ИНСТИТУТ СИСТЕМНОГО АНАЛИЗА И УПРАВЛЕНИЯ

СБОРНИК ОТЧЕТОВ О НАУЧНО-ПРОЕКТНОЙ ДЕЯТЕЛЬНОСТИ ВЫПУСКНИКОВ МЕЖДУНАРОДНОЙ ШКОЛЫ ПО ИНФОРМАЦИОННЫМ ТЕХНОЛОГИЯМ «АНАЛИТИКА БОЛЬШИХ ДАННЫХ»

DATA SCIENCE
IT SCHOOL

Министерство образования Московской области

Государственный университет «Дубна»

Объединенный институт ядерных исследований

СБОРНИК ОТЧЕТОВ
О НАУЧНО-ПРОЕКТНОЙ ДЕЯТЕЛЬНОСТИ ВЫПУСКНИКОВ МЕЖДУНАРОДНОЙ ШКОЛЫ ПО ИНФОРМАЦИОННЫМ ТЕХНОЛОГИЯМ «АНАЛИТИКА БОЛЬШИХ ДАННЫХ»

Сборник трудов

ВЫПУСК 1

Под редакцией В.В. Кореньков, Е.Н. Черемисиной, О.И. Стрельцовой, Д.И. Пряхиной

DATA SCIENCE
IT SCHOOL

ДУБНА

2020

УДК 004.42
ББК 32.97я43
С 232

С 232 **Сборник отчетов о научно-проектной деятельности выпускников Международной школы по информационным технологиям «Аналитика больших данных»** : сборник трудов. Выпуск 1 / под ред. В.В. Коренькова и др. – Дубна : Гос. ун-т «Дубна», 2020. – 52 с.

ISBN 978-5-89847-609-0

Сборник включает в себя краткие отчеты об итоговой работе студентов государственного бюджетного образовательного учреждения высшего профессионального образования Московской области «Университет «Дубна», завершивших обучение в Международной школе по информационным технологиям «Аналитика больших данных».

Студенты в период обучения были включены в реальные перспективные проекты Объединенного института ядерных исследований (ОИЯИ, Дубна, Россия), работу в которых учащиеся осуществляли в рамках дисциплины «Научно-проектная деятельность» под руководством сотрудников ОИЯИ и университета «Дубна».

УДК 004.42
ББК 32.97я43

ISBN 978-5-89847-609-0

СОДЕРЖАНИЕ

ПРЕДИСЛОВИЕ

Международная школа по информационным технологиям «Аналитика больших данных» (далее ИТ-школа) — совместный образовательный проект Лаборатории информационных технологий (ЛИТ) Объединенного института ядерных исследований (ОИЯИ) и Института системного анализа и управления (ИСАУ) государственного университета «Дубна», целью которого является подготовка высококвалифицированных ИТ-специалистов для развития компьютинга мегапроектов, аналитики больших данных, цифровой экономики и других перспективных направлений.

Образовательная программа ИТ-школы формируется с учетом кадровых потребностей ОИЯИ и других организаций высокотехнологичного сектора экономики, а также реализуется при их участии. Программа включает изучение таких дисциплин, как:

- Дополнительные главы математики;
- Языки программирования для анализа данных;
- Введение в операционные системы UNIX;
- Инструментарий для коллективной разработки программного обеспечения;
- Введение в облачные технологии;
- Аналитика больших данных;
- Распределенные системы;
- Мультиагентные системы;
- Высокопроизводительные вычисления;
- Английский язык в профессиональной деятельности.

Практические занятия проводятся в компьютерных аудиториях университета «Дубна» с задействованием, в том числе, ресурсов гетерогенной вычислительной платформы *HybriLIT*, которая является частью Многофункционального информационно-вычислительного комплекса ЛИТ ОИЯИ.

В организации учебного процесса и создание программно-информационной среды принимаю участие сотрудники университета «Дубна» и сотрудники ОИЯИ.

В феврале 2019 года впервые состоялся конкурсный отбор студентов университета «Дубна», желающих поступить в ИТ-школу. С 1 марта 2019 года началось обучение, которое является бесплатным. Образовательная программа ИТ-школы осваивается студентами параллельно с основной образовательной программой.

ИТ-школа тесно взаимодействует в своей деятельности с ведущими университетами России, которые готовят квалифицированных ИТ-специалистов. Поэтому за время обучения студенты получили знания и компетенции в области современного компьютинга и аналитики больших данных не только от преподавателей университета «Дубна» и сотрудников ОИЯИ. Лекции также читали преподаватели из таких университетов России, как Национальный

исследовательский ядерный университет «МИФИ», Российский экономический университет им. Г.В. Плеханова и др.

Помимо этого, студенты посещали лекции и семинары от ведущих специалистов российских компаний и зарубежных организаций на английском языке:

- лекция на тему "*Deep and Machine Learning methods for document clustering and classification*" по глубокому и машинному обучению от специалиста по анализу данных компании *SAP SE* (Германия), на базе ЛИТ ОИЯИ (17.04.2019);
- семинар на тему «Архитектуры и технологии *Intel* для высокопроизводительных вычислений и задач машинного/глубокого обучения (*ML/DL*)» от специалистов компаний *Intel* и РСК, на базе ЛИТ ОИЯИ (15.11.2019);
- семинар по высокопроизводительным вычислениям на базе Национального исследовательского университета «Высшая школа экономики» (21.01.2020).

Студенты ИТ-школы, прошедшие серьезный конкурсный отбор, принимали активное участие в студенческих образовательных и научных мероприятиях, в том числе зарубежных:

- Международная ИТ-школа *"Machine Learning, Parallel and Hybrid Computations & Big Data Analytics"* в рамках Международной конференции *"Mathematical Computational Physics – MMCP'2019"* (1-5 июля 2019 г., Словакия);
- Летняя компьютерная школа «Аналитика Больших данных Дубна-2019» (6-13 июля 2019 г., Университет «Дубна», Дубна, Россия);
- Международная школа "*Big Data mining and distributed systems*" в рамках международной конференции "*Symposium on Nuclear Electronics and Computing – NEC'2019*" (29 сентября – 3 октября 2019 г., Черногория);
- Школа молодых ученых «Высокопроизводительные платформы для цифровой экономики и научных проектов класса мегасайенс» (3-4 декабря 2019 г., РЭУ им. Г.В. Плеханова, Москва, Россия);
- XXVII научно-практическая конференция студентов, аспирантов и молодых специалистов (14-27 апреля 2020 г., Университет «Дубна», Дубна, Россия).

В июне 2020 года состоялся первый выпуск студентов ИТ-школы. Выпускники в количестве 21 человек успешно завершили обучение и получили документы о дополнительном образовании, удостоверяющие прохождение профессиональной подготовки по программе «Аналитика больших данных».

Одним из главных принципов ИТ-школы является обучение через исследования. Поэтому студенты в период обучения были включены в реальные перспективные проекты ОИЯИ, работу в которых учащиеся осуществляли в рамках дисциплины «Научно-проектная деятельность». В данном сборнике представлены краткие отчеты об их деятельности.

Дирекция ИТ-школы выражает огромную благодарность преподавателям университета «Дубна» и сотрудникам ОИЯИ за плодотворную работу со студентами.

С учащимися ИТ-школы работали:

- Балашов Н.А., инженер-программист ЛИТ ОИЯИ;
- Белов С.Д., ведущий программист ЛИТ ОИЯИ;
- Герценбергер К.В., к.т.н., научно-экспериментальный отдел физики столкновений тяжелых ионов на комплексе *NICA*, начальник группы математического и программного обеспечения ЛФВЭ ОИЯИ;
- Зрелов П.В., к.ф.-м.н., начальник научно-технического отдела программного и информационного обеспечения ЛИТ ОИЯИ;
- Кадочников И.С., инженер-программист ЛИТ ОИЯИ;
- Калиновский Ю.Л., д.ф.-м.н., ведущий научный сотрудник ЛИТ ОИЯИ;
- Кошлань Д.И., инженер-программист ЛИТ ОИЯИ;
- Мещерская Ю.В., доц. каф. САУ ИСАУ университета «Дубна»;
- Олейник Д.А., ведущий программист ЛИТ ОИЯИ;
- Ососков Г.А., д.ф.-м.н., главный научный сотрудник ЛИТ ОИЯИ;
- Папоян В.В., инженер-программист ЛИТ ОИЯИ;
- Пелеванюк И.С., инженер-программист ЛИТ ОИЯИ;
- Петросян А.Ш., ведущий программист ЛИТ ОИЯИ;
- Пятков Ю.В., д.ф.-м.н., ведущий научный сотрудник ЛЯР ОИЯИ;
- Стадник А.В., к.ф.-м.н., инженер-программист ЛИТ ОИЯИ;
- Сычев П.П., доц. каф. РИВС ИСАУ университета «Дубна»;
- Kullenberg Ch., научный сотрудник ЛЯП ОИЯИ.

Научные руководители ИТ-школы:

***В.В. Кореньков**, д.т.н., директор ЛИТ ОИЯИ, заведующий каф. РИВС ИСАУ;*

***Е.Н. Черемисина**, д.т.н., профессор, академик РАЕН, директор ИСАУ.*

Директор ИТ-школы:

***О.И. Стрельцова**, к.ф.-м.н., старший научный сотрудник ЛИТ ОИЯИ, доц. каф. РИВС ИСАУ.*

Ученый секретарь ИТ-школы:

***Д.И. Пряхина**, инженер-программист ЛИТ ОИЯИ, ст. преп. каф. РИВС ИСАУ.*

itschool.jinr.ru

ВЫПУСКНИКИ 2020 ГОДА

1. Артемьев Алексей Владимирович
2. Габдрахимов Даурен Куанышкалиевич
3. Гаврилов Дмитрий Иванович
4. Жаткина Кристина Николаевна
5. Зизганов Тимофей Андреевич
6. Ильина Анна Владимировна
7. Кисеева Виктория Ильинична
8. Костоправов Антон Александрович
9. Кузьменков Игорь Викторович
10. Матвеев Иван Андреевич
11. Махлов Егор Вячеславович
12. Полонский Денис Дмитриевич
13. Постолов Илья Станиславович
14. Потапов Денис Сергеевич
15. Резвая Екатерина Петровна
16. Рогожина Елизавета Дмитриевна
17. Руденко Михаил Олегович
18. Рябов Андрей Русланович
19. Сметанин Артем Алексеевич
20. Тюпин Денис Николаевич
21. Ячменёв Андрей Алексеевич

ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ

РАЗРАБОТКА WEB-ИНТЕРФЕЙСА НОВОСТНОГО АГРЕГАТОРА ПО ТЕМАТИЧЕСКОМУ НАПРАВЛЕНИЮ «ОБЛАЧНЫЕ ТЕХНОЛОГИИ»

Габдрахимов Даурен Куанышкалиевич[1], Кошлань Диана Игоревна[2]

[1] Студент;
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 21;
Направление обучения по основной образовательной программе:
Программная инженерия, группа 4253;
e-mail: daur9613@gmail.com.

[2] Инженер-программист;
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Аспирант;
Кафедра системного анализа и управления;
Государственный университет «Дубна».

Ключевые слова: агентные технологии, поиск и обработка информации, разработка *web*-интерфейса.

В современном мире наблюдается значительный прирост количества источников научно-технической информации в сети Интернет, что затрудняет ее качественную обработку учеными. В связи с этим возникает необходимость создания новостного агрегатора для оперативного ознакомления сотрудников Лаборатории информационных технологий Объединенного института ядерных исследований с новыми публикациями по направлению «Облачные технологии».

Целью проекта является разработка *web*-интерфейса новостного агрегатора по тематическому направлению «Облачные технологии».

Система работает следующим образом: агентом производится сбор данных с более чем 100 источников авторитетных изданий в области информационных технологий. Агрегирование новостного материала производится в централизованную базу данных [1]. Для реализации цели данного проекта была решена задача создания сайта, отображающего новостную информацию, собранную агентом.

В результате работы изучены технологии создания сайтов с использованием фреймворка *Django* [2], разработана система хранения данных при помощи *Django ORM*, реализованы дизайн интерфейса новостного агрегатора и выгрузка интересующей информации из базы данных агента на *web*-страницу (см. рис. 1). Разработка сайта проведена на языке программирования *Python* с помощью *Django*. Для сбора и обработки информации применяются библиотеки *Requests, Beautiful Soup, Xpath* [3].

Новостной агрегатор

Начальная страница | Контакты

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

Рис. 1. Web-интерфейс новостного агрегатора

Дальнейшими действиями по проекту являются реализация расписания работы новостного агрегатора и публикация сайта-новостного агрегатора в Интернете при помощи *apache*-сервера для использования облачной командой Лаборатории информационных технологий [4].

Список литературы

1. *Кошлань Д.И., Третьяков Е.С., Кореньков В.В., Оныкий Б.Н., Артамонов А.А. Мультиагентная информационно-аналитическая система по тематическому направлению «Облачные технологии» // Математика. Компьютер. Образование.: межд. конф. (Дубна, 27 января – 1 февраля 2020). Ижевск: Изд-во АНО «Ижевский институт компьютерных исследований», 2020. С. 198.*
2. *Веб-фреймворк Django (Python). – 2020. – [Электронный ресурс]. URL: https://developer.mozilla.org/ru/docs/Learn/Server-side/Django.*
3. *Web Scraping с помощью python. – 2016. – [Электронный ресурс]. URL: https://habr.com/ru/post/280238/.*
4. *Apache HTTP Server. – 2010. – [Электронный ресурс]. URL: https://help.ubuntu.ru/wiki/apache2.*

ПРИМЕНЕНИЕ МЕТОДОВ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ ДЛЯ ЦИФРОВОЙ ОБРАЗОВАТЕЛЬНОЙ ПЛАТФОРМЫ

Жаткина Кристина Николаевна[1], Стрельцова Оксана Ивановна[2]

[1] Студент;
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 22;
Направление обучения по основной образовательной программе:
Системный анализ и управление, группа 6014;
e-mail: zhatkina-96@mail.ru.

[2] к.ф.-м.н., старший научный сотрудник;
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Доцент;
Кафедра распределенных информационно-вычислительных систем;
Государственный университет «Дубна».

Ключевые слова: агент, нейронная сеть, цифровая образовательная платформа.

В настоящее время существует множество электронных образовательных платформ, которые предлагают различные курсы по всевозможным тематикам. В связи с последними события все больше становится открытых бесплатных курсов на каждой из платформ. Каждый курс в свое очередь имеет схожие элементы на большинстве популярных платформ: название, описание, ссылка на структуру курса. Поэтому было принято решение проанализировать возможности интеграции различных курсов с образовательных платформ в одном месте для удобства их использования.

Одним из вариантов реализации единой образовательной платформы служит способ создания агента [1], который по структуре платформы собирает необходимую информацию с сайтов, таких как: *coursera*, *stepik* и т.д. Выходной файл агента - *json* файл со структурой курса внутри. Агент реализован на языке *Python*. Проблемы, возникшие при реализации: на платформе курсы доступны только зарегистрированным пользователям, ресурсы не позволяют повторный парсинг сайта, некоторые платформы имеют уже разбитые на темы курсы (блоки курсов), не все курсы собираются агентом.

Следующим этапом работы стал выбор и апробация архитектуры нейронной сети для построения индивидуальной траектории обучающегося. В настоящий момент реализуется этап классификации курсов по темам. Входными данными для нейронной сети при анализе текста является вектор – текст в виде чисел (применяется метод векторизации). Метод токенизации – удаление функциональных слов (семантически нейтральных слов, таких как союзы, предлоги, артикли и пр.). Далее осуществляется морфологический анализ (производятся разметка по частям речи и стемматизация). Это позволяет значительно сократить размерность пространства. Методы извлечения признаков из текста (*N* граммы (последовательности слов длиной от 1 до *N*), мешок слов (*bag of words*, множество всех слов). В нейронных сетях плотное векторное представление слов (каждому токену сопоставляется вектор, размерность вектора ниже, чем у *one hot encoding*) определяется в процессе обучения [2]. На первом этапе элементы векторов инициализируются случайными числами, а изменение значений векторов происходит с помощью метода обратного распространения ошибки. Как итог, подготовка данных к подаче на вход нейронной сети является самым долгим и трудозатратным процессом. Для апробации методов искусственного анализа данных с помощью нейронной сети выбраны рекуррентные нейронные сети (сети с циклами). Для решения имеющих проблем у *RNN* (обучение требует длительного времени, проблема исчезающего градиента, ограниченная «длительность» запоминания предыдущей информации) выбраны более совершенные архитектуры рекуррентных сетей *LSTM* и *GRU* и одномерные сверточные нейронные сети [3, 4].

Результатом проделанной работы является агент, собирающий курсы с популярных образовательных платформ, и нейронная сеть, классифицирующая набор входных данных из *json* файла по темам курсов. Также планируется применение нейронной сети для предоставления пользователю, зарегистрировавшемуся на единой образовательной цифровой платформе, блоков курсов по шаблонам. В настоящий момент реализована структура агента для сайта *coursera*, 3 архитектуры нейронной сети (рекуррентные сети *LSTM* и *GRU* и одномерная сверточная нейронная сеть) с апробацией на размеченном новостном датасете с использованием библиотек *tensorflow.keras*, *pandas*, *numpy*, *matplotlib* и др. В дальнейшем возможно изучение веб-фреймворка *Django* (*Python*) для визуализации результатов работы агента и нейронной сети. А также доработка нейронной сети для анализа стека компетенций с целью последующего создания индивидуальной траектории обучения.

Список литературы

1. *IBM Cloud Application Performance Management. Конфигурирование агента Python agent, 2019.*
2. *Шолле Ф. Глубокое обучение на Python, 2018, 400 с.*
3. *Николенко С., Кадурин А., Архангельская Е.. Глубокое обучение. Погружение в мир нейронных сетей, 2018, 480 с.*
4. *Черняк Е. Технологии // Глубинное обучение в обработке и анализе текстов, 2019.*

АНАЛИЗ ВОЗМОЖНОСТЕЙ ИСПОЛЬЗОВАНИЯ ТЕХНОЛОГИИ РАСПОЗНАВАНИЯ ОБРАЗОВ ДЛЯ РЕШЕНИЯ ЗАДАЧИ АВТОМАТИЗАЦИИ УЧЕТА ПОСЕЩАЕМОСТИ

Ильина Анна Владимировна[1], **Пелеванюк Игорь Станиславович**[2]

[1] *Студент;*
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 22;
Направление обучения по основной образовательной программе:
Системный анализ и управление, группа 6015;
e-mail: anna.ilina.1307@yandex.ru.

[2] *Инженер-программист;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Ассистент;
Кафедра распределенных информационно-вычислительных систем;
Государственный университет «Дубна».

Ключевые слова: распознавание изображений, искусственный интеллект, автоматизация учета посещаемости.

Задача подсчета количества людей актуальна при проведении разного рода мероприятий, к которым могут относиться семинары, лекции, конференции, собрания, концерты и пр. Взамен монотонного ручного подсчета участников гораздо эффективнее использовать некоторое оборудование с необходимым программным обеспечением, которое позволяло бы детектировать изображения лиц участников и выдавать общее количество всех участников.

Целью проекта является исследование возможностей использования технологии распознавания лиц на изображениях или видеопотоке для решения задачи автоматизации учета посещаемости. Работа носит преимущественно исследовательский характер, но не исключена возможность использования результирующей системы в практических целях. Ожидаемым результатом является быстро внедряемый программно-аппаратный комплекс, позволяющий решить поставленную задачу и провести оценку возможности использования такого решения в практических целях. Для решения задачи использовалось следующее программное и аппаратное обеспечение: готовая нейронная сеть с необходимым *API face_recognition* [1], язык программирования *Python v*3.6, набор расширений графического фреймворка *Qt* для языка *Python PyQt*4, одноплатный микрокомпьютер *Raspberry Pi* 3 *Model B*+, ноутбук *ASUS X*751*L*. В результате разработана система детектирования и распознавания как изображений, уже имеющихся в базе, так и неизвестных системе. При этом неизвестное лицо записывается в хранилище системы как *Unknown*_Номер и изображение сохраняется для правильного переименования.

Был осуществлен выбор использования нейронной сети *face_recognition* ввиду ее простого и достаточного для решения задачи *API*, предложен графический интерфейс разрабатываемой системы, а также разработана сама система, произведены тестирования работы на двух выбранных наборах квадратных изображений: *Face Research Lab London Set* [2] и *Labeled Faces in the Wild* [3], а также представлены анализы этих результатов: произведены замеры времени работы системы при распознавании лиц в видеопотоке (при этом использовались: встроенная веб-камера ноутбука, камера от *Raspberry Pi*, веб-камера *CBR CW* 555*M*), произведены замеры времени работы алгоритма на данных наборах изображений, имеющих измененный мной размер от 1350 *px* до 50 *px* (на 10 фотографиях в размерах 1350, 1000, 800, 500, 400, 300, 200, 100, 90, 80, 70, 60, 50 *px*; на всем количестве фотографий для каждого набора в размере 200, 100, 90 и 80 *px*). Было подсчитано среднее время: кодирования одного изображения, поиска лица на изображении, сравнения найденного на изображении лица со всеми имеющимися в хранилище; общее время работы программы при: кодировании изображений, распознавании изображений. Произведены замеры количества закодированных и распознанных изображений для каждого из размеров. Построены соответствующие диаграммы, отображающие динамику изменения времени от размера изображений.

Тестирование производилось на ноутбуке и микрокомпьютере отдельно, в результате чего обнаружилась заметная разница во времени работы системы (на микрокомпьютере система работает примерно в два раза медленнее). Как показали результаты тестирования, оптимальный размер изображений составляет 100 * 100 *px*, что определяет наименьшее количество ошибок *I* и *II* родов при наименьшем времени работы системы (~16 мин на *Raspberry Pi* и ~6,17 мин на ноутбуке на наборе из 1020 фотографий). Из сказанного вытекает, что в работе будущей системы оптимально использовать изображения размером 100 * 100 *px*.

Данная работа имеет продолжение, в результате которого планируется разбить систему на серверную и клиентскую часть, а также получить ответы на дополнительный ряд вопросов, связанный с ускорением работы системы на других процессорах (в том числе с использованием технологии ускорения вычислений на нейронных сетях *Intel Movidius*), оптимизацией алгоритма, а также с возможностями внедрения данной системы на предприятиях.

Список литературы

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

ИССЛЕДОВАНИЕ ПРИМЕНИМОСТИ КОРРЕЛЯЦИОННЫХ ФИЛЬТРОВ К ЗАДАЧЕ ПОИСКА СХОЖИХ ИЗОБРАЖЕНИЙ В БАЗЕ

Кисеева Виктория Ильинична[1], Стрельцова Оксана Ивановна[2], Стадник Алексей Викторович[3]

[1] Студент;
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 22;
Направление обучения по основной образовательной программе:
Прикладная математика и информатика, группа 6181;
e-mail: vika.kiseeva@yandex.ru.

[2] к.ф.-м.н., старший научный сотрудник;
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Доцент;
Кафедра распределенных информационно-вычислительных систем;
Государственный университет «Дубна».

[3] к.ф.-м.н., инженер программист;
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Доцент;
Кафедра высшей математики;
Государственный университет «Дубна».

Ключевые слова: корреляционный фильтр, компьютерное зрение, работа с изображениями.

Существует ряд задач, в которых необходима реализация алгоритмов нахождения схожих изображений в базе: систематизация, составление каталогов, рекомендательные сервисы, поиск похожих людей в толпе, трекинг. В общем случае поиск схожих изображений подразумевает поиск изображений визуально похожих с точки зрения человека.

Целью проекта было положено построение простого и эффективного дескриптора для оценки визуальной близости изображений с использованием корреляционного фильтра, а также осуществление сравнительного анализа с работой алгоритма перцептивного хэша [1].

Использование корреляционного фильтра подразумевает применение преобразования Фурье и теоремы о свертке [2][3]. Алгоритм реализуется в среде *Visual Studio* на языке программирования *C++*.

Создание алгоритма с соответствующими характеристиками означает реализацию оптимального метода, точнее простейших методов (таких как перцептивный хэш), с более широким диапазоном применимости чем у «тяжелых» методов (таких как сиамские нейросети).

Оценка эффективности алгоритмов осуществляется за счет сравнительного анализа на открытых данных и данных, собранных специально для задачи, а именно: база изображений *Holiday dataset, Fashion MNIST,* а также раскадровка видеофрагментов.

На данном этапе был реализован алгоритм перцептивного хэша. Для этого была использована библиотека компьютерного зрения *OpenCV* [4]. Алгоритм был реализован на языке программирования *Phyton.* На рис. 1. представлены этапы работы алгоритма построения перцептивного хэша:

- уменьшение изображения без учета пропорций сторон;
- перевод изображения в градации серого.

Построение хэша с использованием информации о среднем показателе яркости для всего изображения: каждому пикселю ставится в соответствие 0 или 1, в зависимости от того он больше или меньше среднего показателя [1].

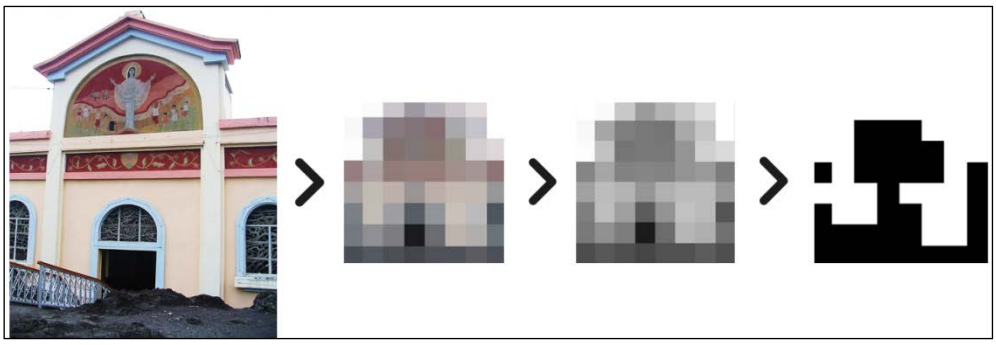


Рис. 1. Результат работы алгоритма построения перцептивного хэша

В дальнейшем планируется закончить реализацию алгоритма, использующего корреляционный фильтр и провести сравнительный анализ работы алгоритмов.

Список литературы

1. *Looks Like It. – [Электронный ресурс]. URL: https://www.hackerfactor.com/blog/?/archives/432-Looks-Like-It.html.*
2. *Садовников П. Оптические трекеры: ASEF и MOSSE. – [Электронный ресурс]. 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. – [Электронный ресурс]. URL: https://docs.opencv.org/.*

ПРОГРАММНЫЕ КОМПОНЕНТЫ ДЛЯ СЕРВИСА АНАЛИЗА МРТ ИЗОБРАЖЕНИЙ

Костоправов Антон Александрович[1], Стрельцова Оксана Ивановна[2]

[1] Студент;
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 21;
Направление обучения по основной образовательной программе:
Автоматизация технологических процессов и производств, группа 2231;
e-mail: akstud451@gmail.com.

[2] к.ф.-м.н., старший научный сотрудник;
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Доцент;
Кафедра распределенных информационно-вычислительных систем;
Государственный университет «Дубна».

Ключевые слова: классификация, МРТ, обработка изображений.

Целью проекта является создание программных компонентов для сервиса анализа МРТ изображений головного мозга. Заявленные программные компоненты должны анализировать цифровые томографические медицинские изображения для обнаружения патологий.

Программный продукт был реализован в сервисе разработки *JupiterNotebook*. Для реализации был выбран язык программирования высокого уровня *Python* версии 3.6.8 и открытая программная библиотека для машинного обучения *TensorFlow* версии 2.0 [1].

Первым шагом был поиск набора данных, а также поиск и изучение уже созданных решений по его классификации. Для выполнения поставленной задачи на сайте *Kaggle* был выбран размеченный набор данных "*Brain MRI ImagesforBrainTumorDetection*", состоящий из 253 изображений: 88 без опухолей и 145 с опухолями [2]. Для обучения нейронной сети изображения были разбиты на тренировочную выборку, состоящую из 202 изображений и проверочную из 51 изображения. В результате изучения готовых решений было установлено, что наибольшая точность классификации достигалась при использовании предобученных нейронных сетей, представленных в *Keras* [3]. Наивысшая точность классификации в 93% на проверочном наборе данных была достигнута при использовании предобученной на наборе данных *ImageNet* нейронной сети *VGG19*.

Следующим шагом стало улучшение эффективности обучения нейронной сети. Для этого была создана функция обратного вызова, замедляющая скорость обучения, если доля правильных ответов на проверочном наборе данных не изменяется в течении трех эпох обучения (см. рис. 1). Так же для повышения точности классификации были заморожены первые пять слоев предобученной нейронной сети. Результатом этого стало достижение 98% точности классификации на проверочном наборе данных.

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

Рис. 1. Функция обратного вызова

В дальнейшем обученная нейронная сеть может быть использована в сервисе анализа изображений головного мозга или дообучена для классификации других томографических снимков.

Список литературы

1. *TensorFlow // Открытая программная библиотека для машинного обучения. – 2020. – [Электронный ресурс]. URL: https://www.tensorflow.org/.*
2. *Kaggle // Набор данных с МРТ изображениями головного мозга, для выявления опухолей. – 2020. – [Электронный ресурс]. URL: https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumor-detection.*
3. *Keras // Модуль приложений библиотеки глубокого обучения Keras. – 2020. – [Электронный ресурс]. URL: https://keras.io/applications/.*

ПОСТРОЕНИЕ ТЕПЛОВОЙ КАРТЫ ДВИЖЕНИЯ ОБЪЕКТОВ

Полонский Денис Дмитриевич[1], **Стрельцова Оксана Ивановна**[2]

[1] *Студент;*
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 21;
Направление обучения по основной образовательной программе:
Программная инженерия, группа 4252;
e-mail: denuha@mail.ru.

[2] *к.ф.-м.н., старший научный сотрудник;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Доцент;
Кафедра распределенных информационно-вычислительных систем;
Государственный университет «Дубна».

Ключевые слова: определение движения, тепловая карта, *OpenCV*, видеофайл.

В данном проекте реализовано построение тепловой карты движения объектов. Тепловая карта представляет собой сумму времени, в течении которого объект находился в движении в определенной точке в поле зрении камеры. Для отображения частоты движения используется шкала из трех цветов (зеленый, желтый, красный), которые прозрачным слоем накладываются на изображение [1].

Данное решение позволяет определять популярность различных мест (стоек, витрин) в магазине; выявлять предпочтительные маршруты движения людей или транспортных средств на территории; анализировать посещение различных объектов [1].

На вход подается видеофайл, записанный на стационарную камеру. Он разбивается на кадры (см. рис. 1). К каждому кадру применяется сжатие (уменьшение размера изображения) и размытие Гаусса для уменьшения шумов.

Для обнаружения движения объектов в кадре, находится разница между текущим и предыдущим кадром. Обнаруженное движение вырезается из фона (см. рис. 2) [2, 3, 4].

Рис. 1. Кадр из видеофайла

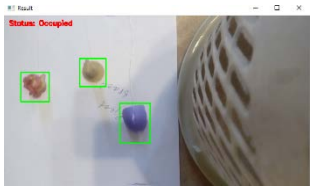


Рис. 2. Определение движения

Каждое обнаруженное движение в каждом кадре сохраняется и в результате складывается. В конечном итоге выводится изображение: фон, который представляет собой первый кадр и наложенное на него цветовое обозначение интенсивности движения. Зеленый цвет означает низкую интенсивность движения, желтый – среднюю, красный – высокую (см. рис. 3).

Для работы с изображениями используется библиотека *OpenCV*.

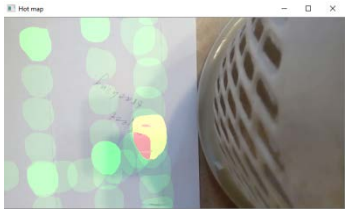


Рис. 3. Результат

Перспективой развития является реализация дружественного графического пользовательского интерфейса с возможностью управления сжатием изображения, размытием по Гауссу, количеством пропущенных кадров для повышения производительности и поддержки работы на различных типах архитектур.

Список литературы

1. *Тепловая карта интенсивности движения. – [Электронный ресурс]. URL: https://macroscop.com/assets/documentation/macroscop-2-6/win-client/analytics/hotmap.htm.*
2. *WebCam Motion Detector in Python – GeeksforGeeks. – [Электронный ресурс]. URL: https://www.geeksforgeeks.org/webcam-motion-detector-python/.*
3. *Высокопроизводительный детектор движения с подавлением шума на Python, OpenCV и Numba. – [Электронный ресурс]. URL: https://bitworks.software/high-speed-movement-detector-opencv-numba-numpy-python.html.*
4. *Basic motion detection and tracking with Python and OpenCV – PyImageSearch. – [Электронный ресурс]. URL: https://www.pyimagesearch.com/2015/05/25/basic-motion-detection-and-tracking-with-python-and-opencv/.*

ПОИСК ОТРАЖАЮЩИХ ПОВЕРХНОСТЕЙ НА ИЗОБРАЖЕНИЯХ

Постолов Илья Станиславович[1], Стрельцова Оксана Ивановна[2], Стадник Алексей Викторович[3]

[1] *Студент;*
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 22;
Направление обучения по основной образовательной программе:
Информационные системы и технологии, группа 4281;
e-mail: innpot@uni-dubna.ru.

[2] *к.ф.-м.н., старший научный сотрудник;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Доцент;
Кафедра распределенных информационно-вычислительных систем;
Государственный университет «Дубна».

[3] *к.ф.-м.н., инженер программист;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Доцент;
Кафедра высшей математики;
Государственный университет «Дубна».

Ключевые слова: глубокое обучение, машинное обучение, машинное зрение, изображение, зеркальное отражение.

Зеркала и зеркальные поверхности довольно часто встречаются в нашей повседневной жизни. Существующие системы компьютерного зрения не учитывают зеркальные отражения при обработке изображений и, следовательно, могут неверно интерпретировать данные из-за отраженного содержимого внутри зеркала. Однако определить, есть на изображении зеркало или другая отражающая поверхность, крайне трудно. Основная проблема заключается в том, что зеркальные поверхности, как правило, отражают содержимое, аналогичное их окружению, что затрудняет их поиск [1].

В этом проекте исследуются возможности нейросетевого подхода для решения задачи обнаружения зеркальных поверхностей на изображениях. В рамках данного проекта решаются следующие задачи: формирование обучающей выборки (см. рис. 1), исследование применимости нейросетевого подхода путем апробации различных архитектур нейросетей: *U-net, CNN*, сиамские нейросети, *MirrorNet* для решения поставленной задачи [2]. На данные момент в проекте реализованы *U-net* и *CNN* архитектуры для решения поставленной задачи. В дальнейшем в проекте планируется реализация сиамской нейросети, применение архитектуры *MirrorNet* и обучение на тестовом наборе данных, а также анализ и сравнение полученных результатов от применения реализованных архитектур нейросетей.

Рис. 1. Тестовый набор данных MirrorNet

Список литературы

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. *Шолле Ф. Глубокое обучение на Python, 2018, 400 с.*

ИСПОЛЬЗОВАНИЕ МЕТОДОВ ГЛУБОКОЙ ДОМЕННОЙ АДАПТАЦИИ В ЗАДАЧЕ РАСПОЗНАВАНИЯ БОЛЕЗНЕЙ РАСТЕНИЙ

Резвая Екатерина Петровна[1], Ососков Геннадий Алексеевич[2], Гончаров Павел Владимирович[3]

[1] Студент;
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 22;
Направление обучения по основной образовательной программе:
Информатика и вычислительная техника, группа 4013;
e-mail: rezvaya2016@gmail.com.

[2] д.ф.-м.н., главный научный сотрудник;
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Профессор;
Кафедра системного анализа и управления;
Государственный университет «Дубна».

[3] Аспирант;
Кафедра системного анализа и управления;
Государственный университет «Дубна».

Ключевые слова: классификация болезней растений, искусственные нейронные сети, методы доменной адаптации.

Потеря урожая из-за болезней растений является серьезной проблемой для сельского хозяйства и экономики. Возможность определить тип заболевания поможет предотвратить распространение инфекции. В ЛИТ ОИЯИ была разработана платформа для определения болезней растений (*PDDP*) [1, 2].

В *PDDP* для решения проблемы распознавания болезней растений по фотографиям их листьев успешно используются методы глубокого обучения [2]. Но такие методы требуют большой обучающей выборки. Сбор и маркировка подходящего набора данных – очень сложная и дорогостоящая процедура. Иногда может быть собрана только относительно небольшая база размеченных изображений. В то же время, существует ряд методов, используемых для решения задач классификации в случае малой обучающей выборки [3]. К таким методам относятся *transfer learning* [4], обучение сиамской нейронной сети [4], *zero-shot learning* [4], доменная адаптация [5].

В сети в открытом доступе есть большая база изображений *Plant Village*, состоящая из 50000 размеченных снимков листьев растений. Но все фотографии сделаны на белом фоне со строгим позиционированием листа в центре, что помешает получить хороший результат в случае загрузки изображения листа в реальных условиях (смещенный относительно центра лист, яркий фон, посторонние предметы в кадре). В отличие от *Plant Village dataset, PDD* содержит реальные изображения из Интернета. Именно наличие двух схожих наборов данных повлияло на выбор методов доменной адаптации для обучения. Суть этих методов в том, что используется сразу два набора данных – домен-источник (*Plant Village dataset*) и целевой (*PDD*). Уровень схожести между целевым и исходным наборами данных определяет насколько успешным будет обучение [5]. В работе были применены два метода. Оба метода подразумевают обучение сверточной нейронной сети. Перед обучением была проведена нормализация данных, аугментация (были добавлены зеркально отраженные дубликаты, повернутые в случайном направлении изображения и экземпляры с искаженной цветовой палитрой).

Первый из примененных методов – *Domain-Adversarial Training of Neural Networks* (*DANN*): нейронная сеть, в этом случае, состоит из двух функций потерь – классификатора и смешения доменов. Минимизируя эти функции, можно добиться того, что обе выборки будут неразличимы для классификатора, что позволит достигнуть высокой точности [5]. Для обучения была использована предобученная сеть *MobileNetV*2. Выбранный метод не позволил получить ожидаемый результат на наборе данных *PDD* [1]. Точность классификации осталась в пределах 60%.

Второй метод подразумевает двухэтапное обучение предобученной сети *ResNet34*. Сначала сеть обучается на домене-источнике с фиксированным шагом обучения, а затем, после предварительной замены классификатора и заморозки нескольких слоев, дообучается на целевом наборе данных. Причем, во время обучения есть несколько контрольных точек (каждые 100 эпох), в которых происходит уменьшение значения шага обучения на два порядка. Это позволяет повысить точность классификации, избегая переобучения. Глобальная цель этого метода – приблизить распределение весов к нужному на целевой базе и получить более высокое качество классификации [5]. Данный метод позволил достигнуть 90% точности классификации. Также планируется провести классификацию с использованием метода *Unsupervised Domain Adaptation with Deep Metric Learning* (*M-ADDA*) [5].

Список литературы

1. *Платформа для определения болезней растений // ЛИТ ОИЯИ. – 2019. – [Электронный ресурс]. 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. *Николенко С., Кадурин А., Архангельская Е. Глубокое обучение // Спб.: Питер, 2018. – 480 с.*
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. *Обзор основных методов Deep Domain Adaptation (Часть 1) // Блог компании Mail.ru Group. – 2018. – [Электронный ресурс]. URL: https://habr.com/ru/company/mailru/blog/426803/.*

МОДЕЛИРОВАНИЕ ТОНКИХ СТРУКТУР В РАСПРЕДЕЛЕНИЯХ ПРОДУКТОВ ЯДЕРНЫХ РЕАКЦИЙ ПО МАССЕ И ИХ РАСПОЗНАВАНИЕ МЕТОДАМИ МАШИННОГО ОБУЧЕНИЯ

Руденко Михаил Олегович[1], Ососков Геннадий Алексеевич[2], Пятков Юрий Васильевич[3]

[1] *Студент*
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 21;
Направление обучения по основной образовательной программе:
Программная инженерия, группа 4253;
e-mail: michadas@yandex.ru.

[2] *д.ф.-м.н., главный научный сотрудник;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Профессор;
Кафедра системного анализа и управления;
Государственный университет «Дубна».

[3] *д.ф.-м.н., профессор;*
Национальный исследовательский ядерный университет «МИФИ».
Ведущий научный сотрудник;
Лаборатория ядерных реакций им. Г.Н. Флерова;
Объединенный институт ядерных исследований.

Ключевые слова: распады тяжелых ядер, моделирование, глубокое обучение, нейроклассификатор.

Глобальная цель проекта – анализ проявлений кластеризации в редких многотельных распадах тяжелых ядер [1]. Были поставлены следующие задачи:

- разработать компьютерную модель тонкой структуры, найденную физиками из ЛЯР ОИЯИ, на основе экспериментов с трансурановым элементом калифорний ^{252}Cf (см. рис. 1);
- проверить гипотезу о том, что найденная структура объективно существует, а не является шумовым артефактом.

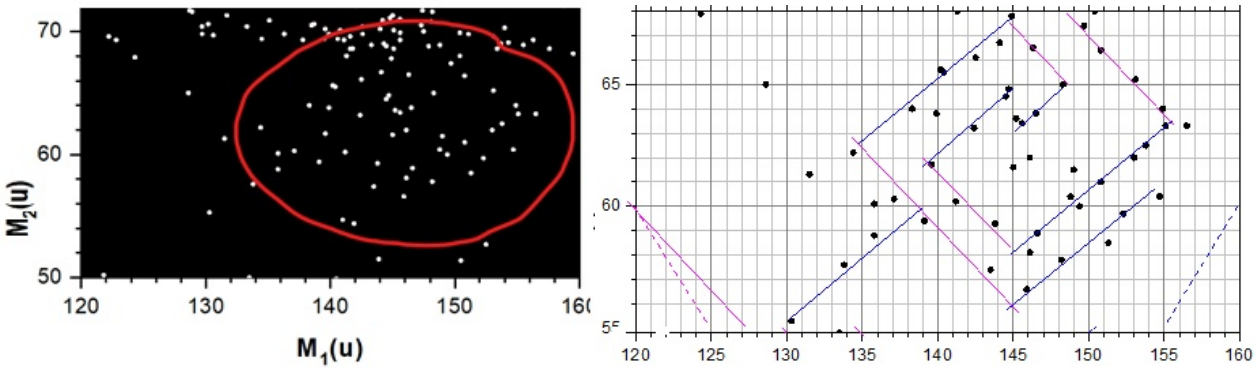


Рис. 1. Фрагмент корреляционно-массового распределения осколков деления ^{252}Cf из работы [1]. Специфическая ромбо-спиральная структура на левом рисунке отмечена овалом. Справа та же структура в более крупном масштабе с выделением подогнанных линий, образующих ромбический меандр

С помощью метода поворотных гистограмм [2] на экспериментальном изображении (рис.1 слева) распознаны 10 отрезков прямых линий, образующих ромбический меандр, и на основе статистического анализа определены их вероятностные параметры. Это позволило создать числовую модель меандра и разработать программу-генератор как самой тонкой структуры, так и ее альтернативной стохастической модели в виде такого же числа случайных точек, равномерно распределенных по полю меандра. Была создана глубокая сверточная нейронная сеть в качестве бинарного классификатора, обученного на большой выборке из модельных и шумовых изображений, полученных программой-генератором. В процессе решения использовался язык программирования *Python* с подключенными библиотеками: *matplotlib, keras, tensorflow, scikit-learn, numpy, pandas* [3-8]. Для подтверждения гипотезы о неслучайном происхождении структуры в виде ромбического меандра в работе был проведен численный эксперимент, в результате которого с помощью глубокого нейроклассификатора была получена вероятность обнаружения ромбического меандра на массиве из 10^5 статистически независимых наборов случайных точек, оказавшаяся пренебрежимо малой (0.00017). Вероятность наличия ромбо-спиральной структуры на оригинальном изображении (рис. 1) составила 99.913955%.

Список литературы

1. *Yu. V. Pyatkov, et al., Eur. Phys. J. A 48, 94 (2012)*
2. *Никитин В.А., Ососков Г.А., Автоматизация измерений и обработки данных физического эксперимента (монография) // Изд. МГУ, Москва, 1986, 185 стр.*
3. *Matplotlib: Visualization with Python. – [Электронный ресурс]. URL: https://matplotlib.org.*
4. *Keras. Simple. Flexible. Powerfull. – [Электронный ресурс]. URL: https://keras.io.*
5. *Tensorflow. An end-to-end open source machine learning platform. – [Электронный ресурс]. URL: https://www.tensorflow.org.*
6. *Scikit-learn. Machine Learning in Python. – [Электронный ресурс]. URL: https://scikit-learn.org.*
7. *NumPy. Fundamental package for scientific computing with Python. – [Электронный ресурс]. URL: https://numpy.org.*
8. *Pandas. Fast, powerful, flexible and easy to use open source data analysis and manipulation tool. – [Электронный ресурс]. URL: https://pandas.pydata.org.*

ПРИМЕНЕНИЕ СЕТИ U-NET В ЗАДАЧАХ СЕГМЕНТАЦИИ ИЗОБРАЖЕНИЙ

Рябов Андрей Русланович[1], Стрельцова Оксана Ивановна[2]

[1] *Студент;*
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 22;
Направление обучения по основной образовательной программе:
Прикладная информатика, группа 4071;
e-mail: rarjobs@mail.ru.

[2] *к.ф.-м.н., старший научный сотрудник;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Доцент;
Кафедра распределенных информационно-вычислительных систем;
Государственный университет «Дубна».

Ключевые слова: *U-Net*, архитектура, сверточные сети, задача сегментации.

Целью проекта является рассмотрение архитектуры сети *U-Net* и применение ее в задачах детектирования объектов. Для этого была определена задача, которая заключается в детектировании морских судов и дальнейшем выяснении эффективности применения *U-Net.*

U-Net была разработана для сегментации биомедицинских изображений. Эта сверточная нейросеть характеризуется высокой точностью предсказания и небольшим числом тренировочных данных. На нисходящем (сужающемся) пути последовательно выполняются операции свертки 3*x*3 с функцией активации *ReLu* и операции объединения (*MaxPool* 2*x*2 с шагом 2), после одной такой итерации каналы свойств удваиваются. На восходящем (расширяющемся) пути применяется свертка 2*x*2, которая уменьшает число каналов свойств, затем происходит объединение с соответствующим обрезанным изображением, затем свертка 3*x*3 с функцией активации ReLu (см. рис. 1).

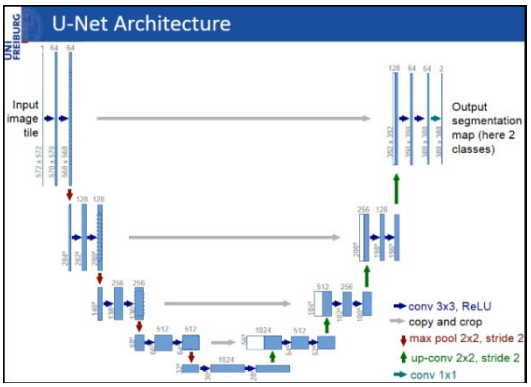


Рис. 1. Архитектура U-Net [1-3]

После изучения архитектуры была рассмотрена задача обнаружения судов на аэроснимках (см. рис. 2). Эта сеть натренированна на 1950 экземплярах, и 50 экземпляров выделены на валидацию. Таким образом на данных в размере 2000 изображений получается, что точность детектирования составляет 0,9888. Отсюда можно сделать вывод о том, что сеть качественно сегментирует изображение в условиях малой обучающей выборки и подходит для точного детектирования объектов.

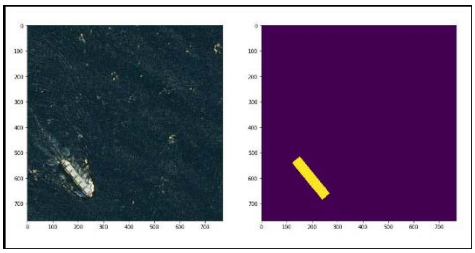


Рис. 2. Детектирование судов [4]

В будущем планируется рассмотреть нейросеть на задаче детектирования самолетов на аэроснимках аэродромов. Для этого предполагается найти соответствующие данные, разметить их и подать в сеть. Следующим шагом можно использовать данные автомобилей на аэроснимках.

Список литературы

1. *U-Net: Convolutional Networks for Biomedical Image Segmentation – №1. – [Электронный ресурс]. URL: https://lmb.informatik.uni-freiburg.de/people/ronneber/u-net.*
2. *U-Net: нейросеть для сегментации изображений. – [Электронный ресурс]. URL: https://neurohive.io/ru/vidy-nejrosetej/u-net-image-segmentation.*
3. *Сегментация изображений при помощи нейронной сети: U-Net. – [Электронный ресурс]. URL: http://robocraft.ru/blog/machinelearning/3671.html.*
4. *Keras Based UNet Model Construction Tutorial – №2. – [Электронный ресурс]. URL: https://www.kaggle.com/krishanudb/keras-based-unet-model-construction-tutorial/notebook.*

ВЫБОР МЕТОДОВ ГЛУБОКОГО ОБУЧЕНИЯ ДЛЯ РЕШЕНИЯ ЗАДАЧИ РАСПОЗНАВАНИЯ БОЛЕЗНЕЙ РАСТЕНИЙ В УСЛОВИЯХ МАЛОЙ ОБУЧАЮЩЕЙ ВЫБОРКИ

Сметанин Артем Алексеевич[1], **Ососков Геннадий Алексеевич**[2], **Гончаров Павел Владимирович**[3]

[1] *Студент;*
Государственный университет «Дубна»;
Международная школа по информационным технологиям «Аналитика больших данных», группа 21;
Направление обучения по основной образовательной программе: Информатика и вычислительная техника, группа 4013;
e-mail: webstermaster777@gmail.com.

[2] *д.ф.-м.н., главный научный сотрудник;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Профессор;
Кафедра системного анализа и управления;
Государственный университет «Дубна».

[3] *Аспирант;*
Кафедра системного анализа и управления;
Государственный университет «Дубна».

Ключевые слова: распознавание, искусственные нейронные сети, глубокое обучение, перенос обучения.

Потеря урожая из-за болезней растений является серьезной проблемой для сельских жителей, экономики и продовольственной безопасности, требующей принятия своевременных мер для выявления и предотвращения болезней.

В последнее время для решения задачи распознавания болезней растений по фотографиям их листьев стали с успехом применяться нейросетевые методы глубокого обучения. В настоящем исследовании выполнен анализ методов, используемых для обучения глубоких сверточных нейронных сетей в условиях малой обучающей выборки. В нашем исследовании мы применили методы глубокого обучения, которые хорошо зарекомендовали себя на практике [1].

Для данных *PDD* (см. рис. 1) (*http://pdd.jinr.ru*) мы используем методику обучения переноса и сиамский нейросетевой метод с трехчленной функцией ошибки [2]. Для базовой сети мы используем архитектуру *MobileNetV2* [3], который позволяет запускать сеть на мобильном устройстве. Мы используем *KNN* в качестве классификатора, потому что этот метод не требует переподготовки при добавлении нового класса в набор данных. Метод, которым мы пользуемся предложенная классификация достигает 99,5% точности [4].

Рис. 1. Примеры изображений из PDD dataset

В дальнейшем планируется добавление новых культур, а так же улучшение точности и скорости обучения уже имеющихся моделей.

Список литературы

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. *Наборы изображений PDDP. – [Электронный ресурс]: 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. *Репозиторий Plant Disease Detection. – [Электронный ресурс]: https://github.com/WEBSTERMASTER777/pdd.*

ЗАДАЧА ПО ЗАМЕРУ ВЕРОЯТНОСТНОГО МЕТОДА (LSH) И СКАЛЯРНОГО ПРОИЗВЕДЕНИЯ В АНАЛИЗЕ БОЛЬШОГО НАБОРА ДАННЫХ

Тюпин Денис Николаевич[1], **Папоян Владимир Владимирович**[2]

[1] *Студент;*
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 22;
Направление обучения по основной образовательной программе:
Информационные системы и технологии, группа 4281;
e-mail: super.denistjupin2013kraft@yandex.ru.

[2] *Инженер-программист;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Аспирант;
Кафедра системного анализа и управления;
Государственный университет «Дубна».

Ключевые слова: *LSH*, скалярное произведение, анализ больших данных.

Сопоставление записей представляет собой ключевой шаг во многих проблемах анализа больших данных, особенно при работе с информацией из разрозненных источников больших данных. Методы вероятностного связывания записей обеспечивают хорошую основу для поиска и интерпретации частичных совпадений записей. Однако вычисление расстояния, между строками для сравниваемых записей, требует объединения. Прямое использование вероятностного связывания записей требует обработки декартового произведения наборов записей. В результате используется этап «блокирования», когда пары записей-кандидатов группируются по категориальному полю, что значительно уменьшает количество записей необходимых для работы [1].

В этом проекте была рассмотрена задача по замене вероятностного метода (*Locality-sensitive hashing*) на скалярное произведение в сопоставлении в многомерных данных с помощью видеокарт на языке программирования *Python* [2]. Локальное хеширование — это метод понижения размерности многомерных данных. *LSH* в этом подходе отображает множество точек в высокоразмерном пространстве во множество бинов (набор объектов) в хеш-таблице. Этот метод обладает способностью воспринимать местоположение (хэш, зависящий от местоположения), благодаря чему способен помещать соседние точки в один и тот же бин [3].

Скалярное произведение выступает как метод для ускорения сопоставления записей многомерных данных. При увеличении объемов данных выстраивается метод, в котором выполняются операции над векторами, в результате чего получается скаляр (число) [4]. Это позволяет приблизить результат проводимого анализа к «похожим» шаблонам. В рамках этой модели каждая запись описывается вектором, в котором каждому терму (информационном поиске называют слова, из которых состоит текст), использующемуся в документе, ставится в соответствие его весовое значение, определяемое на основе статистической информации о его появлении. Как в отдельном документе, так и во всем документальном массиве [5].

Участие в научном исследовании является завершенным.

Список литературы

1. *Noga Alon, Joel H. Spencer 2nd Edition the Probabilistic Method. Wiley- Interscience Series in Discrete Mathematics and Optimization // Изд. Бином, 2007.*
2. *Карау Холден, Уоррен Рейчел. Эффективный Spark. Масштабирование и оптимизация // Изд. Питер, 2018.*
3. *Salton, G., Buckley, C., 1988. Term-weighting approaches in automatic text retrieval. Information Processing & Management 24, 513–523. – [Электронный ресурс]. 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. – [Электронный ресурс]. 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. – [Электронный ресурс]. URL: https://doi.org/10.1016/j.jbi.*

ВИЗУАЛИЗАЦИЯ ВРЕДОНОСНЫХ СЕТЕВЫХ АТАК

Зизганов Тимофей Андреевич[1], Потапов Денис Сергеевич[2],
Пелеванюк Игорь Станиславович[3]

[1] Студент;
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 21;
Направление обучения по основной образовательной программе:
Программная инженерия, группа 4251;
e-mail: timaraylog1998@gmail.com.

[2] Студент;
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 21;
Направление обучения по основной образовательной программе:
Конструирование и технология электронных средств, группа 4141;
e-mail: potapov-deniska@inbox.ru.

[3] Инженер-программист;
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Ассистент;
Кафедра распределенных информационно-вычислительных систем;
Государственный университет «Дубна».

Ключевые слова: сетевые атаки, визуализация, сервер, карта распределенный отказ в обслуживании, злоумышленники.

Атаки на *SSH* – распределенные атаки класса «отказ в обслуживании», атаки на вычислительные системы, имеющие целью сделать их недоступными для пользователей. Эти атаки заключаются в одновременной отправке в сторону определенного ресурса большого количества запросов с одного или многих компьютеров. Если десятки тысяч или миллионы компьютеров одновременно начнут посылать запросы в адрес определенного сервера, то либо не выдержит сервер, либо не хватит полосы пропускания канала связи к этому серверу. В обоих случаях, пользователи этой сети не смогут получить доступ к атакуемому серверу, или даже ко всем серверам и другим ресурсам, подключенным через заблокированный канал связи.

Цель совместного проекта – разработать инструмент, который позволяет визуализировать вредоносные сетевые атаки на конкретный сервер.

Были поставлены следующие задачи.

1. Анализ логов SSH демона, извлечение из них информации о том, под каким логином и с какого ip-адреса злоумышленники пытаются войти в систему.
2. Создание демона, который способен автоматически анализировать логи *SSH* демона и результаты записывать в базу данных *ip_address, geo_coordinates, login.*
3. Создание веб-страницы на которой отображается карта мира и несколько таблиц.

На карте точкой обозначается местоположение хоста. При старте страницы берется список атак за последние 24 часа. На карте расставляются все точки с которых велись атаки. В зависимости от количества атак линия между источником атаки и нашим хостом меняет цвет от желтого, до красного.

- На одной таблице отображается статистика по тому, какие логины чаще всего используют злоумышленники.
- На второй таблице отображается из каких стран в основном ведутся атаки.
- На третей таблице отображаются лидеры *ip* по источникам атак.

При помощи *Virtual Box* [1] была создана виртуальная машина на операционной системе “*Ubuntu Server* 19.10” [2]. Далее, были написаны скрипты на языке *bash* [3]. С помощью команды *lastb* (настроен на вывод журнальных записей, файла */var/log/btmp*, который содержит записи обо всех неудавшихся попытках регистрации пользователей в системе) получал *ip*-адреса и *login*, которые были за последние сутки. Дальше формировался *GET*-запрос на сторонний ресурс [4], для получения информации о данном *IP*-адресе (страна, город, широта, долгота). После получения этих данных, они вносились в базу данных *SQLite* [5], которые в дальнейшем использовал мой коллега Денис для визуализации (см. рис. 1). Скрипт *install.sh*, нужен для установки необходимого программного обеспечения на сервер, и добавления запуска скрипта с анализом журнала записей информации о злоумышленниках в *crontab* (используется для управления демоном *cron* – классический демон, использующийся для периодического выполнения заданий в определенное время) настроенном на выполнении раз в минуту, и создание сервиса *systemd* (подсистема инициализации и управления службами в *Linux*), который запускает веб-сервер *Flask* [6] в фоновом режиме. Скрипт *remove.sh,* удаляет сервис *systemd* и автоматический запуск из *cron.*

Создается сайт при помощи веб-фреймворка *Flask* [6] и отображается *html*-страница. Благодаря библиотеке *OpenLayers* [7] делается привязка карты к созданной странице. На этой карте отображаются вредоносные атаки на наш сервер (см. рис. 1). Список злоумышленников для визуализации атак берется из базы *SQLite* [5] (файл неудачных попыток авторизации *db.sqlite*). База данных находится на виртуальной машине *Virtual Box* [1], которой занимается мой коллега Тимофей. Из этой базы необходимо взять координаты злоумышленника (страна, город, широта, долгота). Для этого создаются запросы *select*. Связь базы данных с картой осуществляется на языке программирования *python* [8]. Далее наносятся координаты на карту по широте(*lat*) и долготе(*lon*), ставиться маркер, показывающий из какой точки мира идет атака на сервер и рисуется линия между сервером и злоумышленником. Для этих целей используется язык программирования *JavaScript* [9]. Так же, по запросу *./rest/getCountryStatistic* на *html*-странице отображаются три таблицы: Таблица по популярности *login*, Таблица по популярности стран, Таблица по популярности *IP* (см. рис. 2).

Проделанная работа не является окончательной. Готов рабочий прототип [10], проведены все подготовительные работы, связанные с изучением отдельных технологий, готов репозиторий в гит. Требуется тестирование, решение проблем, связанных с тем, что количество запросов к *API* ограничено, серьезная доработка дизайна *UI*.

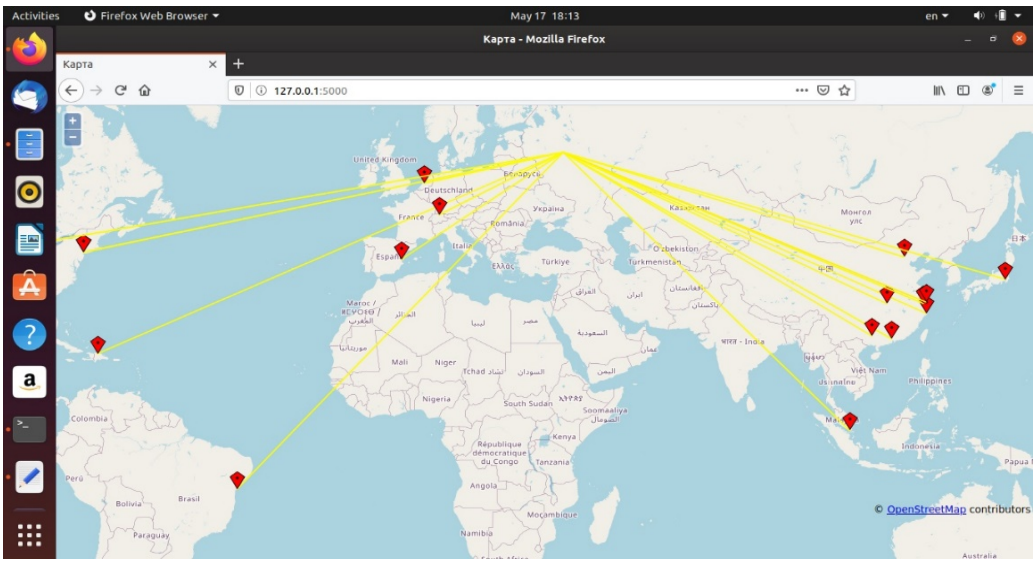
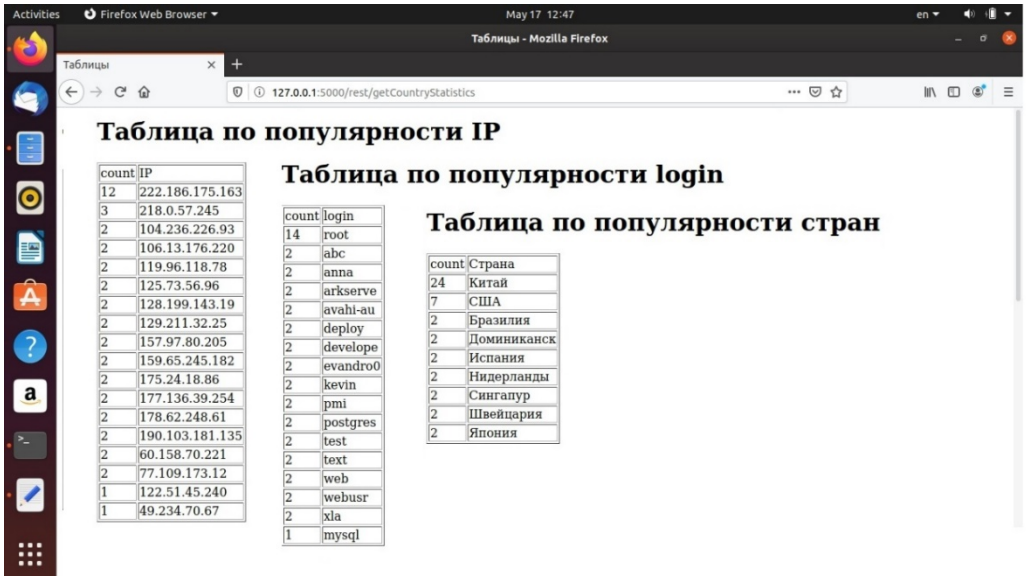


Рис. 1. Визуализация атак на сервер



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

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

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

Рис. 2. Таблицы со статистикой

Список литературы

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

COMPUTING & SOFTWARE ДЛЯ ЭКСПЕРИМЕНТОВ НА УСКОРИТЕЛЬНОМ КОМПЛЕКСЕ NICA

РАСШИРЕНИЕ ФУНКЦИОНАЛЬНОСТИ ПАКЕТА CERN ROOT ПО РАБОТЕ С ДАННЫМИ СУБД ORACLE ФОРМАТА «ДАТА-ВРЕМЯ»

Артемьев Алексей Владимирович[1], Герценбергер Константин Викторович[2]

[1] *Студент;*
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 21;
Направление обучения по основной образовательной программе:
Автоматизация технологических процессов и производств, группа 2231;
e-mail: fizz4ever1337@gmail.com.

[2] *к.т.н., Научно-экспериментальный отдел физики столкновений тяжелых ионов на комплексе NICA,*
Начальник группы математического и программного обеспечения;
Лаборатория физики высоких энергий;
Объединенный институт ядерных исследований.

Ключевые слова: *Oracle*, *TIMESTAMP*, *ROOT*.

Целью данного проекта является расширение возможностей последней версии пакета *CERN ROOT 6* по взаимодействию с данными СУБД *Oracle* временного формата *TIMESTAMP*. Текущий функционал фреймворка *ROOT 6* по работе с данным форматом обладает следующими ограничениями:

Размер буферной строки не позволяет хранить часовой пояс и микросекунды.

Отсутствует возможность возвращать из базы данных значения типа TTimeStamp, тем самым отсутствует функционал по получению и записи времени с точностью до долей секунд.

Запись и чтение данных формата времени с часовым поясом происходят некорректно.

Первым делом было проведено развертывание операционной системы *CentOS 8* с последующей установкой и настройкой на ней пакета *CERN ROOT* версии 6.20/04 и СУБД *Oracle* версии 19.3 [1].

Следом были изучены возможности пакета *ROOT 6* по работе с временными данными, хранимыми в СУБД *Oracle*, а также изучены возможности самой СУБД по хранению, чтению и записи данных формата *TIMESTAMP*, а также была развернута тестовая база данных для последующей проверки функционала пакета *CERN ROOT* на практике [2].

После изучения модулей пакета *ROOT 6*, таких как *TOracleStatement*, которые предоставляют инструментарий для работы с СУБД Oracle было выявлено, что буферная строка реализована корректно (п. 1), так как в нем изначально реализован динамический буфер строки [3].

На текущий момент осуществляется подключение к СУБД *Oracle* через инструментарий, предоставляемый модулем *TOracleServer* пакета *CERN ROOT* для последующей проверки описанных выше недоработок, а также для обнаружения новых [3].

В планах по окончанию работ по расширению возможностей данного пакета с СУБД *Oracle* перейти к корректировке и реализации данных возможностей в оставшихся СУБД, поддерживаемых *ROOT 6*, а именно: *MySQL* и *SQLite*.

Список литературы

1. *Centos 8 // Образ операционной системы и рекомендации по установке. – 2020. – [Электронный ресурс]. URL: https://www.centos.org/.*
2. *Документация СУБД Oracle // Инструкция по установке базы данных на Linux. – 2020. – [Электронный ресурс]. URL: https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/.*
3. *Исходный код и документация CERN ROOT // Исходный код ветки Master. – 2020. – [Электронный ресурс]. URL: https://root.cern.ch/doc/master/.*

АДАПТАЦИЯ СЕРВЕРНОЙ КОМПОНЕНТЫ СИСТЕМЫ УПРАВЛЕНИЯ НАГРУЗКОЙ (ЗАДАЧАМИ И ЗАДАНИЯМИ) PANDA ДЛЯ ИНТЕГРАЦИИ С СИСТЕМОЙ УПРАВЛЕНИЯ ПРОЦЕССОМ ОБРАБОТКИ ДАННЫХ ДЛЯ ЭКСПЕРИМЕНТА BM@N

Гаврилов Дмитрий Иванович[1], Петросян Артем Шмавонович[2], Олейник Данила Анатольевич[3]

[1] *Студент;*
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 21;
Направление обучения по основной образовательной программе:
Программная инженерия, группа 4254;
e-mail: dmitriygavrofficial@gmail.com.

[2] *Ведущий программист;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.

[3] *Ведущий программист;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.

Ключевые слова: *PanDA, PandaServer, workflow.*

Система обработки данных с эксперимента *BM@N* состоит из нескольких компонентов, которые взаимодействуют между собой [1]:

- система транспортировки данных;
- система контроля данных;
- система управления процессом обработки данных;
- система управления нагрузкой.

Система управления нагрузкой необходима для организации работы с вычислительными ресурсами разных типов [2].

Цели использования системы управления нагрузкой:

- унифицировать интерфейс доступа к различным вычислительным ресурсам;
- оптимизация загрузки ресурсов задачами;
- управление выполнением заданий с учетом приоритетов.

В качестве системы управления нагрузкой было выбрано программное обеспечение *PanDA* (*Production and Distributed Analysis System*) (рис. 1) [3].

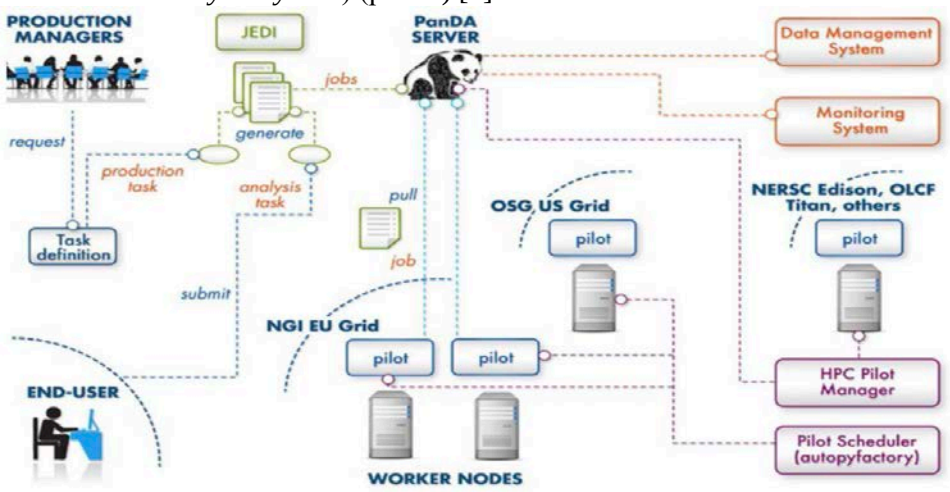


Рис. 1. PanDA

Был установлен и настроен *PandaServer* (ядро, где выполняются основные операции *PanDA*), создана база данных для *PandaServer*, установлены необходимые компоненты, такие как *httpd, httpd-devel, mod_ssl, gridsite*. Было протестировано добавление задач.

Список литературы

1. *Petrosyan A. Workflow Services for distributed processing BM@N data. – [Электронный ресурс]. 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. – [Электронный ресурс]. URL: https://indico.jinr.ru/event/738/contributions/6446/.*
3. *The PanDA Production and Distributed Analysis System. – [Электронный ресурс]. URL: https://twiki.cern.ch/twiki/bin/view/PanDA/PanDA.*

РАЗРАБОТКА СИСТЕМЫ МОНИТОРИНГА БАЗЫ ДАННЫХ ЭКСПЕРИМЕНТА BM@N ПРИ ПОМОЩИ ПАКЕТА GRAFANA

Кузьменков Игорь Викторович[1], **Герценбергер Константин Викторович**[2]

[1] *Студент;*
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 21;
Направление обучения по основной образовательной программе:
Информатика и вычислительная техника, группа 4012;
e-mail: ivt2018tms16@gmail.com.

[2] *к.т.н., Научно-экспериментальный отдел физики столкновений тяжелых ионов на комплексе NICA,*
Начальник группы математического и программного обеспечения;
Лаборатория физики высоких энергий;
Объединенный институт ядерных исследований.

Ключевые слова: мониторинг, метрики, визуализация, база данных, *PostgreSQL, Grafana, InfluxDB, Telegraf.*

Целью данной работы является мониторинг и визуализация метрик базы данных (БД) эксперимента *BM@N* проекта *NICA* и сервера, на котором она расположена. Рассматриваемая база данных хранит информацию о сеансах эксперимента, детекторах и их геометрии, конфигурации, калибровке и других параметрических данных, которые используются при офлайн обработке данных эксперимента. Система управления базой данных (СУБД) является *PostgreSQL* [1].

Необходимость создания системы веб-мониторинга обусловлена требованиями надежности, предъявляемыми к программному обеспечению эксперимента, с возможностью оперативно оценить параметры работы систем при помощи удобного пользовательского интерфейса, реализованного с применением пакета *Grafana* [2], а в случае аппаратных сбоев или программных ошибок автоматически оповестить членов коллаборации, ответственных за поддержку, при помощи средств, предоставляемых пакетом. Сохраняемый в специализированной базе данных временных рядов *InfluxDB* [3] журнал значений параметров системы, собираемых при помощи программного сервиса *Telegraf* [4], позволяет провести анализ состояния системы за все время наблюдения, выявить причину возникновения сбоя, что может использоваться для дальнейшего развития информационной системы эксперимента, построенной на рассматриваемой базе данных.

Сервис для сбора метрик *Telegraf* предоставляет возможность сбора данных как о сервере, на котором развернута база данных эксперимента, так и статистики по работе самой базы данных. Сбор метрик ограничивается теми показателями, которые представляются наиболее важными для оценки работы системы за счет конкретизации запросов на языке *SQL*. Выходом в терминологии сборщика данных является база данных временных рядов, в качестве которой выбрана система *InfluxDB*.

В ходе данной работы была развернута виртуальная машина с операционной системой *CentOS* 7, на которой были установлены программные решения *InfluxData*, такие как: *InfluxDB* и *Telegraf*, выполняющие необходимые функции по сбору метрик и хранению временных рядов для последующего использования в системе мониторинга. Также на машине был развернут сервис визуализации метрик *Grafana*. Выбранный комплекс решений показал свою работоспособность и относительную простоту использования.

В результате выполненной работы полученные метрики базы данных эксперимента и ее сервера визуализируются при помощи реализованного веб-сервиса на системе *Grafana*. Решение позволяет оценить загруженность процессора сервера БД, параметры работы устройства хранения, включая количество и скорость операций записи/чтения, оценить работу индексов. Также данная система мониторинга предоставляет историю изменений показателей работы базы данных эксперимента при помощи *SQL* запросов и визуализирует полученные в результате запросов значения метрик.

Таким образом, разрабатываемая система позволяет решить вопросы мониторирования и оценки интенсивности использования базы данных в любой момент времени с возможностью выбрать интервал сбора данных для детального изучения, а также предоставить возможность исследования функционирования базы данных, статистики по работе индексов, доступу, нагрузке на диск и, собственно, сам сервер БД. Реализованное решение позволяет проводить оптимизацию и отладку работы базы данных эксперимента и оперативно исправлять возникающие проблемы. Дальнейшая работа будет направлена на изучение необходимых метрик из предлагаемого набора сборщика для баз данных на *PostgreSQL*. Предстоит выбрать необходимые для мониторинга метрики и визуализировать их в унифицированном веб-сервисе *Grafana* ОИЯИ.

Список литературы

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

АДАПТАЦИЯ/РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ В РАМКАХ РАСПРЕДЕЛЕННОЙ СИСТЕМЫ ОБРАБОТКИ ДАННЫХ ЭКСПЕРИМЕНТА BM@N

Матвеев Иван Андреевич[1], **Олейник Данила Анатольевич**[2], **Петросян Артем Шмавонович**[3]

[1] *Студент;*
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 22;
Направление обучения по основной образовательной программе:
Прикладная информатика, группа 4071;
e-mail: matveev.a.ivan@yandex.ru.

[2] *Ведущий программист;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.

[3] *Ведущий программист;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.

Ключевые слова: *unified*, *resource*, *management*, *system*, *baryonic*, *matter*, *nuclotron*, *computing resource*, *information*, *catalogue*.

Унифицированная система управления ресурсами — это ИТ-экосистема, состоящая из набора подсистем и сервисов, которые должны унифицировать доступ к данным и вычислительным ресурсам в гетерогенной распределенной среде, автоматизировать большинство операций, связанных с массовой обработкой данных, избегать дублирования основных функций путем совместного использования систем различными пользователями, в результате — снизить эксплуатационные расходы, повысить эффективность использования ресурсов, обеспечить прозрачный учет использования ресурсов (см. рис. 1) [1].

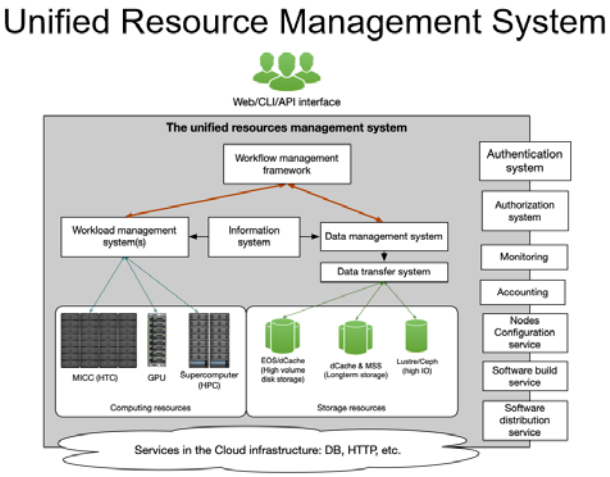


Рис. 1. Unified Resource Management System schema

Основной задачей проекта является развертка информационной системы в рамках унифицированной системы управления ресурсами для проведения экспериментов на установке *Nuclotron-based Ion Collider fAcility* (*NICA*) в которую входит эксперимент *Baryonic Matter at Nuclotron* (*BM@N*) [2, 3]. Для решения данной задачи была выбрана информационная система *Computing Resource Information Catalogue* (*CRIC*) [4] (см. рис. 2).

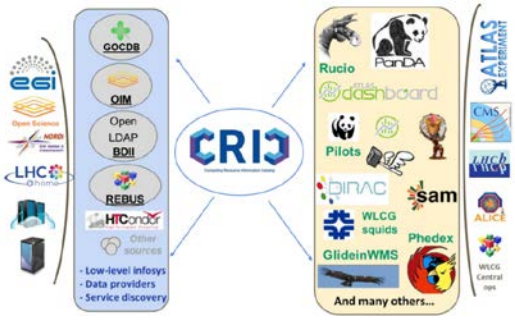


Рис. 2. Computing Resource Information Catalogue

Было проведено развертывание тестовой версии системы *CRIC* на ресурсах облачной инфраструктуры ЛИТ ОИЯИ. Работа над проектом будет продолжаться. Следующим этапом является определение форматов экспорта данных в другие системы.

Список литературы

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

ПРИМЕНЕНИЕ НЕЙРОСЕТЕВОГО ПОДХОДА ДЛЯ ЗАДАЧ РЕКОНСТРУКЦИИ ТРЕКА ЭКСПЕРИМЕНТА MPD ПРОЕКТА NICA

Махлов Егор Вячеславович[1], **Стрельцова Оксана Ивановна**[2]

[1] *Студент;*
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 21;
Направление обучения по основной образовательной программе:
Программная инженерия, группа 4251;
e-mail: fictioncentralacc@gmail.com.

[2] *к.ф.-м.н., старший научный сотрудник;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Доцент;
Кафедра распределенных информационно-вычислительных систем;
Государственный университет «Дубна».

Ключевые слова: нейронные сети, обучение нейросетей, логический вывод нейросетей, нейросетевые архитектуры, вычислительные архитектуры, реконструкция треков частиц, *NICA, MPD, HEP*.

Одной из важнейших проблем обработки данных экспериментов физики высоких энергий является процедура реконструкции треков частиц, которая заключается в построении треков (траекторий) частиц на основе множества пространственных попаданий, именуемых хитами, этих частиц в чувствительные слои детектора. Современные условия экспериментов физики высоких энергий диктуют необходимость поиска новых методов обработки данных, поскольку между классическими методами, применяемыми десятилетиями, и оптимальными может существовать существенный разрыв. Для решения данной проблемы создаются различные исследовательские группы [1].

Данная работа была проведена в рамках исследовательской группы ОИЯИ, занимающейся поиском, исследованием, и созданием методов обработки данных для эксперимента *MPD* проекта *NICA* в сотрудничестве с компанией РСК Технологии.

Целью этой работы является апробация вычислительных архитектур для задач распознавания треков частиц с помощью нейросетевого подхода. Первым шагом для достижения поставленной цели стало выявление и описание применяющихся для реконструкции треков нейросетевых подходов, разработанных другими исследовательскими группами. Рекуррентные архитектуры являются основными в задачах реконструкции, поскольку способны обрабатывать последовательности [1][2]. Имеются применения сверточных архитектур с некоторыми ограничениями [3]. Также разрабатывается решения на менее известных архитектурах, такие как графовые нейросетевые архитектуры, но тестирование таких решений на реальных данных не проводилось [1].

Вторым шагом, для непосредственной апробации вычислительных архитектур, стало использование моделей сверточной и рекуррентной архитектуры написанные с помощью нейросетевой библиотеки *Keras*. Исследование проводилось на вычислительных устройствах в единичном экземпляре: *Intel Xeon 7680*, *Intel Xeon Phi 6048* и *Nvidia Tesla K100*. Ниже приведены результаты обучения и вывода сетей.

Рис. 1. Слева — медиана мс на шаг для обучения LSTM архитектуры; справа — медиана мс на шаг для обучения CNN архитектуры. Важно отметить, что при обучении использовался параметр samples_per_epoch при котором число выборок, обрабатываемых для каждой эпохи равно batch_size умноженному на steps_per_epoch, поэтому увеличение batch_size увеличивает время вычисление, а не уменьшает его.

Для логического вывода *LSTM* сети наилучший результат был получен на *Xeon 7680* (139 мкс), следом идет *Tesla K100* (221 мкс). Худший результат у *Xeon Phi 6048* (1 мс). Для логического вывода *CNN* сети градация идентична: *Xeon 7680* (36 мкс), *Tesla K100* (63 мкс), *Xeon Phi 6048* (281 мкс).

Приведенные результаты не являются окончательными, поскольку исследование ускорения от использования различных вычислительных архитектур проводилось на имитации реальных нейросетевых моделей, не применяемых в экспериментах, и объемах данных, отдаленных от объемов современных экспериментов. В дальнейшем планируется исследование логического вывода и обучения нейросетевых моделей применяемых для обработки больших объемов данных конкретного эксперимента.

Список литературы

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.*

ЧИСЛЕННЫЙ АНАЛИЗ ПРОЦЕССА РАССЕЯНИЯ ЧАСТИЦ ПРИ КОНЕЧНЫХ ТЕМПЕРАТУРАХ ЯДЕРНОЙ МАТЕРИИ

Рогожина Елизавета Дмитриевна[1], Калиновский Юрий Леонидович[2], Голяткина Любовь Игоревна[3]

[1] Студент;
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 22;
Направление обучения по основной образовательной программе:
Информационные системы и технологии, группа 4281;
e-mail: liorinoff@mail.ru.

[2] д.ф.-м.н., ведущий научный сотрудник;
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.
Доцент;
Заведующий кафедрой высшей математики;
Государственный университет «Дубна».

[3] Студент;
Кафедра информационных технологий;
Государственный университет «Дубна».

Ключевые слова: ядерная физика, *NICA*, диаграммы Фейнмана, рассеяние частиц, пион, мезон.

Цель работы – выполнить расчеты поперечных сечений рассеяния частиц при конечной температуре и плотности для проекта *NICA*.

Основной задачей является создание аналитического кода в *Wolfram Mathematica*, преобразовании этого результата в код *C++* и размещении этого кода на *HybriLIT* для помощи в расчетах экспериментов проекта *NICA*.

Основные процессы рассеяния определяются диаграммами Фейнмана двух типов: боксами и диаграммами рождения промежуточного мезона или, как их еще называют, треугольниками [1].

Для расчета пион-пион рассеяния использовались диаграммы типа «Бокс» и «Треугольник» [1] (см. рис. 1).

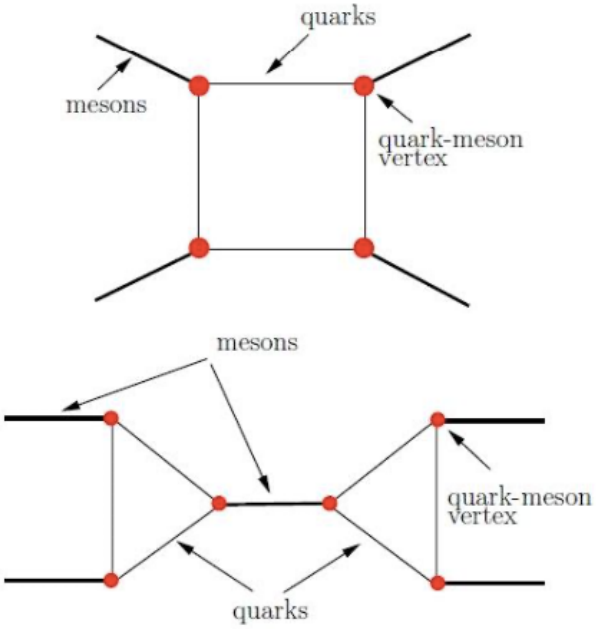


Рис. 1. Диаграммы типа «Бокс» и «Треугольник»

Для расчетов мы используем *Wolfram Mathematica* с *Package-X* и *LoopTools* для вычисления некоторых петлевых интегралов [2].

На данный момент были проведены аналитические расчеты амплитуд и сечений рассеяния мезонов и разработан *C++* код для помощи в расчетах экспериментов проекта *NICA*.

В дальнейшем планируется применить этот подход к глюодинамике и исследовать процессы $gg \to \pi\pi$.

Список литературы

1. *Калиновский Ю.Л., Тонеев В.Д., Фризен А.В. Фазовая диаграмма барионной материи в SU(2)-модели Намбу – Иона-Лазинио с петлей Полякова, УФН, 186 (4) 2016. Объединенный институт ядерных исследований.*
2. *Hiren H. Patel Package-X. – [Электронный ресурс]. URL: https://packagex.hepforge.org.*

РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ ПРОЦЕССОМ ОБРАБОТКИ ДАННЫХ НА ЭКСПЕРИМЕНТЕ BM@N

Ячменёв Андрей Алексеевич[1], Петросян Артем Шмавонович[2], Олейник Данила Анатольевич[3]

[1] *Студент;*
Государственный университет «Дубна»;
Международная школа по информационным технологиям
«Аналитика больших данных», группа 21;
Направление обучения по основной образовательной программе:
Программная инженерия, группа 4254;
e-mail: andrew91.99@yandex.ru

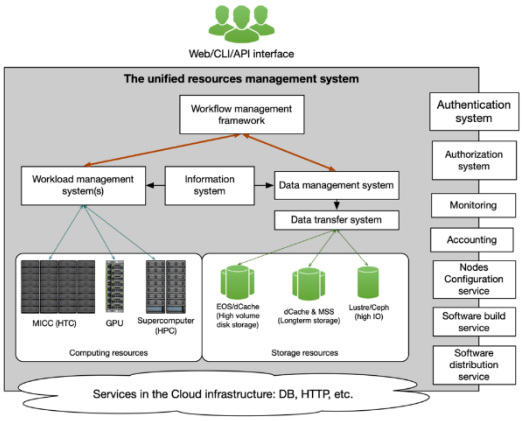
[2] *Ведущий программист;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.

[3] *Ведущий программист;*
Лаборатория информационных технологий;
Объединенный институт ядерных исследований.

Ключевые слова: *airflow*, *luigi*, *workflow management*, *BM@N*.

Единая система управления ресурсами [1] — это ИТ-экосистема, состоящая из совокупности подсистем и сервисов (см. рис. 1), которые должны:

- унифицировать доступ к данным и вычислительным ресурсам в гетерогенной распределенной среде;
- автоматизировать большинство операций, связанных с массивной обработкой данных;
- не допускать дублирования основных функциональных возможностей путем совместного использования систем между различными пользователями (если это возможно);
- как следствие — снижение эксплуатационных затрат, повышение эффективности использования ресурсов;
- прозрачный учет использования ресурсов.



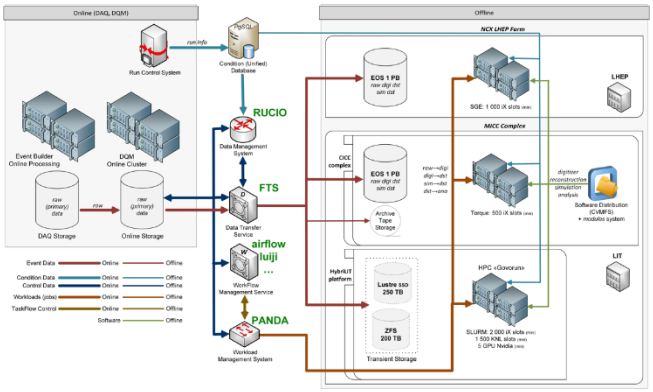


Рис. 1. Unified Resource Management System *Рис. 2. Automation of BM@N reconstruction workflow*

Система обработки данных с эксперимента *BM@N* состоит из нескольких компонентов [1], которые взаимодействуют между собой (см. рис. 2).

Цель моего проекта подобрать и настроить систему управления процессом обработки данных (*workflow management framework*).

Система управления процессом обработки данных необходима для описания цепочек задач удобным способом, запускать последовательности задач, следить за зависимостями и контролировать ход выполнения задач.

Решено использовать средство с открытым исходным кодом, которое активно поддерживается сообществом, чтобы минимизировать усилия на разработку и поддержку системы.

Для этих целей было подобрано два средства: *luigi* [2] и *airflow* [3]. Это средства для построения, управления, контроля выполнения последовательностей задач.

Сравнение этих средств показало, что для целей построения системы управления процессом обработки данных лучше подходит *airflow*, так как является расширяемым и более гибким решением.

На текущем этапе построения системы настроена среда для создания системы, развернут *Airflow*. Ведется работа над созданием расширения для взаимодействия с системой управления нагрузкой *PanDA*. После настройки системы необходимо описать последовательности задач на языке программирования *Python*.

Список литературы

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

Научное издание

Сборник отчетов о научно-проектной деятельности выпускников Международной школы по информационным технологиям «Аналитика больших данных»

Сборник трудов
Выпуск 1

Под редакцией
Владимир Васильевич Кореньков, Евгения Наумовна Черемисина,
Оксана Ивановна Стрельцова, Дарья Игоревна Пряхина

В авторской редакции

Подписано в печать 31.05.2020.
Формат 60x90/8. Усл. печ. л. 6,5.
Тираж 100 экз. Заказ № 2.

Отпечатано в ООО «Диверпринт».
117335, г. Москва, ул. Архитектора Власова, д. 21.