



ISBN 978-1-64655-135-4

SCIENTIFIC ELECTRONIC
LIBRARY
LIBRARY.RU

Google
scholar

[HTTPS://SCIENTIFIC-CONFERENCE.COM](https://scientific-conference.com)



**LIBRARY OF
CONGRESS (USA)**

XXVI INTERNATIONAL CORRESPONDENCE SCIENTIFIC SPECIALIZED CONFERENCE

**INTERNATIONAL SCIENTIFIC REVIEW
OF THE TECHNICAL SCIENCES,
MATHEMATICS AND COMPUTER SCIENCES**

Boston. USA. December 22-23, 2022

ISBN 978-1-64655-135-4

UDC 08

**XXVI INTERNATIONAL CORRESPONDENCE
SCIENTIFIC SPECIALIZED CONFERENCE
«INTERNATIONAL SCIENTIFIC REVIEW OF
THE TECHNICAL SCIENCES, MATHEMATICS
AND COMPUTER SCIENCE»
(Boston. USA. December 22-23, 2022)**

BOSTON. MASSACHUSETTS
PRINTED IN THE UNITED STATES OF AMERICA
2022

INTERNATIONAL SCIENTIFIC REVIEW OF THE PROBLEMS OF THE TECHNICAL SCIENCES, MATHEMATICS AND COMPUTER SCIENCE / COLLECTION OF SCIENTIFIC ARTICLES. XXVI INTERNATIONAL CORRESPONDENCE SCIENTIFIC SPECIALIZED CONFERENCE (Boston, USA, December 22-23, 2022). Boston. 2022

EDITOR: EMMA MORGAN
TECHNICAL EDITOR: ELIJAH MOORE
COVER DESIGN BY DANIEL WILSON

CHAIRMAN OF THE ORGANIZING COMMITTEE: *VALTSEV SERGEI*
CONFERENCE ORGANIZING COMMITTEE:

Abdullaev K. (PhD in Economics, Azerbaijan), *Alieva V.* (PhD in Philosophy, Republic of Uzbekistan), *Akbulaev N.* (D.Sc. in Economics, Azerbaijan), *Alikulov S.* (D.Sc. in Engineering, Republic of Uzbekistan), *Anan'eva E.* (D.Sc. in Philosophy, Ukraine), *Asaturova A.* (PhD in Medicine, Russian Federation), *Askarhodzhaev N.* (PhD in Biological Sc., Republic of Uzbekistan), *Bajtasov R.* (PhD in Agricultural Sc., Belarus), *Bakiko I.* (PhD in Physical Education and Sport, Ukraine), *Bahor T.* (PhD in Philology, Russian Federation), *Baulina M.* (PhD in Pedagogic Sc., Russian Federation), *Blejh N.* (D.Sc. in Historical Sc., PhD in Pedagogic Sc., Russian Federation), *Bobrova N.A.* (Doctor of Laws, Russian Federation), *Bogomolov A.* (PhD in Engineering, Russian Federation), *Borodaj V.* (Doctor of Social Sciences, Russian Federation), *Volkov A.* (D.Sc. in Economics, Russian Federation), *Gavrilenkova I.* (PhD in Pedagogic Sc., Russian Federation), *Garagonich V.* (D.Sc. in Historical Sc., Ukraine), *Glushhenko A.* (D.Sc. in Physical and Mathematical Sciences, Russian Federation), *Grinchenko V.* (PhD in Engineering, Russian Federation), *Gubareva T.* (PhD in Laws, Russian Federation), *Gutnikova A.* (PhD in Philology, Ukraine), *Datij A.* (Doctor of Medicine, Russian Federation), *Demchuk N.* (PhD in Economics, Ukraine), *Divnenko O.* (PhD in Pedagogic Sc., Russian Federation), *Dmitrieva O.A.* (D.Sc. in Philology, Russian Federation), *Dolenko G.* (D.Sc. in Chemistry, Russian Federation), *Esenova K.* (D.Sc. in Philology, Kazakhstan), *Zhamuldinov V.* (PhD in Laws, Kazakhstan), *Zholdoshev S.* (Doctor of Medicine, Republic of Kyrgyzstan), *Zelenkov M.YU.* (D.Sc. in Political Sc., PhD in Military Sc., Russian Federation), *Ibadov R.* (D.Sc. in Physical and Mathematical Sciences, Republic of Uzbekistan), *Il'inskih N.* (D.Sc. Biological, Russian Federation), *Kajrakbaev A.* (PhD in Physical and Mathematical Sciences, Kazakhstan), *Kaftaeva M.* (D.Sc. in Engineering, Russian Federation), *Klinkov G.T.* (PhD in Pedagogic Sc., Bulgaria), *Koblanov Zh.* (PhD in Philology, Kazakhstan), *Kovaljov M.* (PhD in Economics, Belarus), *Kravcova T.* (PhD in Psychology, Kazakhstan), *Kuz'min S.* (D.Sc. in Geography, Russian Federation), *Kulikova E.* (D.Sc. in Philology, Russian Federation), *Kurmanbaeva M.* (D.Sc. Biological, Kazakhstan), *Kurpajanidi K.* (PhD in Economics, Republic of Uzbekistan), *Linkova-Daniels N.* (PhD in Pedagogic Sc., Australia), *Lukienko L.* (D.Sc. in Engineering, Russian Federation), *Makarov A.* (D.Sc. in Philology, Russian Federation), *Macarenko T.* (PhD in Pedagogic Sc., Russian Federation), *Meimanov B.* (D.Sc. in Economics, Republic of Kyrgyzstan), *Muradov Sh.* (D.Sc. in Engineering, Republic of Uzbekistan), *Musaev F.* (D.Sc. in Philosophy, Republic of Uzbekistan), *Nabiev A.* (D.Sc. in Geoinformatics, Azerbaijan), *Nazarov R.* (PhD in Philosophy, Republic of Uzbekistan), *Naumov V.* (D.Sc. in Engineering, Russian Federation), *Ovchinnikov Ju.* (PhD in Engineering, Russian Federation), *Petrov V.* (D.Arts, Russian Federation), *Radkevich M.* (D.Sc. in Engineering, Republic of Uzbekistan), *Rakhimbekov S.* (D.Sc. in Engineering, Kazakhstan), *Rozyhodzhaeva G.* (Doctor of Medicine, Republic of Uzbekistan), *Romanenkova Yu.* (D.Arts, Ukraine), *Rubcova M.* (Doctor of Social Sciences, Russian Federation), *Rumyantsev D.* (D.Sc. in Biological Sc., Russian Federation), *Samkov A.* (D.Sc. in Engineering, Russian Federation), *San'kov P.* (PhD in Engineering, Ukraine), *Selitrenikova T.* (D.Sc. in Pedagogic Sc., Russian Federation), *Sibircev V.* (D.Sc. in Economics, Russian Federation), *Skipko T.* (D.Sc. in Economics, Ukraine), *Sopov A.* (D.Sc. in Historical Sc., Russian Federation), *Strekalov V.* (D.Sc. in Physical and Mathematical Sciences, Russian Federation), *Stukalenko N.M.* (D.Sc. in Pedagogic Sc., Kazakhstan), *Subachev Ju.* (PhD in Engineering, Russian Federation), *Sulejmanov S.* (PhD in Medicine, Republic of Uzbekistan), *Tregub I.* (D.Sc. in Economics, PhD in Engineering, Russian Federation), *Uporov I.* (PhD in Laws, D.Sc. in Historical Sc., Russian Federation), *Fedos'kina L.* (PhD in Economics, Russian Federation), *Khiltukhina E.* (D.Sc. in Philosophy, Russian Federation), *Cuculjan S.* (PhD in Economics, Republic of Armenia), *Chiladze G.* (Doctor of Laws, Georgia), *Shamshina I.* (PhD in Pedagogic Sc., Russian Federation), *Sharipov M.* (PhD in Engineering, Republic of Uzbekistan), *Shevko D.* (PhD in Engineering, Russian Federation).

PROBLEMS OF SCIENCE
PUBLISHED WITH THE ASSISTANCE OF NON-PROFIT ORGANIZATION
«INSTITUTE OF NATIONAL IDEOLOGY»
VENUE OF THE CONFERENCE:
1 AVENUE DE LAFAYETTE, BOSTON, MA 02111, UNITED STATES
TEL. OF THE ORGANIZER OF THE CONFERENCE: +1 617 463 9319 (USA, BOSTON)
THE CONFERENCE WEBSITE:
[HTTPS://SCIENTIFIC-CONFERENCE.COM](https://scientific-conference.com)

PUBLISHED BY ARRANGEMENT WITH THE AUTHORS
Attribution-ShareAlike 4.0 International (CC BY-SA 4.0)
<https://creativecommons.org/licenses/by-sa/4.0/deed.en>

Contents

TECHNICAL SCIENCES.....	4
<i>Kasimova S.R. (Republic of Azerbaijan) METHODS OF MEASURING THE DIELECTRIC COEFFICIENTS OF HIGHLY ABSORBING SOLID AND LIQUID SUBSTANCES / Касимова С.Р. (Азербайджанская Республика) МЕТОДЫ ИЗМЕРЕНИЯ ДИЭЛЕКТРИЧЕСКИХ КОЭФФИЦИЕНТОВ СИЛЬНО.....</i>	4
<i>Schukina A.V., Belozеров O.I. (Russian Federation) ONLINE EDUCATIONAL SERVICES USING GAMIFICATION AS THE MAIN TOOL OF THE EDUCATIONAL PROCESS / Щукина А.В., Белозеров О.И. (Российская Федерация) ОБРАЗОВАТЕЛЬНЫЕ ОНЛАЙН-СЕРВИСЫ, ИСПОЛЬЗУЮЩИЕ ГЕЙМИФИКАЦИЮ, КАК ОСНОВНОЙ ИНСТРУМЕНТ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА</i>	8
<i>Bozhenko Z.M., Chernobrivets V.O., Kostin I.V. (Russian Federation) INVESTIGATION OF CHANGES IN THE STRESS-STRAIN STATE OF A BERTHING STRUCTURE MADE OF CAISSON BERTH DURING THE CONVERSION OF THE BERTH / Боженко З.М., Чернобривец В.О., Костин И.В. (Российская Федерация) ИССЛЕДОВАНИЕ ИЗМЕНЕНИЯ НАПРЯЖЕННО-ДЕФОРМИРОВАННОГО СОСТОЯНИЯ ПРИЧАЛЬНОГО СООРУЖЕНИЯ ИЗ МАССИВОВ-ГИГАНТОВ ПРИ ПЕРЕПРОФИЛИРОВАНИИ ПРИЧАЛА</i>	13
<i>Sinelnik K.E., Limanova N.I. (Russian Federation) CREATE A PROGRAM THAT CAN RECOGNIZE DRAWINGS DRAWN BY THE USER USING A LEARNING NEURAL NETWORK IN THE OBJECT-ORIENTED JAVA LANGUAGE / Синельник К.Э., Лиманова Н.И. (Российская Федерация) СОЗДАТЬ ПРОГРАММУ, КОТОРАЯ СМОЖЕТ РАСПОЗНАВАТЬ РИСУНКИ, НАРИСОВАННЫЕ ПОЛЬЗОВАТЕЛЕМ С ПОМОЩЬЮ ОБУЧАЮЩЕЙСЯ НЕЙРОННОЙ СЕТИ НА ОБЪЕКТНО-ОРИЕНТИРОВАННОМ ЯЗЫКЕ JAVA.....</i>	21
<i>Karnaukhov A.V. (Russian Federation) HOW ALPHAGO WORKS - THE FIRST PROGRAM TO BEAT A HUMAN IN THE GAME OF GO / Карнаухов А.В. (Российская Федерация) КАК РАБОТАЕТ АЛЬФАГО - ПЕРВАЯ ПРОГРАММА, ПОБЕЖДАЮЩАЯ ЧЕЛОВЕКА В ИГРЕ ГО.....</i>	44

TECHNICAL SCIENCES

METHODS OF MEASURING THE DIELECTRIC COEFFICIENTS OF HIGHLY ABSORBING SOLID AND LIQUID SUBSTANCES

Kasimova S.R. (Republic of Azerbaijan)

*Kasimova Sevda Rasim – PhD in physico-mathematics Sciences,
Associate Professor,
ENGINEERING PHYSICS AND ELECTRONICS DEPARTMENT,
AZERBAIJAN TECHNICAL UNIVERSITY,
BAKU, REPUBLIC OF AZERBAIJAN*

Abstract: *the dependences of the wave reflection coefficient on the layer thickness of a liquid absorbing dielectric and a magnet in free space and in a TE waveguide are studied. The functional relationships between the position and the magnitude of the extreme of these dependences and the values of the dielectric or magnetic properties of the substance are determined.*

Keywords: *dielectric loss, dielectric constant, liquid and solid substances.*

МЕТОДЫ ИЗМЕРЕНИЯ ДИЭЛЕКТРИЧЕСКИХ КОЭФФИЦИЕНТОВ СИЛЬНО ПОГЛОЩАЮЩИХ ТВЕРДЫХ И ЖИДКИХ ВЕЩЕСТВ

Касимова С.Р. (Азербайджанская Республика)

*Касимова Севда Расим гызы – кандидат физико-математических наук, доцент,
кафедра инженерной физики и электроники,
Азербайджанский технический университет,
г. Баку, Азербайджанская Республика*

Аннотация: *исследованы зависимости коэффициента отражения волны от толщины слоя жидкого поглощающего диэлектрика и магнетика в свободном пространстве и в ТЭ волноводе. Определены функциональные зависимости между положением и величиной экстремума этих зависимостей и величинами диэлектрических или магнитных свойств вещества.*

Ключевые слова: диэлектрические потери, диэлектрическая проницаемость, жидкие и твердые вещества.

UDC 621:3.035.222.7.:621.317.335.3

A number of microwave methods for measuring the dielectric properties of liquid and solid substances with $\operatorname{tg} \delta < 0.8$, have been developed with the use of a resistance transformer in the measurement circuit. Their practical application allows obtaining reliable information on the dielectric and magnetic properties of liquid objects of research, and on their basis about their molecular structure. In particular, the analysis of the behavior of the frequency and temperature dependences of the permittivity and dielectric loss of polar liquids and their solutions allows us to determine the dipole moments of polar molecules, polarizability, relaxation times, and energy activations of dipole relaxation, the nature of the orientations of the dipoles and a number of other important molecular characteristics of the substance [1]. It is very promising to apply it, to assess the influence of the nearest environment on the relaxation processes of dipole molecules when analyzing the dielectric properties of concentrated solutions, the components of which have different molecular nature [2]. They allow to determine the possibility of the formation of molecular associates and complexes due to the action of inter- and intramolecular hydrogen bonds.

In this regard, one of the relatively simple methods for determining the wave impedance and the associated values of the dielectric coefficients of a substance is the inserting of a thickness-adjustable, flat layer of a non-absorbing liquid located on the surface of the study into the measurement circuit blown substance [3]. Measurement scheme ε' and ε'' using a waveguide water-guiding system, in which as an analogue of a variable resistance transformer a flat quarter-wave non-absorbing dielectric that is adjustable in position is used the plate is shown in Fig. 1.

As matching liquid, it is permissible to use non-polar liquids that do not have absorption, for example, benzol, hexane, etc. [4]. To measure the standing wave coefficient, two directional couplers 10 are used, the outputs of the detector sections 11 of which are connected to the standard voltage standing wave meter 12.

The consideration of Fig. 2 implies the possibility at definite choices of the dielectric properties of the coating material to achieve one of the minima of the function $\rho(x)$ of zero value. It is characteristic that, with a decrease in wave attenuation in matter, the interface between the normal and anomalous regions shifts toward higher x values. The latter indicates that the conditions for the complete absorption of incident radiation in the dielectric layer are satisfied. It is associated with the appearance of zero minima of the function $\rho(x)$.

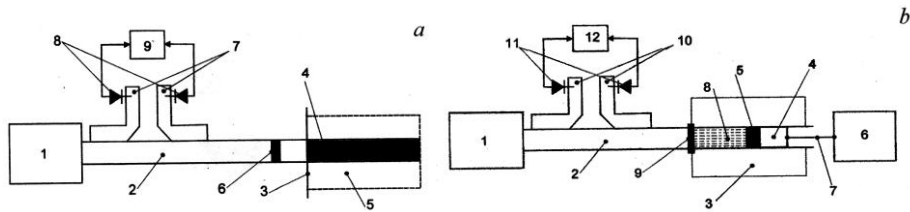


Fig. 1. Block diagrams of measuring the dielectric coefficients of highly absorbing solid and liquid substances using a panoramic standing wave coefficient meter and using a quarter-wavelength plate (a) and matching liquid (b). 1 - klystron generator with an attenuator and a wave meter; 2 - a directing path; 3 - directional coupler; 4 - detector; 5 - coefficient of stand wave's meter; 6 - plate; 7 - measuring cell; 8 - thermostat; 9 - measured substance; 10 - matching liquid; 11 - piston; 12 - micrometric device

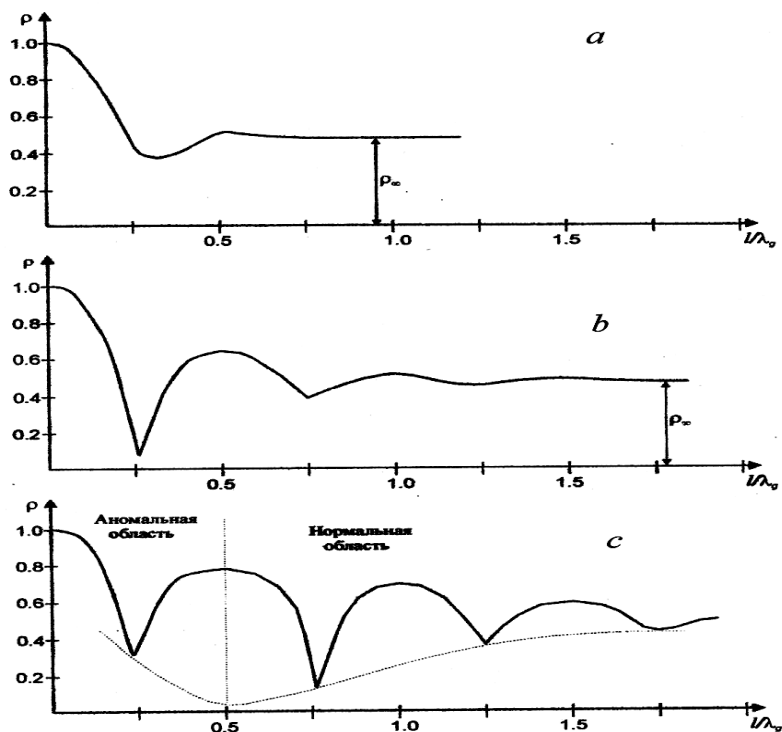


Fig. 2. The dependences of the modulus of the reflection coefficient of the wave ρ on the thickness of the layer l of the substance with values of its loss factor γ equal to 0.6 (a), 0.3 (b), and 0.1 (c). λ_d is the wavelength in the substance

References / Список литературы

1. Poley J.Ph. The Computation of the Complex Dielectric Constant from Micro-wave Impedance Measurements. // Appl.Sci.Res., 1955. V. B4. № 5. P. 337.
2. Kasimov R.M., Kasimova S.R. Nonreflective passage of electromagnetic radiation on its incidence at an angle on the absorbing layer of a dielectric // Journal of Engineering Physics and Thermophysics. New-York, USA, 2011. Vol. 84. Is.4. Pp. 787-793.
3. Kasimova S.R. Measurements of the Dielectric Properties of Strongly Absorbing Substances at Microwave Frequencies // Measurement Techniques. USA, New-York, 2016. Volume 58. Issue 12. Pp. 1372-1375.

4. *Kasimova S.R.* Application of the method of pulse sounding the substance for identifying and measuring the dielectric properties of polar liquids // *Paradigmata Poznání*. Prague, Czech Republic. № 3, 2017. P. 59-62.
-

ONLINE EDUCATIONAL SERVICES USING GAMIFICATION AS THE MAIN TOOL OF THE EDUCATIONAL PROCESS

Schukina A.V.¹, Belozerov O.I.² (Russian Federation)

¹*Schukina Alena Viktorovna – Student;*

²*Belozerov Oleg Ivanovich – PhD in Technics, Associate Professor,
DEPARTMENT OF COMPUTER ENGINEERING AND COMPUTER
GRAPHICS,*

*NATURAL SCIENCE INSTITUTE
FAR EASTERN STATE TRANSPORT UNIVERSITY,
Khabarovsk, Russian Federation*

Abstract: *the article deals with issues related to the introduction of gaming teaching methods in the educational process. The analysis of several successful educational resources using gamification has been carried out. Using the example of three online services, it is considered what specific tasks gamification solves, what game mechanics are used and how it improves learning. The choice of gamification as the main method for working with students allowed the described educational platforms not only to increase attendance, but also to increase the interest of participants in the learning process.*

Keywords: *learning, learning platform, gamification.*

ОБРАЗОВАТЕЛЬНЫЕ ОНЛАЙН-СЕРВИСЫ, ИСПОЛЬЗУЮЩИЕ ГЕЙМИФИКАЦИЮ КАК ОСНОВНОЙ ИНСТРУМЕНТ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

Щукина А.В.¹, Белозеров О.И.² (Российская Федерация)

¹Щукина Алена Викторовна – студент;

²Белозеров Олег Иванович – кандидат технических наук, доцент,
кафедра вычислительной техники и компьютерной графики,

Естественно-научный институт,

Дальневосточный государственный университет путей
связи,

г. Хабаровск, Российская Федерация

Аннотация: в статье рассматриваются вопросы, связанные с внедрением игровых методов обучения в учебный процесс. Проведен анализ нескольких успешных образовательных ресурсов, использующих геймификацию. На примере трёх онлайн-сервисов рассмотрено, какие конкретные специфические задачи решает геймификация, какие игровые механики используются и как это улучшает обучаемость. Выбор геймификации, в качестве основного метода для работы с учениками, позволил описанным образовательным платформам не просто поднять посещаемость, но и увеличить заинтересованность участников в процессе обучения.

Ключевые слова: обучение, учебная платформа, геймификация.

В три года каждый ребёнок начинает познавать мир через игру. Несмотря на то, что, взрослея, человек уже способен фокусироваться на задаче продолжительные отрезки времени, ставить себе цели и сроки выполнения работы, игра всё ещё остаётся самым простым методом познания. Указанный механизм, реализованный в виде игровой формы взаимодействия, успешно используется многими сервисами для привлечения и удержания внимания пользователей. На данный момент множество банков используют кэш-бек для привлечения клиентов [1]. Кэш-бек выступает, как механика «Достижений» - за определённые покупки клиент получает награду.

Не удивительно, что и онлайн-образование всё чаще прибегает к

геймификации. Такой подход способен решить разный круг задач, но чаще всего используется для решения следующих проблем:

отсутствие вовлеченности в образовательный процесс;

нехватка социального взаимодействия;

отсутствие понимания собственного прогресса.

Далее, на примере трёх онлайн-сервисов, будет рассмотрено, какие конкретные специфические задачи решает геймификация, какие игровые механики используются и как это улучшает обучаемость.

Учебная платформа Memrise [2], созданная Эдом Куком и Греггом Детре, в основном ориентирована на изучение иностранных языков, но здесь можно найти и другие темы для изучения. Платформа активно развивается за счёт пользовательского контента. Создавая свой набор карточек, пользователь может им поделиться с другими участниками платформы. Основные механики, которые использует сервис – это визуализация прогресса, достижения, квесты с заданиями, ограничение времени, персональный рейтинг и ежедневные задания. Основной упор сервис делает на яркую визуализацию контента. Аналогия получения знаний здесь завязана на механике взращивания собственного сада. Для каждого изучаемого слова используется своё растение, рост которого отображает процесс усвоения. Сервис очень популярен и имеет как версию для ПК, так и версию для мобильных устройств.

Другой сервис, в первую очередь ориентированный на подготовку школьников к ЕГЭ и ОГЭ, называется Castle Quiz [6]. В основе данного сервиса лежит механика дуэлей. Каждая дуэль приносит пользователю очки, которые он может потратить на улучшение своего замка. Пользователь может вступать в кланы, приглашать друзей и параллельно прокачивать свои знания по определённой теме. Каждый бой в игре проходит с живым оппонентом, что добавляет азарта при ответе на вопросы. Проблема заключается в том, что иногда очень трудно найти соперника и это может оттолкнуть. На данный момент в игру Castle Quiz уже сыграло более 3 миллионов человек и количество пользователей постоянно растёт.

Ещё один пример удачного кейса, использующего геймификацию, это обучающий квест «Путь волонтера» от сайта

Dobro.ru. Квест призван помочь волонтерам-новичкам освоиться в системе и найти свой уникальный путь волонтера. Для этого онлайн-проект использует следующие механики: визуализация прогресса, награды, велком-сценарий, квест с заданиями, возможность второй попытки, прокачка уровней, профиль пользователя и дополнительный контент, который можно разблокировать в процессе прохождения.

Сам сайт Dobro.ru позиционирует себя, как экосистема для волонтеров [12]. Его главной задачей является вовлечение большего количества пользователей в добровольческую активность. Квест как нельзя лучше подходит для того, чтобы новичок смог освоиться и найти свой путь волонтера. Каждый зарегистрированный пользователь получает возможность пройти один из путей: волонтер-культуры, волонтер-медицины или любое другое направление. Выбор зависит от индивидуальных предпочтений пользователя. Далее, участник квеста проходит 27 заданий, которые помогут ему развиваться на выбранном пути. Чем больше активности, тем больше шансов попасть, в последствии, в ту организацию, куда хочет сам волонтер.

В данной статье было рассмотрено три удачных кейса применения геймификации в образовательном процессе. Каждый из них имеет свои специфические особенности, характерные для данного образовательного направления. Выбор геймификации, в качестве основного метода для работы с учениками, позволил описанным образовательным платформам не просто поднять посещаемость, но и увеличить заинтересованность участников, в процессе обучения. Игрофикация образовательного процесса позволила сохранить сервисам аудиторию, а ученикам получить новый опыт и приобрести качественные знания.

Список литературы / References

1. Куликов В.П. Игровые элементы геймификационной системы / В.П. Куликов, С.Д. Байжуманов, К.Р. Кусманов // Вестник Алматинского университета энергетики и связи, 2019. № 4(47). С. 250-255. DOI 10.51775/1999-9801_2019_47_4_250. EDN QINYYN.

2. Разбор сервиса для изучения иностранных языков Memrise [Электронный ресурс]. Режим доступа: <https://www.gamification-now.ru/cases/memrise-obrazovatel'naya-platforma/> (дата обращения: 09.06.2022).
3. *Елисеева Е.В.* Технология геймификации в современном вузе / Е.В. Елисеева, С.Н. Злобина, В.В. Ерохин // Актуальные проблемы развития вертикальной интеграции системы образования, науки и бизнеса: экономические, правовые и социальные аспекты: материалы IV Международной научно-практической конференции, Воронеж, 22–23 октября 2015 года. Воронеж: Воронежский центр научно-технической информации, 2015. С. 190-194. –EDN VQRXBL.
4. *Демченко Т.С.* Геймификация в системе высшего образования: актуальность исследования / Т.С. Демченко // Новое поколение, 2016. № 9. С. 55-60. EDN VXNGWJ.
5. *Tsitavets T.Y.* Gamification in higher education: contemporary issues / T.Y. Tsitavets // Университет – территория опережающего развития: Сборник научных статей Международной научно-практической конференции, посвящённый 80-летию ГрГУ им. Янки Купалы, Гродно, 19–20 февраля 2020 года / Редколлегия: Ю.Я. Романовский (гл. ред.) [и др.]. Гродно: Гродненский государственный университет имени Янки Купалы, 2020. С. 256-257. EDN HSNILP.
6. Разбор сервиса Castle Quiz. [Электронный ресурс]. Режим доступа: <https://www.gamification-now.ru/cases/castle-quiz-intellektualnye-igry-dlya-podgotovki-k-ege-oge-i-samorazvitiya/> (дата обращения: 04.06.2022).
7. *Биджиева С.Х.* Геймификация образования: проблемы использования и перспективы развития / С.Х. Биджиева, Ф.А.А. Урусова // Мир науки. Педагогика и психология, 2020. Т. 8. № 4. С. 14. EDN FDZEAM.
8. *Сулеймен А.* Сравнительное исследование геймификации в образовании и обучение на основе игр / А. Сулеймен // Аллея науки, 2019. Т. 3. № 6(33). С. 857-863. EDN WGKLZF.

9. *Широколобова А.Г.* Геймификация в условиях цифровой трансформации образования / А.Г. Широколобова // Вестник Самарского государственного технического университета. Серия: Психолого-педагогические науки, 2022. Т. 19. № 1. С. 5-20. DOI 10.17673/vsgtu-pps.2022.1.1. EDN EHWPZE.
10. *Климкович Е.В.* Развитие геймификации образования в процессе реализации программ высшего и дополнительного образования / Е.В. Климкович // Современное педагогическое образование, 2021. № 8. С. 23-26. EDN LWUELE.
11. *Козлова Ю.Б.* Геймификация в системе современного высшего образования: теоретические основы и практическая значимость / Ю.Б. Козлова // История и педагогика естествознания, 2022. № 1. С. 19-22. DOI 10.24412/2226-2296-2022-1-19-22. EDN JQYNSY.
12. Разбор кейса онлайн-квеста «Путь волонтёра». [Электронный ресурс]. Режим доступа: <https://www.gamification-now.ru/cases/dobro-ru-put-volontyora-kulturi/> (дата обращения: 01.06.2022).
13. Разбор сервиса Lingualeo. [Электронный ресурс]. Режим доступа: <https://www.gamification-now.ru/cases/lingualeo/> (дата обращения: 09.06.2022).

INVESTIGATION OF CHANGES IN THE STRESS-STRAIN STATE OF A BERTHING STRUCTURE MADE OF CAISSON BERTH DURING THE CONVERSION OF THE BERTH

Bozhenko Z.M.¹, Chernobrivets V.O.², Kostin I.V.³
(Russian Federation)

¹*Bozhenko Zakhar Mikhailovich – Student;*

²*Chernobrivets Vladislav Olegovich – Student;*

³*Kostin Igor Vladimirovich – Associate Professor,*

DEPARTMENT OF WATERWAYS, PORTS AND HYDRAULIC STRUCTURES,

ACADEMY OF WATER TRANSPORT

RUSSIAN UNIVERSITY OF TRANSPORT (MIIT),

MOSCOW, RUSSIAN FEDERATION

Abstract: *the article analyzes changes in the stress-strain state of berthing structures made of caisson berth, with changes in the category of regulatory loads, other things being equal.*

Keywords: *caisson berth, stress-strain state.*

**ИССЛЕДОВАНИЕ ИЗМЕНЕНИЯ НАПРЯЖЕННО-
ДЕФОРМИРОВАННОГО СОСТОЯНИЯ ПРИЧАЛЬНОГО
СООРУЖЕНИЯ ИЗ МАССИВОВ-ГИГАНТОВ ПРИ
ПЕРЕПРОФИЛИРОВАНИИ ПРИЧАЛА**

**Боженко З.М.¹, Чернобривец В.О.², Костин И.В.³
(Российская Федерация)**

¹Боженко Захар Михайлович – студент,

²Чернобривец Владислав Олегович – студент,

³Костин Игорь Владимирович – доцент,

кафедра водных путей, портов и гидротехнических сооружений,

Академия водного транспорта

Российский университет транспорта (МИИТ),

г. Москва, Российская Федерация

Аннотация: *в статье анализируются изменения напряженно-деформированного состояния причальной конструкций из массивов-гигантов, при изменении категории нормативных нагрузок при прочих равных.*

Ключевые слова: *массив-гигант, напряженно-деформированное состояние.*

В настоящее время портовое хозяйство Российской Федерации насчитывает 184 порта. В нашей стране эксплуатируются 67 морских и 117 речных портов.

Значительная часть существующих портов была построена или реконструирована в 60-80-е годы прошлого столетия. Таким образом срок службы многих портовых сооружений достигает 50 и более лет. В связи с этим в настоящее время остро стоит проблема реконструкции и ремонта существующих сооружений, которые за длительный период их эксплуатации претерпели как физический, так и моральный износ. Особое место среди портовых гидротехнических сооружений безусловно занимают

причалы, играющие важнейшую роль в обеспечении грузо- и пассажирооборота российских портов. Так же стоит упомянуть, что в период с 2007 по 2020 годы мощность российских морских портов выросла в 2 раза и продолжает расти.

Учитывая факты, изложенные выше, можно утверждать, что России требуется провести большой объем работ по ремонту и реконструкции существующих портов, их сооружений, а также расширение портов и наращивание их мощностей. Одним из путей наращивания мощностей является перепрофилирование существующих причальных сооружений путем установки на них современного более мощного перегрузочного оборудования, переориентирования причалов для приема судов новых типов и т. д. При этом часто процесс перепрофилирования причала приводит к изменению категории действующих на него нормативных эксплуатационных нагрузок, которые определяются согласно требованиям СП 350.1326000.2018 «Нормы технологического проектирования морских портов» [8].

В связи с этим представляет интерес исследование изменения напряженно-деформированного состояния причального сооружения в случае его перепрофилирования, что и рассматривается в данной статье.

Целью работы является проведение численного исследования изменения характера внутренних усилий в элементах причального сооружения при замене действующих на него эксплуатационных нагрузок (со 2-й категории на нулевую). Т. е. исследуется возможность использования причала под действием увеличенных нагрузок без каких-либо дополнительных мероприятий по усилению сооружения.

В данной статье в качестве исследуемой конструкции причального сооружения рассматривается причал гравитационного типа их массивов-гигантов. Для проведения исследования используются следующие исходные данные:

- отметка дна у причала -9,75 м;
- категория нормативных нагрузок на причал – 2, измененная категория - 0;
- порталый кран Альбатрос, в количестве 2 штук;
- максимальная скорость ветра – 25 м/с;

- материал обратной засыпки - песок, грунт основания - галечник.

Отметим, что скорость ветра, а так же грунты основания, обратная засыпка и размеры массива-гиганта назначались условно, а значит, в зависимости от изменения геолокации причала, полученные результаты будут отличаться.

Составленная по исходным данным расчетная схема представлена на рис. 1.

Результаты расчетов представлены в таблицах №1, 2, 3, 4.

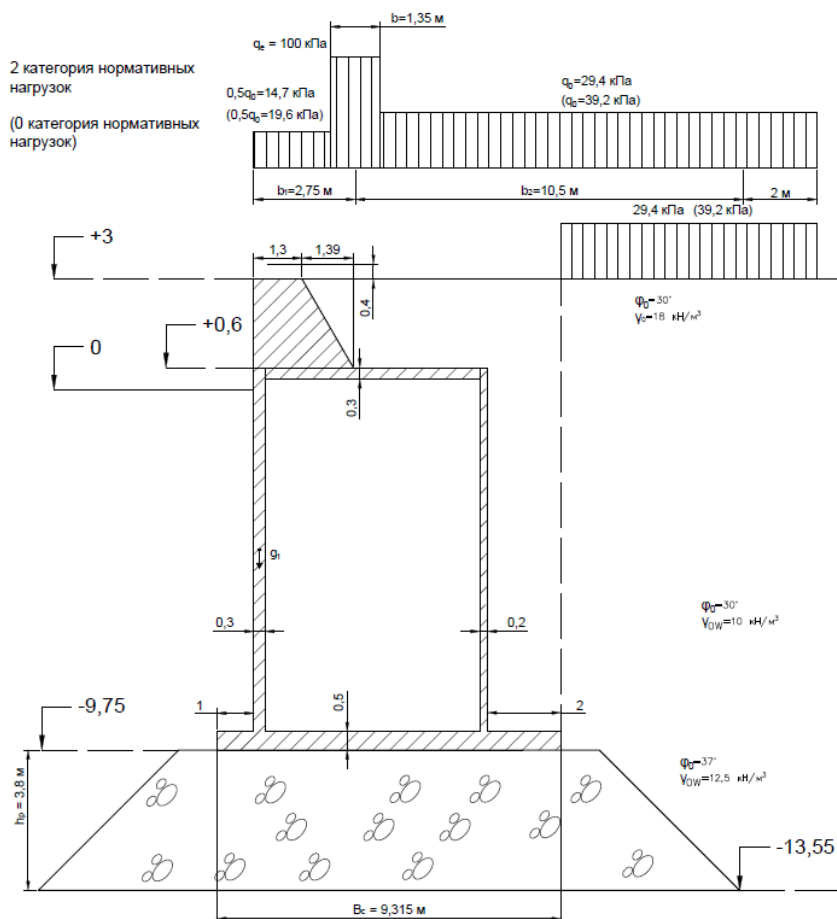


Рис. 1. Схема к расчету причала из массивов-гигантов, с категориями нормативных нагрузок 2 и 0.

Таблица 1. Значения активного давления грунта.

	γ , кН/м ³	x	$\lambda a x$	h, м	q, кПа	$\sigma(x)$, кПа	(x), тс/м ²	q, кПа	(x), кПа	(x), тс/м ²
					2 категория нагрузок			0 категория нагрузок		
над водой	18	3	0,258	0	0	0	0	0	0	0
	18	0,6	0,258	2,4	0	11,12	1,12	0	11,12	1,12
	18	0,6	0,298	2,4	0	12,87	1,29	0	12,87	1,29
	18	0,165	0,298	2,835	0	15,2	1,52	0	15,2	1,52
	18	0,165	0,298	2,835	29,4	23,96	2,4	39,2	26,87	2,69
	18	0	0,298	3	29,4	24,84	2,49	39,2	27,76	2,78
под водой	10	-9,75	0,298	9,75	29,4	53,87	5,39	39,2	56,79	5,68
	10	-9,75	0,249	9,75	29,4	44,97	4,5	39,2	47,41	4,75
	12,5	-13,55	0,249	3,8	29,4	56,78	5,68	39,2	59,22	5,93

Таблица 2. Значения горизонтальных сил и опрокидывающего момента.

2 категория						0 категория					
Горизонтальная сила			Опрокидывающий момент			Горизонтальная сила			Опрокидывающий момент		
	тс	кН/м		тсм	кН*м		тс	кН/м		тсм	кН*м
E _d	40,63	406,27	M _o	190,6	1906	E _d	43,5	435,2	M _o	205	2049,18

Таблица 3. Значения вертикальных сил и удерживающих моментов.

2 категория						0 категория					
Вертикальная сила			Удерживающий момент			Вертикальная сила			Удерживающий момент		
	тс	кН/м		тсм	кН*м		тс	кН/м		тсм	кН*м
G _c	116	1157	M _y	595	5949,4	G _c	117,37	1173,66	M _y	607,16	6071,59

Таблица 4. Напряжения на каменную постель и грунт основания.

Напряжения на каменную постель без учета нормативной нагрузки						Напряжения на каменную постель с учета нормативной нагрузки					
2 категория			0 категория			2 категория			0 категория		
σ_{max} , кПа	217,2		σ_{max} , кПа	225,84		σ_{max} , кПа	231,32		σ_{max} , кПа	239,11	
σ_{min} , кПа	31,21		σ_{min} , кПа	26,16		σ_{min} , кПа	85,64		σ_{min} , кПа	94,61	
Грунт основания											
2 категория						0 категория					
σ'_{max} , кПа			180,59			σ'_{max} , кПа			184,88		
σ'_{min} , кПа			100,36			σ'_{min} , кПа			105,31		

Сводные гистограммы:

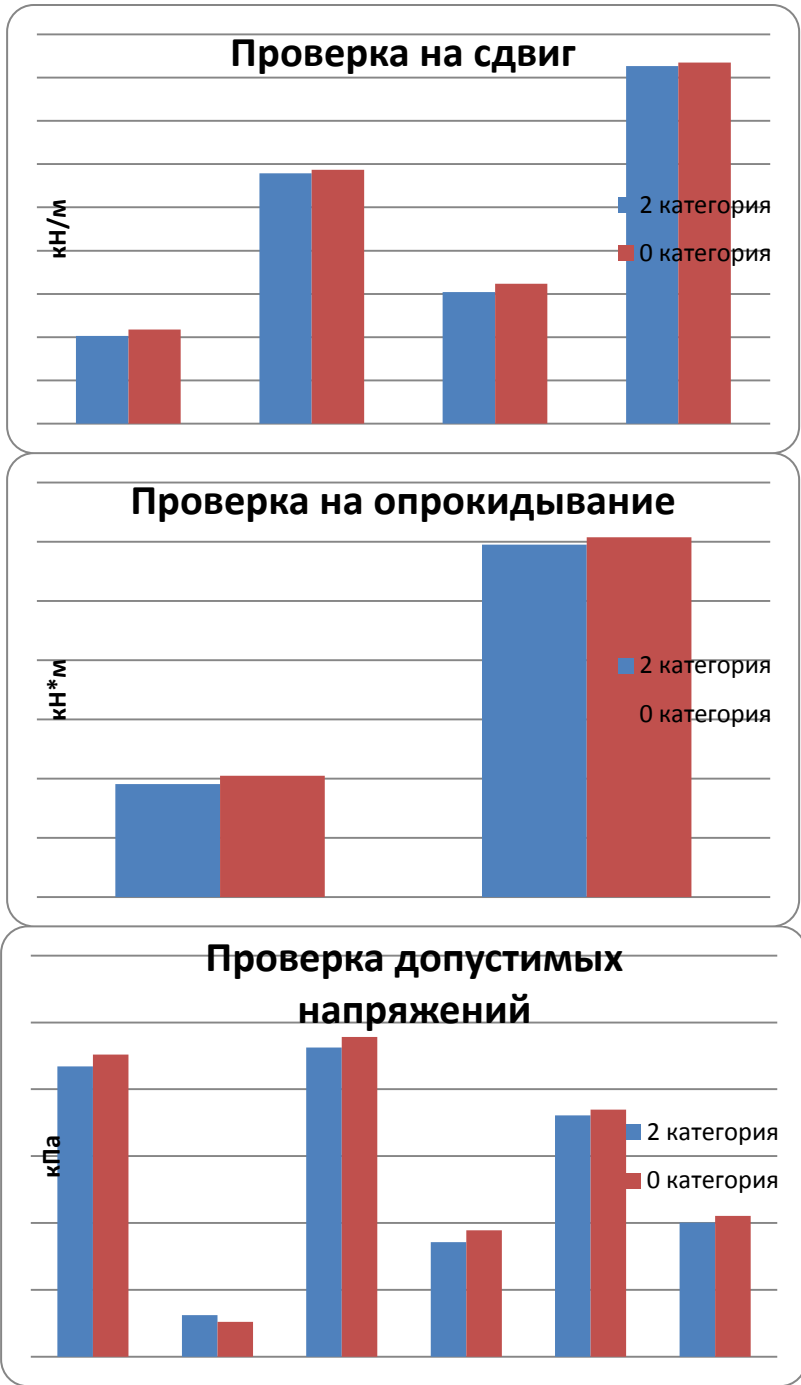


Рис. 2. Сводные гистограммы.

Наглядное противопоставление результатов в виде эпюр будут приведены на рис. 3.

Рис. 3. Эпюры бокового давления и напряжения от категорий нормативных нагрузок 2 и 0.

В результате проведенных поверочных расчетов установлено, что при перепрофилировании условного причала обеспечивается устойчивость и прочность причального сооружения, а рост напряжений не превышает 10%. Это дает основание утверждать, что для аналогичных по конструктивным решениям причалов их перепрофилирование возможно без дополнительных мероприятий по усилению.

Список литературы / References

1. СП 58.13330.2019 «Гидротехнические сооружения», дата введения 2020-06-17.
 2. Порты России // Федеральное агентство морского и речного транспорта. [Электронный ресурс]. Режим доступа: https://morflot.gov.ru/deyatelnost/napravleniya_deyatelnosti/portyi_rf.html/ (дата обращения: 01.04.2022).
 3. Грузооборот морских портов России за 12 месяцев 2021 г. // АССОЦИАЦИЯ МОРСКИХ ТОРГОВЫХ ПОРТОВ. [Электронный ресурс]. Режим доступа: <https://www.morport.com/rus/news/gruzooborot-morskih-portov-rossii-za-12-mesyacev-2021-g/> (дата обращения: 01.04.2022).
 4. Распоряжение от 30 сентября 2018 г. № 2101-р Москва // Правительство Российской Федерации. [Электронный ресурс]. Режим доступа: [http://static.government.ru/media/files/MUNhgWFddP3UfF9RJASDW9VxP8zwcB4Y.pdf/](http://static.government.ru/media/files/MUNhgWFddP3UfF9RJASDW9VxP8zwcB4Y.pdf) (дата обращения: 01.04.2022).
 5. Минтранс уточнил план модернизации портов до 2030 года // ИНТЕРФАКС. [Электронный ресурс]. Режим доступа: <https://www.interfax.ru/russia/726800/> (дата обращения: 01.04.2022).
 6. *Костин И.В.* Учебное пособие. «Причальные сооружения». Альтаир. Москва, 2013. С. 48. С. 24.
 7. *Костин И.В.* Учебное пособие. «Генеральный план морского порта». Альтаир. Москва, 2021. С. 65-67. С. 43.
 8. СП 350.1326000.2018 «Нормы технологического проектирования морских портов», таблица 4.3, дата введения: 2018-09-01.
-

**CREATE A PROGRAM THAT CAN RECOGNIZE DRAWINGS
DRAWN BY THE USER USING A LEARNING NEURAL
NETWORK IN THE OBJECT-ORIENTED JAVA LANGUAGE
Sinelnik K.E.¹, Limanova N.I.² (Russian Federation)**

¹Sinelnik Karim Eldarovich – student;

*²Limanova Natalia Igorevna – Scientific Supervisor, Head of the
Department,*

*DEPARTMENT OF INFORMATION SYSTEMS AND TECHNOLOGY,
VOLGA REGION STATE UNIVERSITY OF
TELECOMMUNICATIONS AND INFORMATICS,
SAMARA, RUSSIAN FEDERATION*

Abstract: *this article describes how using the object-oriented Java language, as well as using a library with a neural network, a program was created that can recognize the drawings that the user drew in the program window. The essence lies precisely in the learning process of the program, how and with what it determines what is drawn, the article pays much attention to the learning process of a neural network. The program code and screenshots also presented, which show how the application works. In conclusion, experiments presented in which it is seen how the neural network copes with the difficulties encountered during the issuance of the result to the user of this program.*

Keywords: *neural network, drawings, learning, programming language, java, development.*

СОЗДАТЬ ПРОГРАММУ, КОТОРАЯ СМОЖЕТ РАСПОЗНАВАТЬ РИСУНКИ, НАРИСОВАННЫЕ ПОЛЬЗОВАТЕЛЕМ С ПОМОЩЬЮ ОБУЧАЮЩЕЙСЯ НЕЙРОННОЙ СЕТИ НА ОБЪЕКТНО- ОРИЕНТИРОВАННОМ ЯЗЫКЕ JAVA

Синельник К.Э.¹, Лиманова Н.И.² (Российская Федерация)

¹Синельник Карим Эльдарович – студент,
Институт информатики и кибернетики;

²Лиманова Наталья Игоревна – научный руководитель,
заведующая кафедрой,
кафедра Информационных систем и технологии,
Поволжский государственный университет телекоммуникаций и
информатики,
г. Самара, Российская Федерация

Аннотация: в данной статье рассказано, как с помощью объектно-ориентированного языка Java, а так же использования библиотеки с нейронной сетью, была создана программа, которая может распознавать рисунки, которые нарисовал пользователь в окне программы. Суть заключается именно в процессе обучения программы, как и с помощью чего она определяет что нарисованно, в статье уделено много внимания процессу обучения нейронной сети. Так же представлен код программы и скриншоты на которых виден принцип работы приложения. В заключении представлены эксперименты в которых видно, как нейросеть справляется с возникшими трудностями во время выдачи результата пользователю данной программы.

Ключевые слова: нейронная сеть, рисунки, обучение, язык программирования, java, разработка.

Программа была выполнена с помощью языка программирования java script для большей скорости работы и производительности, поддержке скриптов всеми популярными браузерами, понятному синтаксису для новичка.

В ходе работы мы использовали `brain.js`-это библиотека с открытым кодом на языке **JavaScript**, используемая для запуска и обработки нейронных сетей.

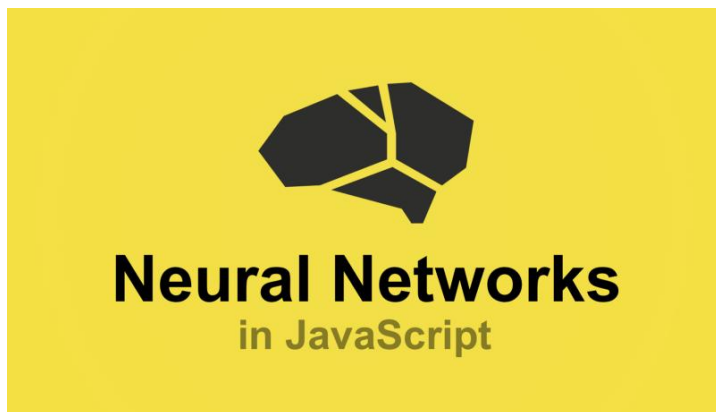


Рис.1 библиотека Brain.js.

`Brain.js` используется, как правило, с `Node.js` или с браузером на стороне клиента для тренировки моделей машинного обучения. Она позволяет легко создавать нейронные сети, а затем обучать их на основе входных / выходных данных.

Также имеются другие библиотеки для нейронной сети такие как:

- **`Limdu.js`**

Это платформа машинного обучения, используемая для `Node.js`.

`Limdu.js` идеально подходит для виртуальных собеседников (чат-ботов), обработки естественного языка и других диалоговых систем.`stdLib`.



Рис.2. библиотека stdLib.

Эта библиотека `JavaScript` используется для создания продвинутых статистических моделей и библиотек машинного

обучения. Кроме того, она может использоваться в графических средствах отображения информации для разведочного анализа данных, а также визуализации данных. Но для нашей работы была выбрана brain.js.

Что такое нейронные сети?

Нейронные сети возникли из исследований в области искусственного интеллекта, а именно, из попыток воспроизвести способность биологических нервных систем обучаться и исправлять ошибки, моделируя низкоуровневую структуру мозга. В простейшем случае она состоит из нескольких соединенных между собой нейронов.

Математический нейрон

Несложный автомат, преобразующий входные сигналы в результирующий выходной сигнал.

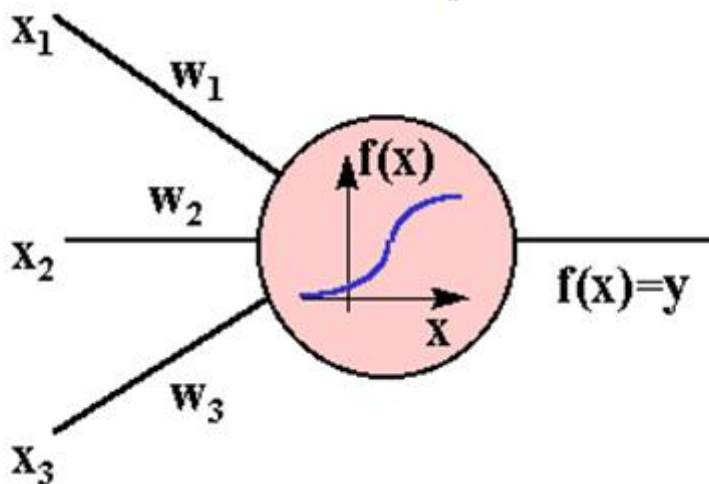


Рис.3. Модель математического нейрона.

Сигналы $x_1, x_2, x_3 \dots x_n$, поступая на вход, преобразуются линейным образом, т.е. к телу нейрона поступают силы: $w_1x_1, w_2x_2, w_3x_3 \dots w_nx_n$, где w_i — веса соответствующих сигналов. Нейрон суммирует эти сигналы, затем применяет к сумме некоторую функцию $f(x)$ и выдаёт полученный выходной сигнал. y . В качестве функции $f(x)$ чаще всего используется сигмоидная или пороговая функции.

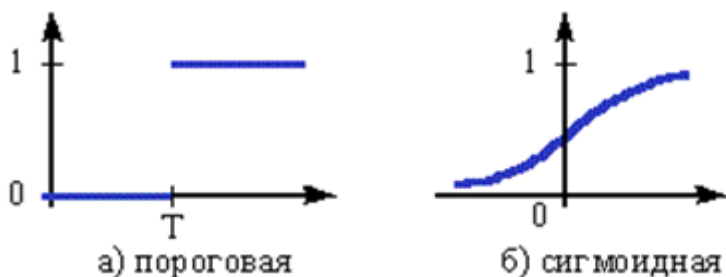


Рис.4. Виды функций полученного выходного сигнала.

Пороговая функция может принимать только два дискретных значения 0 или 1. Смена значения функции происходит при переходе через заданный порог T .

Сигмоидная — непрерывная функция, может принимать бесконечно много значений в диапазоне от 0 до 1.

Типы нейронных сетей

За период развития, нейронные сети поделились на множество типов, которые переплетаются между собой в различных задачах. На данный момент сложно классифицировать какую-либо сеть только по одному признаку. Это можно сделать по принципу применения, типу входной информации, характеру обучения, характеру связей, сфере применения.

Сверточные

Один из популярнейших типов сети, часто используемый для распознавания той или иной информации в фотографиях и видео, обработке языка, системах для рекомендаций. Основные характеристики:

1. Отличная масштабируемость — проводят распознавания образов любого разрешения (какое бы не было оно большое).

2. Использование объемных трехмерных нейронов — внутри слоя, нейроны связаны малым полем, именуемы рецептивным слоем.

3. Механизм пространственной локализации — соседние слои нейронов связаны таким механизмом, за счет чего обеспечивается работа нелинейных фильтров и охват все большего числа пикселей графического изображения.

Идея сложной системы этого типа нейросети возникла при тщательном изучении зрительной коры, которая в больших

полушариях мозга отвечает за обработку визуальной составляющей. Основным критерий выбора в пользу свёрточного типа – она в составе технологий глубокого обучения. Схожий тип с перцептроном, но разница в том, что здесь используется ограниченная матрица весов, сдвигаемая по обрабатываемому слою, вместо полносвязной нейронной сети.

Рекуррентные

Этот тип нейросети, в котором связи между элементами могут обрабатывать серии различных событий во времени или работать с последовательными цепочками в пространстве. Такой тип часто применяют там, где что-то целое разбито на куски. Например, распознавание речи или рукописного текста. От нее пошло множество видов сетей, в том числе Хопфилда, Элмана и Джордана.

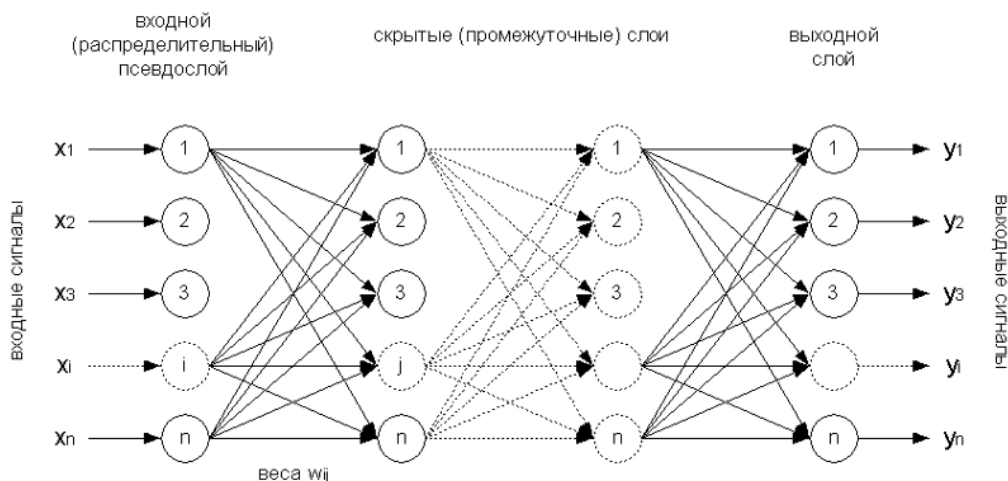


Рис.5. Архитектура нейронной сети - Perceptron.

Архитектура нейронной сети может быть разной, мы рассмотрим одну из простых реализаций нейронной сети – **Perceptron**.

Есть слой входных нейронов (где информация поступает извне), слой выходных нейронов (откуда можно взять результат) и ряд, так называемых, скрытых слоев между ними. Нейроны могут быть расположены в несколько слоёв. Каждая связь между нейронами имеет свой вес **W_{ij}** .

Входные и выходные сигналы

Перед тем, как подавать сигналы на нейроны входящего слоя сети нам их нужно нормализовать. Нормализация входных данных — это процесс, при котором все входные данные проходят процесс «выравнивания», т.е. приведения к интервалу $[0,1]$ или $[-1,1]$. Если не провести нормализацию, то входные данные будут оказывать дополнительное влияние на нейрон, что приведет к неверным решениям. Другими словами, как можно сравнивать величины разных порядков?

На нейронах выходного слоя у нас тоже не будет чистой «1» или «0», это нормально. Есть некий порог, при котором мы будем считать, что получили «1» или «0». Про интерпретацию результатов поговорим позже.

Обучение нейронной сети

Для конструирования процесса обучения, прежде всего, необходимо иметь модель внешней среды, в которой функционирует нейронная сеть - знать доступную для сети информацию.

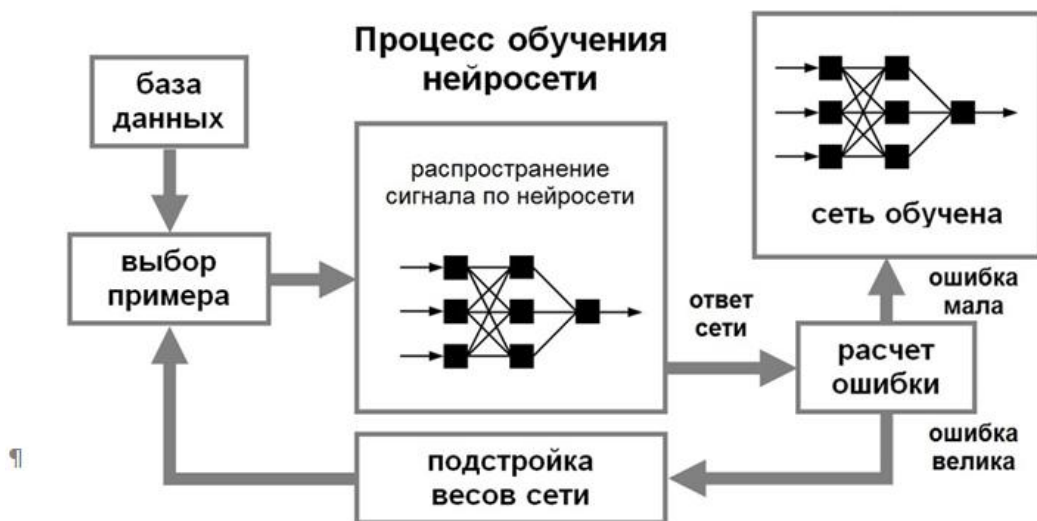


Рис.6. Схема процесса обучения нейронной сети.

Эта модель определяет парадигму обучения. Во-вторых, необходимо понять, как модифицировать весовые параметры сети - какие правила обучения управляют процессом настройки.

Алгоритм обучения означает процедуру, в которой используются правила обучения для настройки весов.

Существуют три парадигмы обучения: "с учителем", "без учителя" (самообучение) и смешанная. В первом случае нейронная сеть располагает правильными ответами (выходами сети) на каждый входной пример. Веса настраиваются так, чтобы сеть производила ответы как можно более близкие к известным правильным ответам. Усиленный вариант обучения с учителем предполагает, что известна только критическая оценка правильности выхода нейронной сети, но не сами правильные значения выхода. Обучение без учителя не требует знания правильных ответов на каждый пример обучающей выборки. В этом случае раскрывается внутренняя структура данных или Корреляции между образцами в системе данных, что позволяет распределить образцы по категориям. При смешанном обучении часть весов определяется посредством обучения с учителем, в то время как остальная получается с помощью самообучения.

Теория обучения рассматривает три фундаментальных свойства, связанных с обучением по примерам: емкость, сложность образцов и вычислительная сложность. Под емкостью понимается, сколько образцов может запомнить сеть, и какие функции и границы принятия решений могут быть на ней сформированы. Сложность образцов определяет число обучающих примеров, необходимых для достижения способности сети к обобщению. Слишком малое число примеров может вызвать «переобученность» сети, когда она хорошо функционирует на примерах обучающей выборки, но плохо – на тестовых примерах, подчиненных тому же статистическому распределению. Известны 4 основных типа правил обучения: коррекция по ошибке, машина Больцмана, правило Хебба и обучение методом соревнования.

Правило коррекции по ошибке. При обучении с учителем для каждого входного примера задан желаемый выход d . Реальный выход сети y может не совпадать с желаемым. Принцип коррекции по ошибке при обучении состоит в использовании сигнала $(d-y)$ для модификации весов, обеспечивающей постепенное уменьшение ошибки. Обучение имеет место только в

случае, когда перцептрон ошибается. Известны различные модификации этого алгоритма обучения.

Обучение Больцмана. Представляет собой стохастическое правило обучения, которое следует из информационных теоретических и термодинамических принципов. Целью обучения Больцмана является такая настройка весовых коэффициентов, при которой состояния видимых нейронов удовлетворяют желаемому распределению вероятностей. Обучение Больцмана может рассматриваться как специальный случай коррекции по ошибке, в котором под ошибкой понимается расхождение Корреляций состояний в двух режимах Правила Хебба. Самым старым обучающим правилом является постулат обучения Хебба. Хебб опирался на следующие нейрофизиологические наблюдения: если нейроны с обеих сторон синапса активизируются одновременно и регулярно, то сила синаптической связи возрастает. Важной особенностью этого правила является то, что изменение синаптического веса зависит только от активности нейронов, которые связаны данным синапсом. Это существенно упрощает цепи обучения в реализации VLSI.

Обучение методом соревнования. В отличие от обучения Хебба, в котором множество выходных нейронов могут возбуждаться одновременно, при соревновательном обучении выходные нейроны соревнуются между собой за активизацию. Это явление известно как правило "победитель берет все". Подобное обучение имеет место в биологических нейронных сетях.

Обучение посредством соревнования позволяет кластеризовать входные данные: подобные примеры группируются сетью в соответствии с корреляциями и представляются одним элементом. При обучении модифицируются только веса "победившего" нейрона. Эффект этого правила достигается за счет такого изменения сохраненного в сети образца (вектора весов связей победившего нейрона), при котором он становится чуть ближе к входному примеру.

Код программы

Для работы с нейронной сетью были созданы следующие файлы.

- **Index.html**- в данной файле будет происходить основная работа.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Нейросеть</title>
    <link rel="stylesheet" href="style.css"> // подключение стилей
  </head>
  <body>
    //подключение canvas для создания рисунков
    <canvas id="canvas"></canvas>
    //использование библиотеки brain.js
    <script src="brain.js"></script>
    // подключение app.js для работы кода
    <script src="app.js"></script>
  </body>
</html>
```

Рис.7. Index.html.

- **Style.css** - для создания внешнего вида и цвета полотна.

```
body {
  background-color: #000000;
  display: flex;
  justify-content: center;
  align-items: center;
  height: 100vh;
}

#canvas {
  background-color: #fff;
}
```

Рис.8. Style.css.

- **App.js** – код java script.

```

//создание элемента canvas
const canvas = document.getElementById( elementId: 'canvas')
// создание класса paint для рисования
class Paint {
  // создание конструктора для определения размеров полотна
  constructor(width, height, pixelSize, color) {
    canvas.width = width
    canvas.height = height
    this.canvas = canvas
    this.ctx = this.canvas.getContext( contextId: '2d')
  }
  // отвечают за цвет и ширину линии
  this.pixelSize = pixelSize
  this.color = color
  this.isMouseDown = false
}
//данный метод создан для того,чтобы линия рисовалась окружностями
drawCircle(x, y, diameter = this.pixelSize, color = this.color) {
  this.ctx.beginPath()
  this.ctx.fillStyle = color
  this.ctx.arc(x, y, radius: diameter / 2, startAngle: 0, endAngle: Math.PI * 2)
  this.ctx.fill()
}
// данный метод -это линия которая связывает окружности
drawConnectingLine(x, y, lineWidth = this.pixelSize, color = this.color) {
  this.ctx.lineTo(x, y)
  this.ctx.lineWidth = lineWidth
  this.ctx.strokeStyle = color
  this.ctx.stroke()
}
//запоминание позиции
storePosition(x, y) {
  this.ctx.beginPath()
  this.ctx.moveTo(x, y)
}
// создание линии для сетки,чтобы нейросеть запоминала изображение
drawLine(x1, y1, x2, y2, lineWidth :number = 2, color :string = '#00bfff') {
  this.ctx.beginPath()
  this.ctx.moveTo(x1, y1)
  this.ctx.lineWidth = lineWidth
  this.ctx.strokeStyle = color
  this.ctx.lineTo(x2, y2)
  this.ctx.stroke()
}

```

Рис.9. App.js.

```

        this.ctx.stroke()
    }
    // рисование ячейки
    drawCell(x1, y1, size = this.pixelSize, color :string = '#00bfff') {
        this.ctx.beginPath()
        this.ctx.fillStyle = color
        this.ctx.rect(x1, y1, size, size)
        this.ctx.fill()
    }
    // создание числового представления
    //если в сетке клетки не закрашены,то ставится 0
    //если в сетке клетка закрашена,то ставится 1
    toNumericalRepresentation(shouldBeDrawn) {
        const numericalRepresentation = []
        let cellsToDraw = []

        for (let x = 0; x < this.canvas.width; x += this.pixelSize) {
            for (let y = 0; y < this.canvas.height; y += this.pixelSize) {
                const cell = this.ctx.getImageData(x, y, this.pixelSize, this.pixelSize)
                let isFilled = false

                for(let i = 0; i < cell.data.length; i += 10) {
                    if (cell.data[i] !== 0) {
                        isFilled = true
                        break
                    }
                }

                if (isFilled) {
                    numericalRepresentation.push(1)
                    cellsToDraw.push({ x, y })
                } else {
                    numericalRepresentation.push(0)
                }
            }
        }

        if (shouldBeDrawn) {
            cellsToDraw.forEach((cell) => this.drawCell(cell.x, cell.y))
        }

        return numericalRepresentation
    }

```

Puc.10. App.js.

```

    if (shouldBeDrawn) {
      cellsToDraw.forEach((cell) => this.drawCell(cell.x, cell.y))
    }

    return numericalRepresentation
  }

  // создание сетки
  drawGrid() {
    for (let x = 0; x < this.canvas.width; x += this.pixelSize) {
      this.drawLine(x, 0, x, this.canvas.height)
    }

    for (let y = 0; y < this.canvas.height; y += this.pixelSize) {
      this.drawLine(0, y, this.canvas.width, y)
    }
  }

  clear() {
    this.ctx.clearRect(0, 0, this.canvas.width, this.canvas.height)
  }

  // задание параметров для рисования
  run() {
    this.canvas.addEventListener('mousedown', listener: () => {
      this.isMouseDown = true
      this.ctx.beginPath()
    })

    this.canvas.addEventListener('mouseup', listener: () => {
      this.isMouseDown = false
    })

    this.canvas.addEventListener('mousemove', listener: (e: MouseEvent) => {
      if (this.isMouseDown) {
        const x = e.offsetX
        const y = e.offsetY

        this.drawConnectingLine(x, y)
        this.drawCircle(x, y)
        this.storePosition(x, y)
      }
    })
  }
}

```

Рис. 11. App.js.


```

    }
    // присвоение размеров полотна и ширину, цвета линии
    const paint = new Paint( width: 600, height: 400, pixelSize: 15, color: 'purple')
    paint.run()
    //вывести ответ
    const statistic = []
    const addToStatistic = (name, numericalRepresentation) => {
    statistic.push({
        input: numericalRepresentation,
        output: {
            [name]: 1
        }
    })
    }
    // нажатие на кнопку С - стирается изображение
    document.addEventListener( type: 'keypress', listener: ({ code :string }) => {
    if (code === 'KeyC') {
        paint.clear()
    }
    // нажатие на кнопку V - нейросеть запоминает изображение
    if (code === 'KeyV') {
        const numericalRepresentation = paint.toNumericalRepresentation( shouldBeDrawn: true)
        addToStatistic(prompt( message: 'ЧТО ИЗОБРАЖЕНО?'), numericalRepresentation)
    }
    // нажатие на кнопку B -запускается нейросеть и угадывает изображение
    if (code === 'KeyB') {
        const neuralNetwork = new brain.NeuralNetwork()
        neuralNetwork.train(statistic, { log: true })
        const numericalRepresentation = paint.toNumericalRepresentation()
        const result = brain.likely(numericalRepresentation, neuralNetwork)

        if (confirm('НА РИСУНКЕ ИЗООБРАЖЕН ${result}. Я ПРАВ?')) {
            addToStatistic(result, numericalRepresentation)
        } else {
            addToStatistic(prompt( message: 'ЕСЛИ НЕТ,ТО ЧТО ИЗООБРАЖЕНО?'), numericalRepresentation)
        }

        console.log(statistic)
    }
    })
}

```

Рис.12. App.js.

Для анализа статистических данных и последующем угадыванием изображения, самым верным будет разбиения изображения(полотна) на сетку, которая будет проверять каждый пиксель, если пиксель закрашен, то ставится 1, если пиксель пустой, то ставится 0.

Для этого в программе создан метод DrawGrid, который будет рисовать данную сетку, чтобы не создавать код на черчение линий, можно сделать функцию, которая названа DrawLine, которая будет принимать начальные и конечные координаты точки, ширину линии и цвет.

```
// создание сетки
drawGrid() {
  for (let x = 0; x < this.canvas.width; x += this.pixelSize) {
    this.drawLine(x, 0, x, this.canvas.height)
  }

  for (let y = 0; y < this.canvas.height; y += this.pixelSize) {
    this.drawLine(0, y, this.canvas.width, y)
  }
}

clear() {
  this.ctx.clearRect(0, 0, this.canvas.width, this.canvas.height)
}
```

Рис.13. Создание сетки.

```
// создание линии для сетки, чтобы нейросеть запоминала изображение
drawLine(x1, y1, x2, y2, lineWidth : number = 2, color : string = '#00bfff') {
  this.ctx.beginPath()
  this.ctx.moveTo(x1, y1)
  this.ctx.lineWidth = lineWidth
  this.ctx.strokeStyle = color
  this.ctx.lineTo(x2, y2)
  this.ctx.stroke()
}
```

Рис.14. Создание линии для сетки.

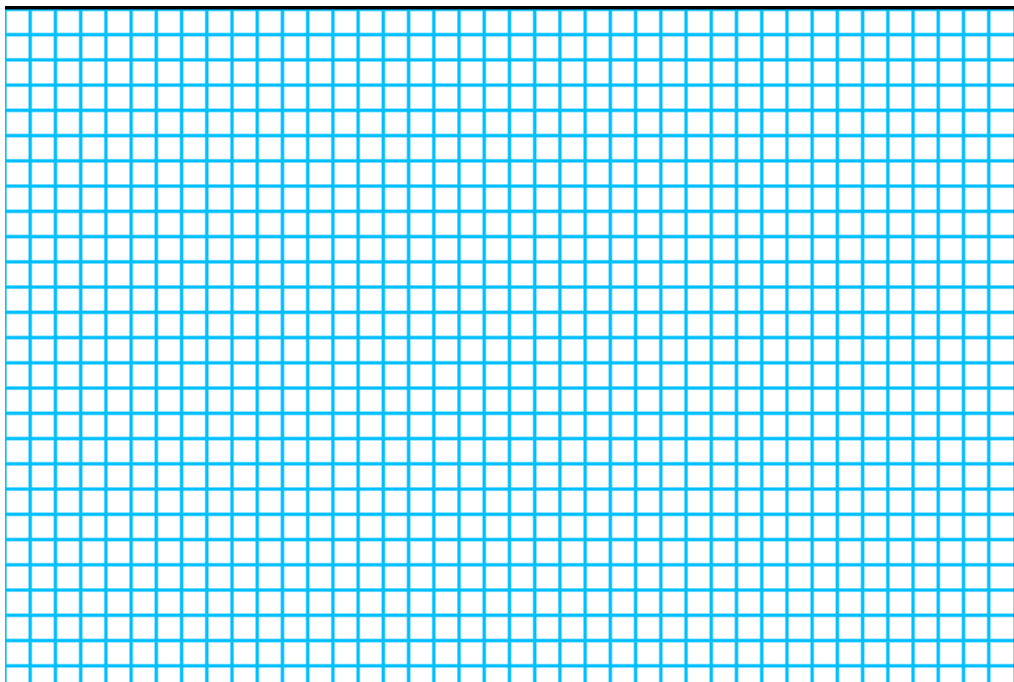


Рис.15. Вид полученной сетки.

Так выглядит разбиение палитры, где каждая клетка является пикселем и будет закрашивать зарисованные пиксели для сохранения статистических данных.

Дальше, если клетка закрашена пользователем, то чтобы нейросеть понимала, что она закрашена, вся клетка будет закрашена синим и ставится 1, если клетка пуста, то 0.

Для этого был создан метод `DrawCell`, который будет закрашивать ячейку. Так же создан метод `ToNumericalRepresentation`, - это является числовым представлением ячейки, было описано ранее (палитра переводится в 1 и 0).

```
drawCell(x1, y1, size = this.pixelSize, color :string = '#00bfff') {  
    this.ctx.beginPath()  
    this.ctx.fillStyle = color  
    this.ctx.rect(x1, y1, size, size)  
    this.ctx.fill()  
}
```

Рис.16. Разбиение палитры на ячейки.

```

// создание числового представления
// создание числового представления
//если в сетке клетки не закрашены,то ставится 0
//если в сетке клетка закрашена,то ставится 1
toNumericalRepresentation(shouldBeDrawn) {
  const numericalRepresentation = []
  let cellsToDraw = []

  for (let x = 0; x < this.canvas.width; x += this.pixelSize) {
    for (let y = 0; y < this.canvas.height; y += this.pixelSize) {
      const cell = this.ctx.getImageData(x, y, this.pixelSize, this.pixelSize)
      let isFilled = false

      for(let i = 0; i < cell.data.length; i += 10) {
        if (cell.data[i] !== 0) {
          isFilled = true
          break
        }
      }

      if (isFilled) {
        numericalRepresentation.push(1)
        cellsToDraw.push({ x, y })
      } else {
        numericalRepresentation.push(0)
      }
    }
  }

  if (shouldBeDrawn) {
    cellsToDraw.forEach((cell) => this.drawCell(cell.x, cell.y))
  }

  return numericalRepresentation
}

```

Рис.17. Числовое представление ячейки.

Эксперименты с рисунками

Сначала мы проводим краткое обучение нейронной сети, чтобы у него появились статистические данные рисунков, с которыми она должна будет сравнивать изображение нарисованное пользователем и затем выдавать ему результат (определить, что нарисовал пользователь).

Пример обучения:

1. Нас приветствует начальный экран, на котором ничего нет.

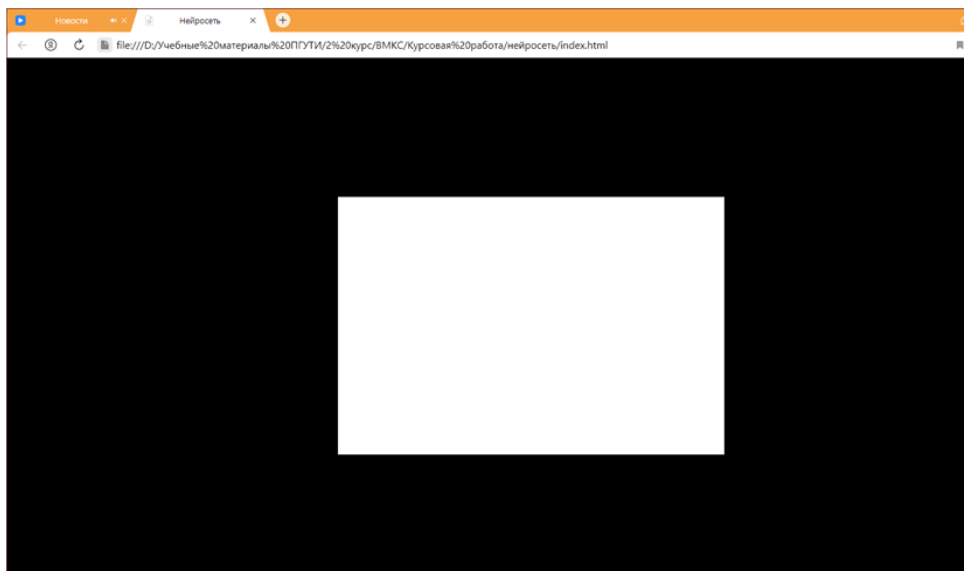


Рис.18. Начальный экран.

2. Мы рисуем изображение на нём.

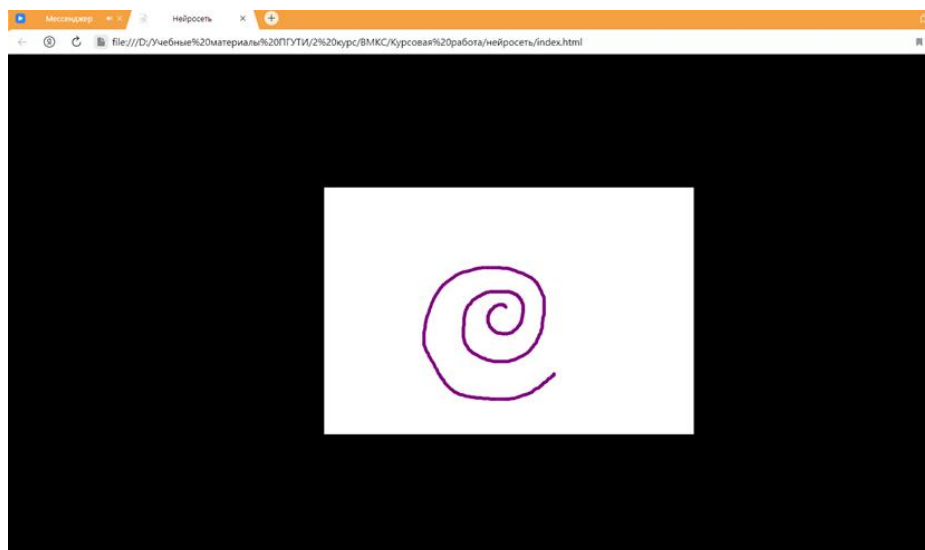


Рис.19. Нарисованное изображение на экране.

3. Вносим наш рисунок в базу данных нейросети.

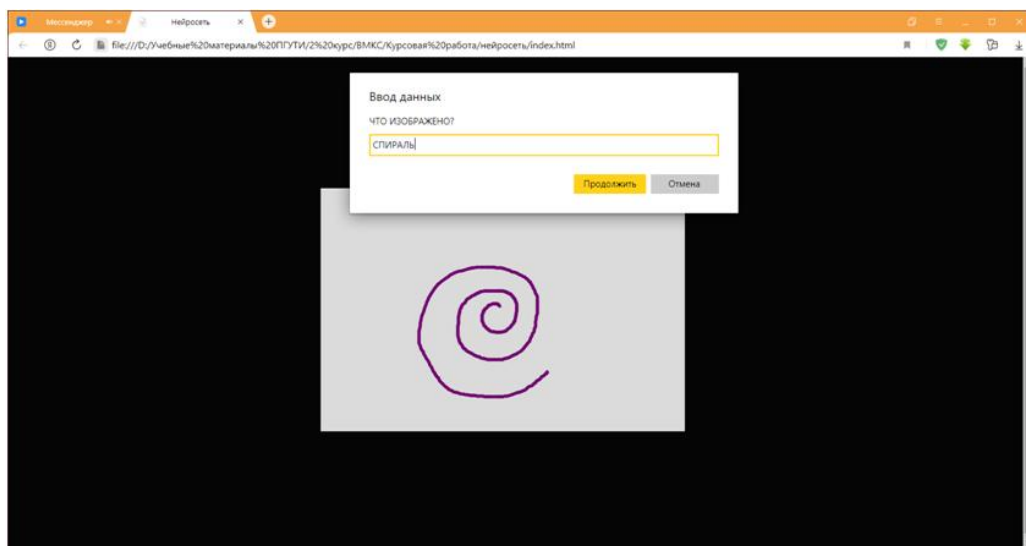


Рис.20. Внесение рисунка в базу данных нейросети.

4. Видим, что нейросеть запомнила наш рисунок.

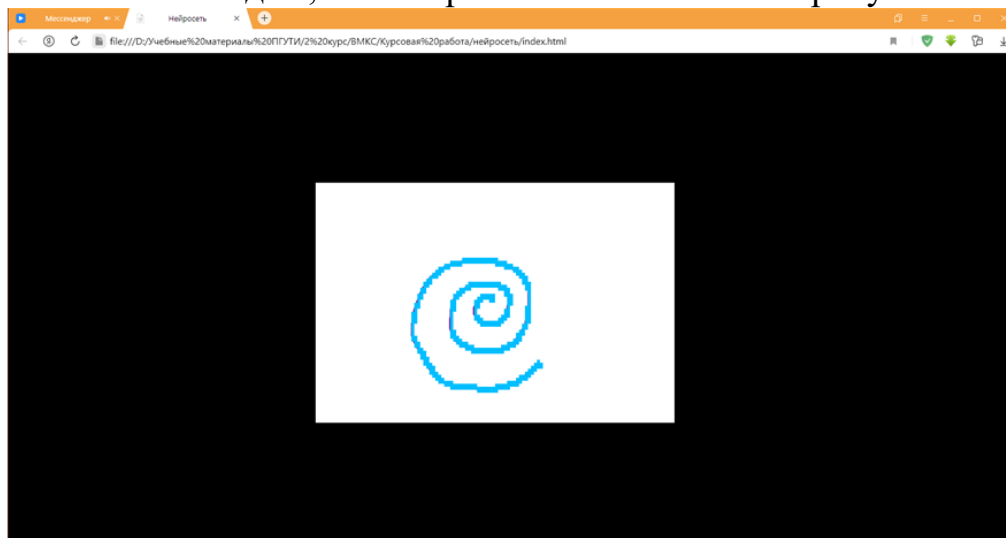


Рис.21. Нейросеть запоминает наш рисунок.

5. Рисуем наш рисунок и просим у нейросети дать результат, и как видим она это делает без ошибки.

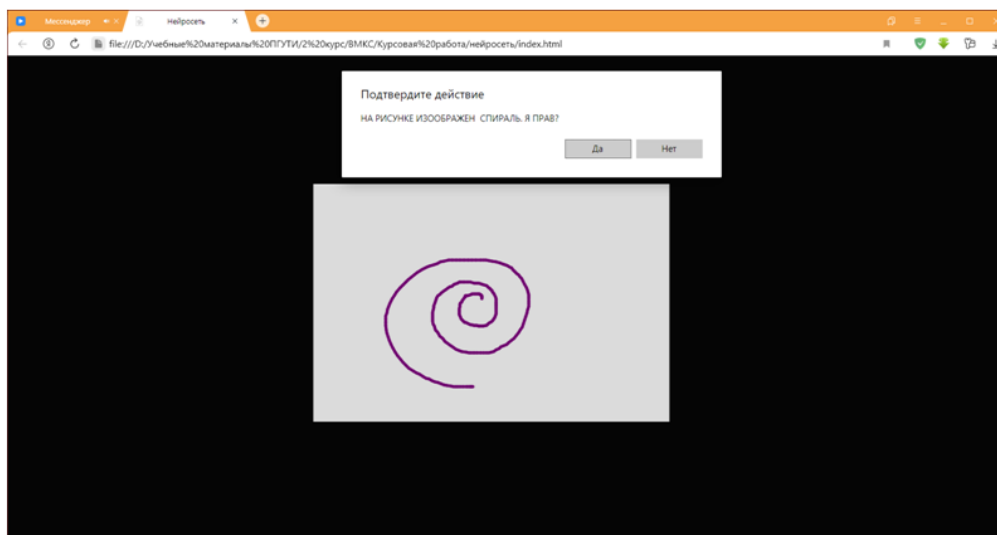


Рис.22. Нейросеть выдаёт результат.

Теперь мы внесём в базу нейросети 3 рисунка, которые будут являться сложными, ибо простые рисунки нейросеть угадывает легко, после мы будем составлять статистику, которая покажет, насколько часто нейросеть будет совершать ошибки, при попытке угадывания рисунка нарисованным пользователем.

1. Кот

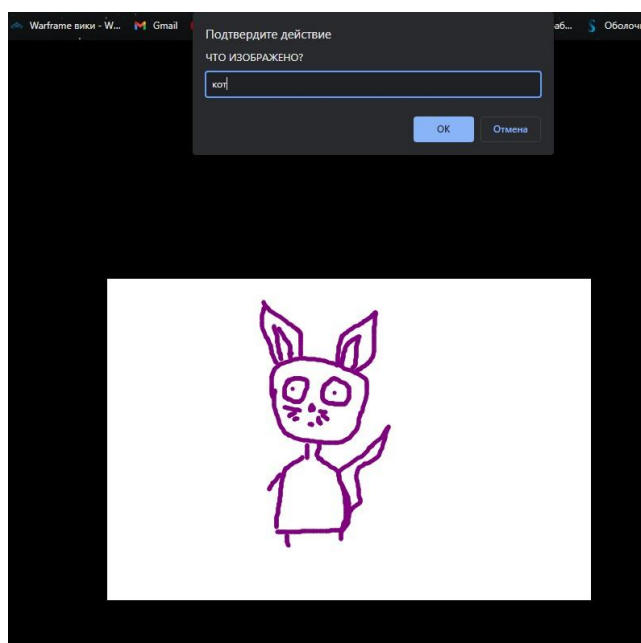


Рис.23. Внесение рисунка в базу нейросети.

2.

Дом

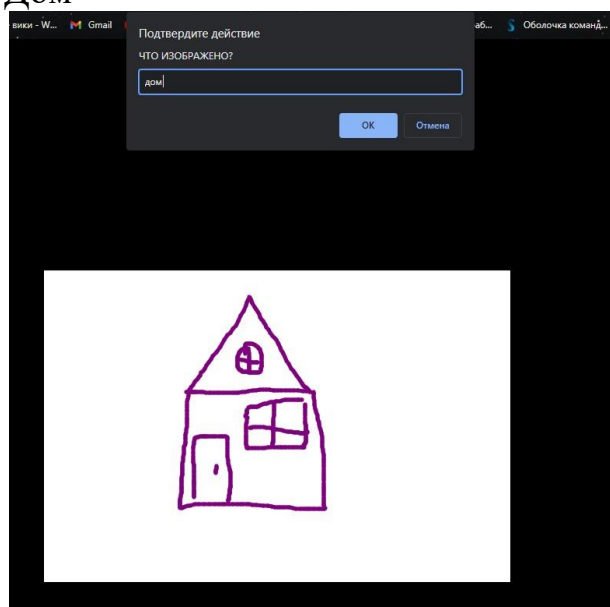


Рис.24. Внесение рисунка в базу нейросети.

3.

Пальмы

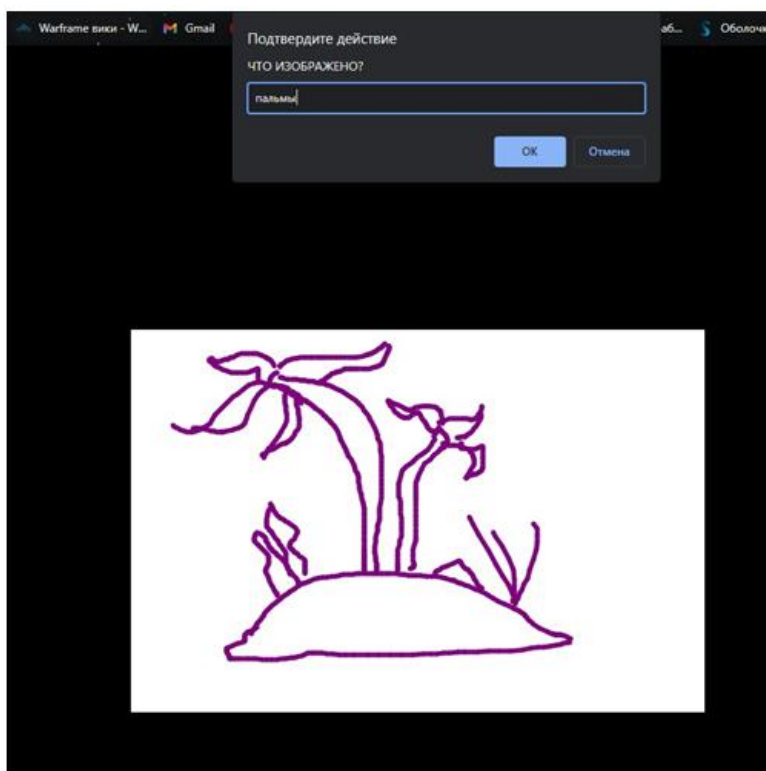


Рис.25. Внесение рисунка в базу нейросети.

Мы внесли рисунки в базу нейросети и теперь можно начинать эксперимент. Нарисуем 100 рисунков и посчитаем количество неверных результатов. Отталкиваясь от статистики, составим график неверных результатов.

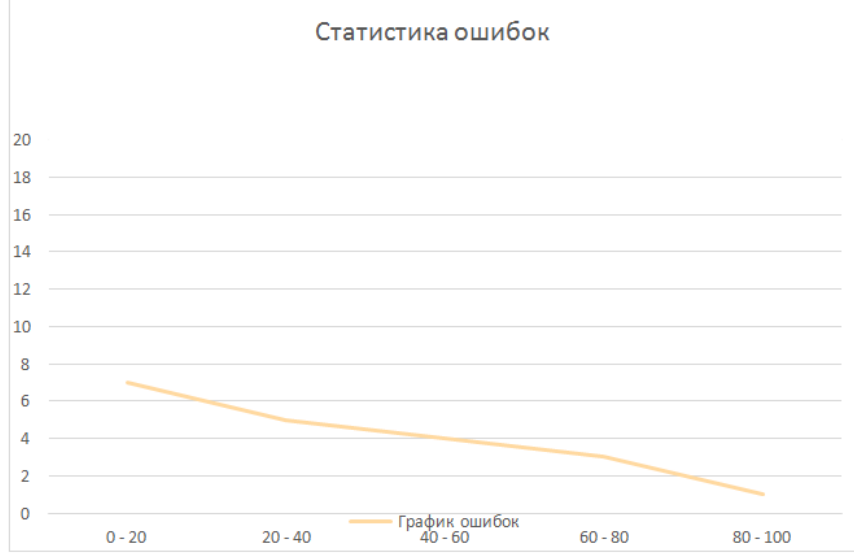


Рис.26. График ошибок нейросети №1.

Как показано на графике, нейросеть обучалась, и с каждым последующим рисунком вероятность получения неверного результата уменьшалась.

Теперь усложним задачу для нашей нейросети, мы нарисуем снова 100 рисунков, но уже не по центру начального экрана, а где-нибудь в углу, рисовать фигуры разного размера, прямые линии делать более кривыми, и посмотрим, как измениться график ошибок нейросети.

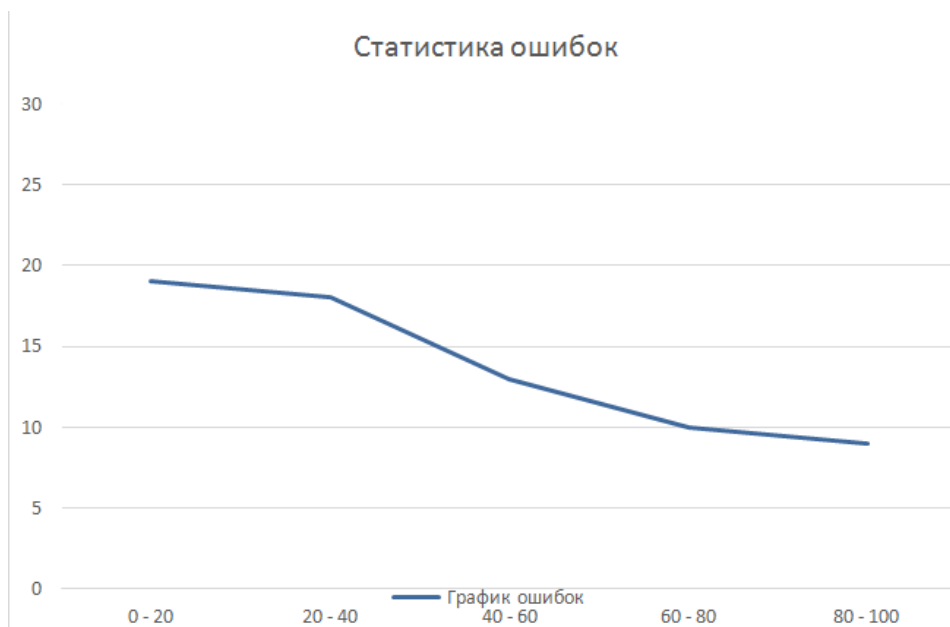


Рис.27. График ошибок нейросети №2.

Как мы можем заметить, в первых двадцати рисунках количество ошибок резко увеличилось, как в последующих, но мы можем отследить положительную динамику, ибо видно, что нейросеть так же продолжает обучаться и ошибок с последующими рисунками становится меньше.

Заключение

Вывод: благодаря, этим теоретическим данным, нейросети Brain.js и возможностям языка JavaScript, мы смогли создать программу, которая может с помощью нейронной сети научиться определять то, что нарисовал пользователь, запоминать это и выводить сообщение пользователю, в котором написано, что он нарисовал на экране.

Список литературы / References

1. Скачать библиотеку. [Электронный ресурс]. Режим доступа: [//github.com/BrainJS/brain.js](https://github.com/BrainJS/brain.js) (дата обращения: 28.11.2022).
2. Информация о нейросетях. [Электронный ресурс]. Режим доступа: <https://habr.com/ru/post/304414/> (дата обращения: 28.11.2022).

3. Принципы построения нейронных сетей. [Электронный ресурс]. Режим доступа: <https://web-panda.ru/post/brainjs-simple> (дата обращения: 28.11.2022).
4. Основные принципы работы [Электронный ресурс]. Режим доступа: <https://medium.com/@stasonmars/основы-основ-в-javascript> (дата обращения: 28.11.2022).
5. Как пользоваться библиотекой. [Электронный ресурс]. Режим доступа: <https://www.youtube.com/watch?v=GUzhH6tTkV4> (дата обращения: 28.11.2022).

HOW ALPHAGO WORKS – THE FIRST PROGRAM TO BEAT A HUMAN IN THE GAME OF GO

Karnaukhov A.V. (Russian Federation)

*Karnaukhov Arseniy Viktorovich – undergraduate Student,
DEPARTMENT OF APPLIED MATHEMATICS,
NATIONAL RESEARCH UNIVERSITY HIGHER SCHOOL OF
ECONOMICS,
MOSCOW, RUSSIAN FEDERATION*

Abstract: *this article analyzes the game of Go that has long been considered the most difficult of the classic computer games because of its huge space of possible positions, as well as the difficulty of evaluating them. The breakthrough in the field of computer Go was made by the AlphaGo program. AlphaGo was the first program to beat a professional player in Go. It was developed by the Google DeepMind in 2015 and described in the article [1]. The purpose of this paper is to give the reader a clear understanding of how the AlphaGo algorithm works and what ideas it uses.*

Keywords: *AlphaGo, network, Go, move, position.*

КАК РАБОТАЕТ АЛЬФАГО – ПЕРВАЯ ПРОГРАММА, ПОБЕЖДАЮЩАЯ ЧЕЛОВЕКА В ИГРЕ ГО

Карнаухов А.В. (Российская Федерация)

*Карнаухов Арсений Викторович – студент бакалавриата,
кафедра прикладной математики,
Национальный исследовательский университет «Высшая школа
экономики,
г. Москва, Российская Федерация*

Аннотация: *в данной статье анализируется игра Го, которая долгое время считалась самой сложной из классических компьютерных игр из-за огромного пространства возможных позиций, а также сложности их оценки. Прорыв в области компьютерного го совершила программа AlphaGo. AlphaGo была первой программой, обыгравшей профессионального игрока в го. Она была разработана Google DeepMind в 2015 году и описана в статье [1]. Цель этой статьи — дать читателю четкое представление о том, как работает алгоритм AlphaGo и какие идеи он использует.*

Ключевые слова: *AlphaGo, сеть, Go, ход, позиция.*

I. About Go and convolutional neural networks

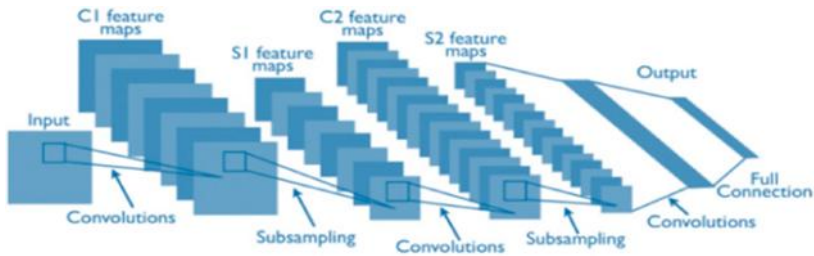
Go is a complex deterministic board game, with complete information, requiring intuition, creativity, and strategic thinking. Computer Go has long been considered a challenge for artificial intelligence.

Approaches that have shown excellent results in other board games have not yielded significant results here. This is partly due to the difficulty of creating a position estimation function, partly due to the very depth and width of the tree of possible moves (if in the chess tree the width is ≈ 35 , and the depth is ≈ 80 , then in Go the width is ≈ 250 , and the depth is ≈ 150).

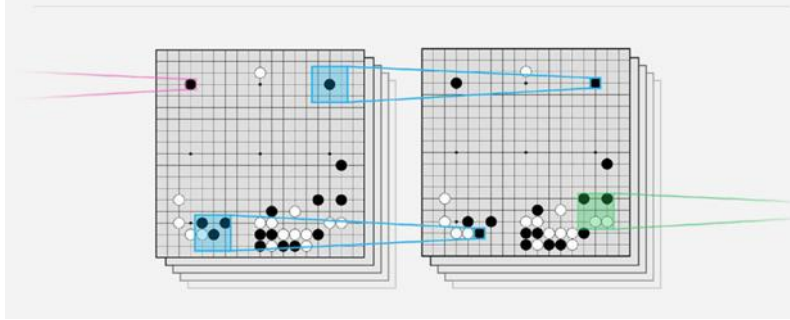
- Briefly about the rules: Stones are placed on a 19 x 19 board (9 x 9, 13 x 13).
- Stones are removed when they are surrounded.
- The one who occupies the most territory (plus the stones eaten) wins.



Recently, deep convolutional neural networks have achieved unsurpassed results in areas such as: image classification, face recognition, Atari-like games. Convolutional networks use multiple layers of neurons arranged in overlapping tiles. This is done to create increasingly abstract, localized representations of an image. The authors use a similar neural network architecture in the AlphaGo algorithm. The network takes a position on the board as a 19×19 image and uses convolutional layers to interpret the position. This approach has achieved unprecedented performance in the task of estimating the position and predicting the.



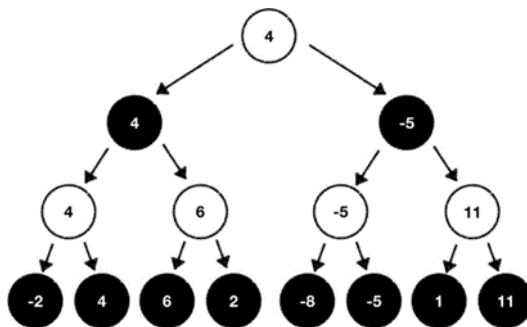
Convolutional neural network



II. Computer Go before AlphaGo

1. Variations of tree search

One of the traditional methods is minimax tree search. It consists of modeling all hypothetical moves on the board up to a certain point, and then using an evaluation function to select the move that leads to its maximization. The process is repeated every move. While tree search was very effective in chess, it was much less effective in Go. This happens because of the nature of the game. It is difficult to create an effective board evaluation function, and also because there are numerous of reasonable moves that result in a large branching factor in the tree.



2. Playing according to preset patterns and rules

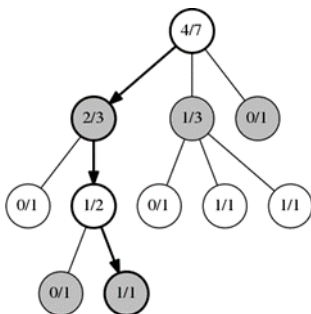
Beginners often learn from game recordings of old games played by experienced players. There are also many books describing the basic

principles of Go. Early works often included attempts to teach AI heuristics of human-style Go knowledge.

Two ways were suggested: learn common stone configurations and positions, and concentrate on local battles. However, as was seen later, this approach lacks both quality and quantity of knowledge.

3. Monte Carlo tree search (MCTS)

One of the main alternatives to using manually coded patterns is to use Monte Carlo methods. The algorithm creates a list of possible moves and for each move, thousands of games are randomly played on the resulting board. The move leading to the best ratio of random game wins for the current player is declared the best. The advantage of this method is that it requires very little subject matter knowledge or expert input, and the trade-off is increased memory and processor requirements.



4. The use of machine learning

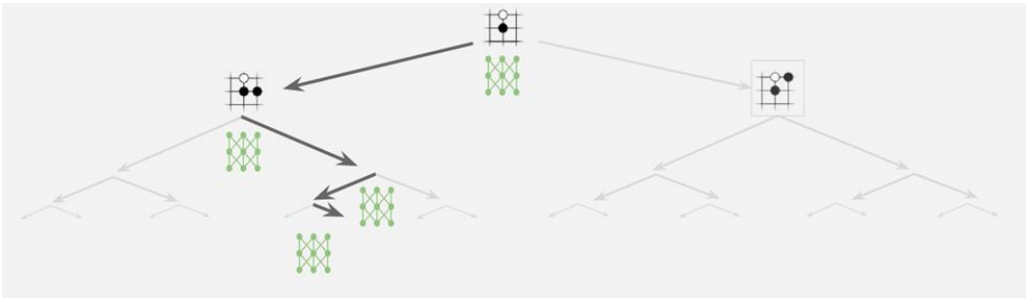
The level of knowledge-based systems is closely tied to the knowledge of their programmers and associated subject matter experts. This limitation makes it difficult to create really strong AIs. Another way is to use machine learning techniques. In them, the only thing programmers need to program are rules and simple algorithms to evaluate how to analyze the value of a position. During the learning process, the machine itself generates its own understanding of patterns, heuristics, and game strategies.

III. How AlphaGo works

AlphaGo is the first computer program to defeat a professional Go player, the first to defeat the World Go Champion, and arguably the strongest Go player in history. AlphaGo algorithm is based on a Monte

Carlo method, which additionally uses convolutional neural networks to optimize the tree width and depth.

1. Shrink the tree in width



Idea:

Train a neural network that will predict the most likely moves of professional players by their position on the board. Then at each moment of branching the tree, we can consider only the most likely moves, which greatly reduces the width of the search.

Implementation:

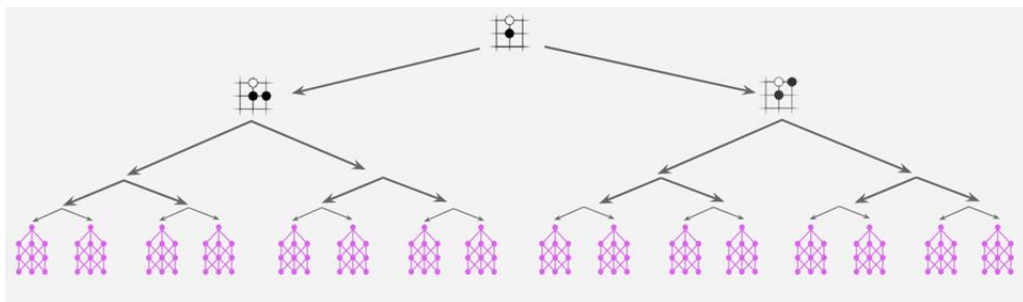
Feature	# of planes	Description
Stone colour	3	Player stone / opponent stone / empty
Ones	1	A constant plane filled with 1
Turns since	8	How many turns since a move was played
Liberties	8	Number of liberties (empty adjacent points)
Capture size	8	How many opponent stones would be captured
Self-atari size	8	How many of own stones would be captured
Liberties after move	8	Number of liberties after this move is played
Ladder capture	1	Whether a move at this point is a successful ladder capture
Ladder escape	1	Whether a move at this point is a successful ladder escape
Sensibleness	1	Whether a move is legal and does not fill its own eyes
Zeros	1	A constant plane filled with 0

30 million games of strong players (KGS 5+ dan) are taken for training. On this data we train a 12-layer convolutional neural network, which the authors call SL-policy network. The input is a tensor of size $19 \times 19 \times 48$. 19×19 is the size of the game board, 48 is the number of features after encoding them using One-Hot Encoding method:

The output is an array of size 19×19 - for each square the probability is predicted that the professional player goes exactly there. It is trained using stochastic gradient descent, maximizing likelihood. The

accuracy of the predictions of this network is 57.0% (the prediction is chosen as the move with the highest probability). Interestingly, if we leave only the "Stone color" trait, the accuracy will be 55.7%. As the authors note, even a small increase in accuracy greatly affects the final strength of the game. The previous accuracy result in this area was 44.4%.

2. Reduce the tree in depth



Idea:

1. Train a neural network that will give the probability of winning this game based on the position given to it.
2. Let's train a very fast model, which will predict the most probable move of a pro.

Now, coming to a new node that we don't know anything about yet, we can quickly estimate the outcome of the game from it using neural network 1, as well as playing the position to the end of the game using fast model 2.

Implementation:

The network architecture is similar to the SL-policy network (the same 12-layer convolutional neural network), except that at the input it receives one more additional attribute - whether the player plays black, and the output is one number - the probability of victory. In the article, this network is called the value network.

If to teach the value network on the same 30 million games as SL, the problem of overlearning appears (many positions belong to a game with the same outcome and the network starts to learn to remember the game).

To solve this problem, the authors use the following strategy: the SL-policy network is copied and called RL-policy network, then RL starts playing a game with a randomly chosen its previous version (the first game is played with itself). At the end of the game weights of RL network are updated so that the network learns to win instead of predicting a professional's move (the gradient sign depends on game result), the updated RL is added to the pool of its versions. This approach results in a neural network that already wins the best Go programs 80% of the time.

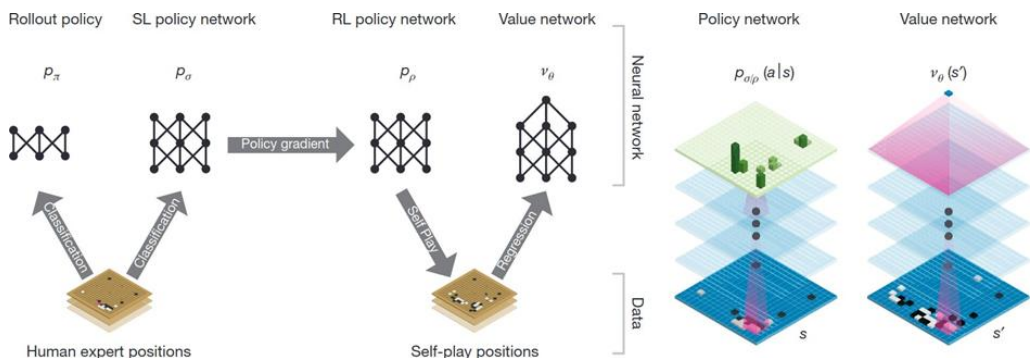
Each game played between RL networks takes a random position and sends it to the training dataset for the value network. So, this whole self-game process is only needed to collect data to train the value network.

2. A very fast model is trained, softmax linear regression, which the authors call fast rollout policy, on the same 30 million batches as SL. This network has a not very good accuracy of 24.2%, but it is 1500 times faster than SL. It uses a lot of preset patterns as features:

Feature	# of patterns	Description
Response	1	Whether move matches one or more response pattern features
Save atari	1	Move saves stone(s) from capture
Neighbour	8	Move is 8-connected to previous move
Nakade	8192	Move matches a <i>nakade</i> pattern at captured stone
Response pattern	32207	Move matches 12-point diamond pattern near previous move
Non-response pattern	69338	Move matches 3×3 pattern around move

As a bottom line, one can estimate a random position as follows: take the value network estimate with a weight of λ , and play that position to the end of the game using fast rollout policy, and take the bottom line with a weight of $(1 - \lambda)$.

3. General scheme of the described neural networks



To summarize, we have:

- Fast rollout policy (on the picture Rollout policy) is a very fast model that predicts the professional's move in a given position. It is used to play the position to the end of the game to find out the result.

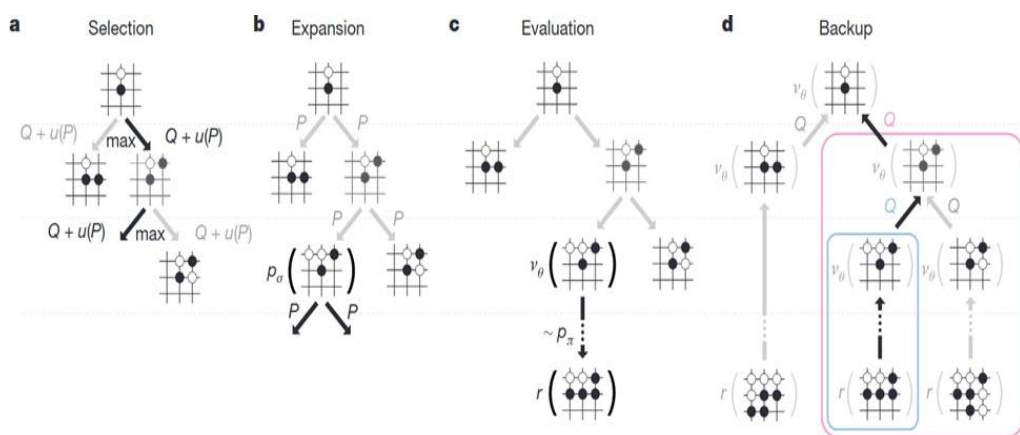
- SL policy network - says on this position with what probability each

Possible move will be made by the professional player. It is needed for choosing the direction of descent in a tree.

- RL policy network - needed only to prepare training data for the value network. It is obtained from the SL policy network by playing with itself.

- Value network - by position tells us what chances we have to win the game.

4. Tree search



Tree of positions is built, with the current one at the root. For each vertex the value Q is stored - the average of all the results of parties that are reachable from that vertex - essentially how much the given

position leads to victory. This tree goes through many iterations of the following algorithm:

At each iteration, we go from the root and go down to where more $Q + u(P)$. $u(P)$ is a special additive that encourages exploration of new paths in the tree. $u(P)$ is the greater the probability of that move in the SL-policy network prediction, and the smaller the more often we have walked through that node. When we reach a new leaf, it is evaluated through the value network and fast rollout policy. The resulting score updates all nodes on the path from leaf to root.

As a result, the best move is the node which was used most often (this is more stable than choosing the maximum by Q).

5. Result

Algorithm iterations during tree search are run in parallel, on multiple graphics cards (176 GPUs in AlphaGo, which played at the championship). As a result, we have a top-1 Go algorithm - AlphaGo which has beaten all top Go programs with the score 494 out of 495 games, and has also beaten multiple European champion Fan Hui (professional 2 dan) with the score 5:0.

Conclusion

The authors have succeeded for the first time in developing a program for playing Go, AlphaGo, based on a combination of deep neural networks and tree search, which plays at the level of the strongest human players. It was also the first to develop effective move selection and position estimation functions in Go, based on deep neural networks.

A further line of work was AlphaGo Master and AlphaGo Zero, which cemented the computer's breakthrough in the game of Go by beating professional players (including top-1 Lee Sedol) 60-0.

References / Список литературы

1. *Silver David et al.* Mastering the game of Go with deep neural networks and tree search. [Electronic Resource]. URL: <https://doi.org/10.1038/nature16961/> (date of access: 07.11.2022).

**XXVI INTERNATIONAL CORRESPONDENCE SCIENTIFIC SPECIALIZED CONFERENCE
INTERNATIONAL SCIENTIFIC REVIEW
OF THE TECHNICAL SCIENCES,
MATHEMATICS AND COMPUTER SCIENCES
Boston. USA. December 22-23 2022
[HTTPS://SCIENTIFIC-CONFERENCE.COM](https://scientific-conference.com)**



**COLLECTION OF SCIENTIFIC ARTICLES
PUBLISHED BY ARRANGEMENT WITH THE AUTHORS**



You are free to:

Share – copy and redistribute the material in any medium or format

Adapt – remix, transform, and build upon the material
for any purpose, even commercially.

Under the following terms:

Attribution – You must give appropriate credit,
provide a link to the license, and indicate if changes were made.

**You may do so in any reasonable manner,
but not in any way that suggests the licensor endorses you or your use.**
ShareAlike – If you remix, transform, or build upon the material, you must
distribute your contributions under the same license as the original.

**ISBN 978-1-64655-135-4
INTERNATIONAL CONFERENCE**

PRINTED IN THE UNITED STATES OF AMERICA