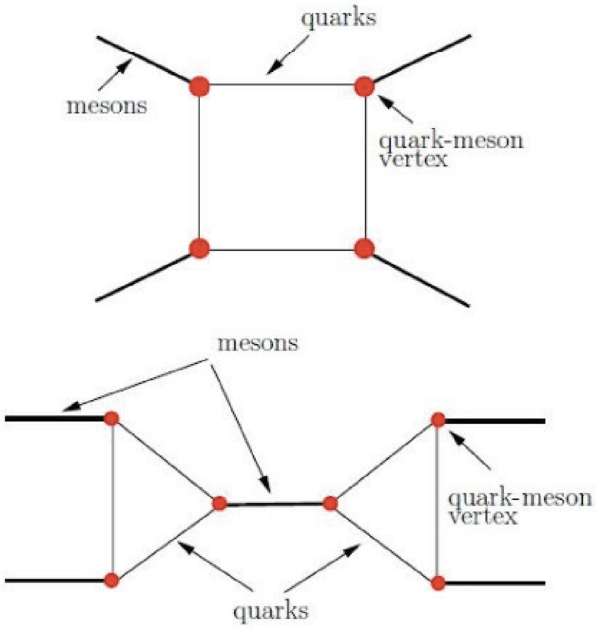


*Fig. 1. "Box" and "Triangle" Diagrams*

For calculations, we use Wolfram Mathematica with Package-X and LoopTools to calculate some loop integrals [2].

At the moment, we have performed analytical calculations of amplitudes and meson scattering cross sections and developed C++ code to help us calculate the experiments of the NICA project.

In the future, we want to apply this approach to gluodynamics and study the processes of gg – $\pi\pi$.

References

1. Kalinovsky Yu.L., Toneev V.D., Friesen A.V. Phase diagram of baryon matter in the SU(2) Nambu – Jona-Lasinio model with a Polyakov loop, UFN, 186 (4) 2016. Joint Institute for Nuclear Research.
2. Hiren H. Patel Package-X. – [Electronic resource]. URL: https://packagex.hepforge.org.

# DEVELOPMENT OF A WORKFLOW MANAGEMENT SYSTEM BASED ON THE BM@N EXPERIMENT

**Yachmenyov Andrew[1], Petrosyan Artem[2], Oleynik Danila[3]**

[1] *Student;*
*Dubna State University;*
*The International School of Information Technologies*
*"Data Science", group 21;*
*The main educational program:*
*Software engineering, group 4254;*
*e-mail: andrew91.99@yandex.ru.*

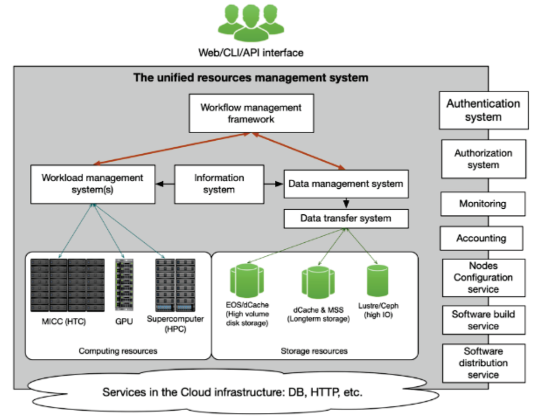
[2] *Lead programmer;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*

[3] *Lead programmer;*
*Laboratory of Information Technology;*
*Joint Institute for Nuclear Research.*

Keywords: airflow, luigi, workflow management, BM@N.

The Unified Resource Management System [1] is an IT ecosystem composed from the set of subsystem and services (see Fig. 1) which should:

- unify of access to the data and compute resources in a heterogeneous distributed environment;
- automate most of the operations related to massive data processing;
- avoid duplication of basic functionality, through sharing of systems across different users (if it possible);
- as a result - reduce operational cost, increase the efficiency of usage of resources;
- transparent accounting of usage of resources.


Web/CLI/API interface
The unified resources management system
Workflow management framework
Workload management system(s)
Information system
Data management system
Data transfer system
Authentication system
Authorization system
Monitoring
Accounting
Nodes Configuration service
Software build service
Software distribution service
MICC (HTC)
GPU
Supercomputer (HPC)
Computing resources
EOS/dCache (High volume disk storage)
dCache & MSS (Longterm storage)
Lustre/Ceph (high IO)
Storage resources
Services in the Cloud infrastructure: DB, HTTP, etc.

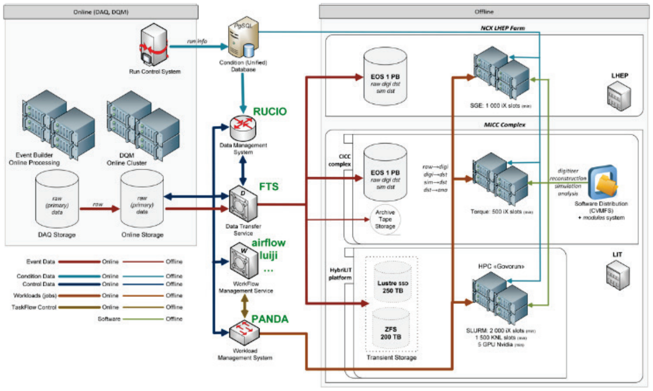


*Fig. 1. Unified Resource Management System*    *Fig. 2. Automation of BM@N reconstruction workflow*

The BM@N data processing system consists of several components [1] that interact with each other (see Fig. 2).

The goal of my project is to select and configure a workflow management framework.

A workflow management system is necessary to describe pipelines in a convenient way, run pipelines, monitor dependencies, and monitor the progress pipelines.

We decided to use an open source tool that is actively supported by the community to minimize the effort to develop and maintain the system.

Two systems were selected for this purpose: luigi [2] and airflow [3]. These are tools for building, managing, and monitoring pipelines.

A comparison of these tools showed that airflow is better suited for building a workflow management system, since it is an extensible and more flexible solution.

At the current stage of building the system, the environment for creating the system is configured, and Airflow is deployed. We are working on creating an extension for interaction with the PanDA workload management system. After configuring the system, we have to describe the task sequences in the Python programming language.

## References

1. Oleynik D. Automation of (big) data processing for scientific research in heterogeneous distributed computing systems. – [Electronic resource]. URL: https://indico.jinr.ru/event/738/contributions/6446/attachme nts/4959/6533/NEC2019.pdf.
2. Luigi docs. – [Electronic resource]. URL: https://luigi.readthedocs.io/en/stabl.
3. Airflow docs. – [Electronic resource]. URL: https://airflow.apache.org/docs/stable/.

*Scientific publication*

# Collection of scientific and project activity reports of the graduates of the International School of Information Technologies "Data Science"

## Collection of works
## Volume 1

Editors:
Vladimir Korenkov, Eugenia Cheremisina,
Oksana Streltsova, Daria Priakhina

Author's edition

Approved for printing on 31 May 2020.
Format 60x90/8. Conventional printed sheet 6.25.
Circulation 100 copies. Order No. 11

Printed at Diverprint, LLC
21, Architect Vlasov st., Moscow, 117335, Russia