

## РАЗРАБОТКА СИСТЕМЫ РАННЕГО ВЫЯВЛЕНИЯ РАКА ЛЕГКИХ НА ОСНОВЕ АНАЛИЗА МЕДИЦИНСКИХ ИЗОБРАЖЕНИЙ С ИСПОЛЬЗОВАНИЕМ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

**Муродиллоева Зулфинисо Ҳаким қизи**

Ташкентского университета информационных  
технологий имени Мухаммад ал- Хоразмий Ассистент

E-mail: [zulfinisozulfiniso@icloud.com](mailto:zulfinisozulfiniso@icloud.com)

### АННОТАЦИЯ

В работе решается задача распознавания онкологических заболеваний по медицинским снимкам томографии легких. Задача сводится к построению и обучению бинарного классификатора.

Для решения задачи предлагается использовать современные архитектуры сверточных нейронных сетей: ResNet50 и Xception. Результатом работы являются обученные нейронные сети указанных архитектур и эксперименты с их гиперпараметрами (метод обучения, скорость обучения, количество эпох)

**Ключевые слова:** Нейронная сеть, алгоритм обучения, перцептрон, Сверточные нейронные сеть, Мультиспиральная компьютерная томография (МСКТ), Tensorflow и Keras.

## DEVELOPMENT OF A SYSTEM FOR EARLY DETECTION OF LUNG CANCER BASED ON THE ANALYSIS OF MEDICAL IMAGES USING ARTIFICIAL INTELLIGENCE

### ABSTRACT

The paper solves the problem of recognizing oncological diseases based on medical images of lung tomography. The task is to build and train a binary classifier.

To solve the problem, it is proposed to use modern convolutional neural network architectures: ResNet50 and Xception. The result of the work are trained neural networks of these architectures and experiments with their hyperparameters (training method, learning rate, number of epochs)

**Keywords:** Neural network, learning algorithm, perceptron, Convolutional neural network (CNN), Multispiral computed tomography (MSCT), Tensorflow and Keras.

**Нейронная сеть** [1] (также искусственная нейронная сеть, ИНС) — математическая модель, а также её программное или аппаратное

воплощение, построенная по принципу организации и функционирования биологических нейронных сетей. На рис.1. представлена схема биологического нейрона.

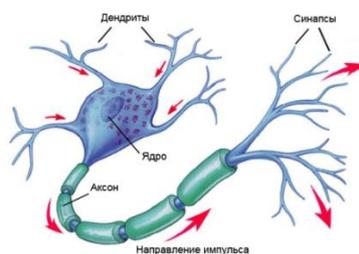


Рис.1 – Биологический нейрон

Обычно нейронная сеть должна настроить свои веса связей по имеющейся обучающей выборке [5], причем функционирование сети улучшается по мере итеративной настройки весовых коэффициентов.

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

Существуют три парадигмы обучения нейросетей [5]:

- С учителем
  - Перцептрон
  - Сверточные нейронные сети
- Без учителя
  - Сети адаптивного резонанса
  - Смешанная
    - Сеть радиально-базисных функций

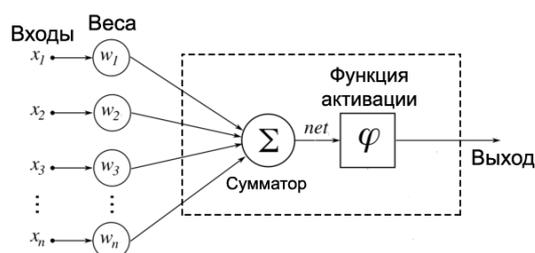


Рис.2 – Искусственный нейрон

Соотнесем известное нам понятие о биологическом нейроне с понятиями, характерными для перцептрона. Данные приведены в следующей таблице:

Таблица 1 – Сравнительная характеристика понятий БНС и ИНС

Биологическая нейронная сеть	Искусственная нейронная сеть
Ядро	Узел
Дендриты	Входные данные
Синапсы	Веса
Аксон	Выходные данные

### Однослойный перцептрон

Один нейрон может выполнять простейшие вычисления, но основные функции нейросети обеспечиваются не отдельными нейронами, а соединениями между ними. Однослойный перцептрон<sup>1</sup> представляет собой простейшую сеть, которая состоит из группы нейронов, образующих слой, как показано на рис.3. Входные данные кодируются вектором значений

$X = (x_1, x_2, \dots, x_k)$ , каждый элемент  $x_i$  подается на соответствующий вход каждого нейрона в слое. В свою очередь, нейроны вычисляют выход независимо друг от друга. Очевидно, что размерность выхода (то есть количество элементов) равна количеству нейронов, а количество синапсов у всех нейронов должно быть одинаково и совпадать с размерностью входного сигнала.

однослойный перцептрон Ф.Розенблатта

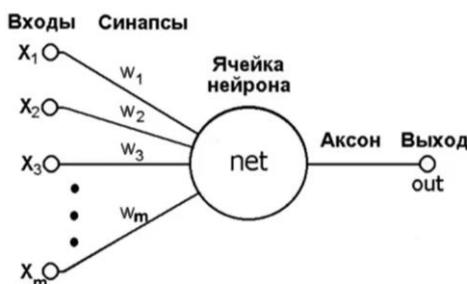


Рис. 3. Однослойный перцептрон

Несмотря на кажущуюся простоту, однослойный перцептрон может выполнять ряд полезных задач, например, классифицировать образы или вычислять значения логических функций [7].

### Многослойный перцептрон

Многослойный перцептрон способен рассчитать выходное значение  $Y$  для входного значения  $X$ . Другими словами, сеть вычисляет значение некоторой векторной функции:  $Y = F(X)$ . Таким образом, условие задачи, которая ставится перцептрону, должно быть сформулировано в виде множества векторов  $\{x^1, \dots, x^s\}$ . Решение задачи будет представлено в виде векторов  $\{y^1, \dots, y^s\}$ , причем для  $\forall s \quad y^s = F(x^s)$ .

Все, что способен сделать перцептрон, - сформировать отображение  $F: X \rightarrow Y$  для  $\forall x \in X$ . Мы не можем «извлечь» из перцептрона данное отображение полностью, а можем только посчитать образы произвольного числа

<sup>1</sup> От англ. «perceptron». Помимо термина «перцептрон» в русскоязычной литературе часто используется термин «персептрон».

точек.

Задача формализации, то есть выбора смысла, которым наделяются компоненты входного и выходного векторов, решается человеком на основе практического опыта. К сожалению, жестких рецептов формализации для нейронных сетей пока не создано.

Чтобы построить многослойный перцептрон, необходимо выбрать его параметры по следующему алгоритму [6]:

- Определить, какой смысл вкладывается в компоненты входного вектора  $X$ . Входной вектор должен содержать формализованное условие задачи, то есть всю информацию, необходимую для того, чтобы получить ответ.

- Выбрать выходной вектор  $Y$  таким образом, чтобы его компоненты содержали полный ответ для поставленной задачи.

- Выбрать вид функции активации нейронов. При этом желательно учесть специфику задачи, так как удачный выбор увеличит скорость обучения.

- Выбрать количество слоев и нейронов в слое.

- Задать диапазон изменения входов, выходов, весов и пороговых уровней на основе выбранной функции активации.

- Присвоить начальные значения весам и пороговым уровням. Начальные значения не должны быть большими, чтобы нейроны не оказались в насыщении (на горизонтальном участке функции активации), иначе обучение будет очень медленным. Начальные значения не должны быть и слишком малыми, чтобы выходы большей части нейронов не были равны нулю, иначе обучение тоже замедлится.

- Провести обучение, то есть подобрать параметры сети так, чтобы задача решалась наилучшим образом. По окончании обучения сеть сможет решать задачи того типа, которым она обучена.

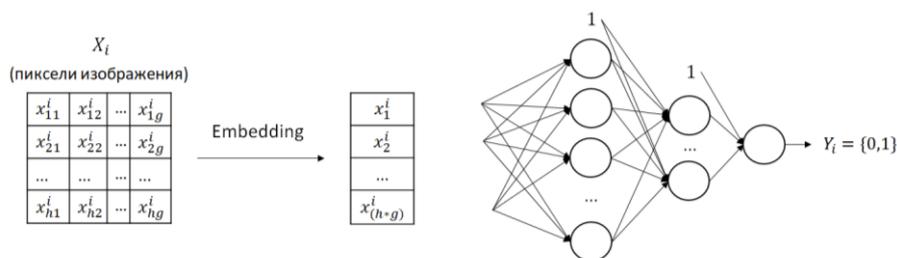
- Подать на вход сети условия задачи в виде вектора  $X$ . Рассчитать выходной вектор  $Y$ , который и даст формализованное решение задачи.

**Свёрточная нейронная сеть** (англ. convolutional neural network, CNN) — специальная архитектура искусственных нейронных сетей, предложенная **Яном Лекуном** в 1988 году [8] и нацеленная на эффективное распознавание образов [9], входит в состав технологий глубокого обучения (англ. deep learning). Использует некоторые особенности зрительной коры [10], в которой были открыты так называемые простые клетки, реагирующие на прямые линии под разными углами, и сложные клетки, реакция которых связана с активацией определённого набора простых клеток. Таким образом, идея свёрточных нейронных сетей заключается в чередовании свёрточных слоёв

(англ. convolution layers) и субдискретизирующих слоёв (англ. subsampling layers или англ. pooling layers, слоёв подвыборки). Структура сети — однонаправленная (без обратных связей), принципиально многослойная. Для обучения используются стандартные методы, чаще всего метод обратного распространения ошибки. Функция активации нейронов (передаточная функция) — любая, по выбору исследователя.

Название архитектура сети получила из-за наличия операции свёртки, суть которой в том, что каждый фрагмент изображения умножается на матрицу (ядро) свёртки поэлементно, а результат суммируется и записывается в аналогичную позицию выходного изображения.

Рассмотрим задачу распознавания объектов на изображении. Имеем множество изображений  $X_1, X_2, \dots, X_n$  и ответы  $Y_i = \{0,1\}$ ,  $i = \overline{1, n}$  о наличии некоторого объекта на изображении. Каждый пиксель  $x_{hg}^i$  принимает числовое значение, характеризующее цвет



Возникают следующие вопросы:

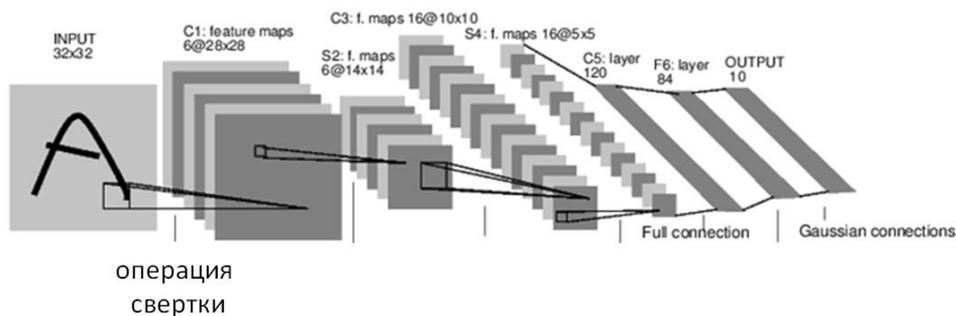
- Как формируются входные сигналы сети?
- Какого размера изображения нужно использовать, чтобы сеть была применима в реальной жизни?
- Если размер изображения большой, то будет очень много входных сигналов. Тогда при полно-связной сети будет очень много связей. Не будет ли сеть слишком сложной?

Пусть размер изображения:  $h = 1024$ ,  $g = 768$ .

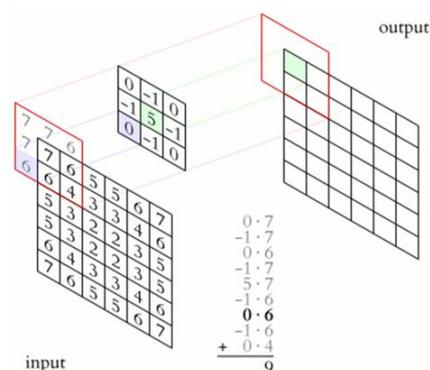
Тогда размер входного вектора: **1024 x 768 пикселей = 786 432 сигналов.**

Тогда при 100 нейронах первого скрытого слоя имеем **78 643 200 связей.**

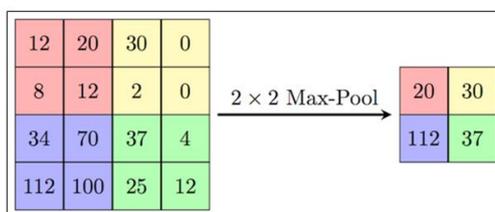
- При обработке изображений в полно-связной нейронной сети необходимо содержать огромное количество связей
  - Вероятнее всего нейронная сеть становится слишком сложной для решаемой задачи
  - Следовательно вероятность переобучения крайне высока
- Сверточная нейронная сеть (CNN - Convolutional Neural Network)



- на вход подается двумерный массив пикселей без преобразования в вектор
- вместо взвешенной суммы всех сигналов используется операция свертки (Conv):
  - в качестве весов используется ядро (массив весов определенного размера)
  - на вход подаются не все сигналы (пиксели), а выбранные скользящем окне ядра
- вместо нейронов скрытого слоя формируются карты сверточного слоя



После свертки используется операция субдискритизации (например: MaxPooling)



Вместо субдискритизации можно использовать слой с функцией активации (например, ReLU). Можно чередовать слои субдискритизации и слои с функцией активации.

Input -> Conv -> ReLU -> Conv -> ReLU -> Pool -> ReLU -> Conv -> ReLU -> Pool -> Fully Connected

В качестве выходного слоя сети используется полно-связная сеть.

Пусть  $Y_i = \{0, 1, 2, \dots, r\}$ . То есть необходимо распознать не 1 объект на изображении, а определить, какой из  $r$  объектов представлен на изображении.

Тогда учитель имеет вид

$$Y_i = (p_i(Y_i = 0), p_i(Y_i = 1), p_i(Y_i = 2), \dots, p_i(Y_i = r))$$

где  $p_i(Y_i = j)$  вероятность, что на изображении представлен объект  $j$

Соответственно нейронная сеть имеет  $r$  выходных нейронов.

**Замечание:** вероятность наличия объекта на изображении определяется не через априорную вероятность по всем изображениям в выборке. Вероятности распределяются отдельно для каждого изображения при разметке данных.

Мультиспиральная компьютерная томография (МСКТ) является основной методикой исследования томографии в медицине. Изображение органов и тканей МСКТ позволяет осуществлять диагностику заболеваний и повреждений внутренних органов и систем человека.

### Описание задачи

Рассматривается база данных снимков компьютерной томографии легких (The IQ OTHNCCD lung cancer). База данных содержит изображения размера 512x512 в оттенках серого. С помощью четырех различных моделей нейронных сетей предсказывалась принадлежность изображений одному из двух классов: здоровый пациент, больной пациент.

Модели будут сравниваться по качеству предсказания на тестовых данных после обучения. Планируется в результате определить стадию болезни, и сравнить производительность всех моделей в зависимости от изменения параметров при обучении нейронной сети.

Рассмотрим снимки здоровых и больных с заболеванием рака лёгких.

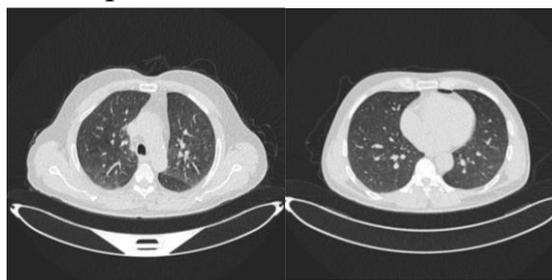


Рис. Снимки людей со здоровыми лёгкими.

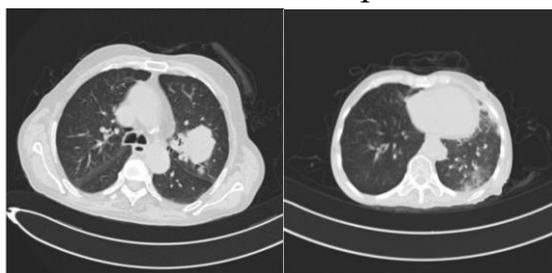


Рис. Снимки людей больных с раком лёгких.

Рассмотрим структуру датасета “IQ OTHNCCD lung cancer”. IQ OTHNCCD lung cancer- это датасет, состоящий из снимков людей со здоровыми лёгкими и больных с раком лёгких. Он включает в себя 977 наблюдений, полученных от 100 пациентов. Таким образом, датасет IQ OTHNCCD lung cancer содержит снимки внутренних органов, т.е лёгких. Каждое наблюдение было проведено аппаратурой отмечено как "0" – здоровый или "1" – больной.

Аугментация - это методика увеличения обучающей выборки из уже имеющихся данных. Существует множество разновидностей аугментации, но чаще всего на практике применяют различные отражения, как по горизонтали (horizontal flip), диапазон сдвига (shear range), приближение (zoom range) и др. В данной работе аугментация реализована с помощью функций из библиотеки keras.

#### # Аугментация

```
train_datagen=ImageDataGenerator(  
    preprocessing_function=keras.applications.resnet50.preprocess_input,  
    shear_range=0.2,  
    zoom_range=0.2,  
    horizontal_flip=True)  
test_datagen=ImageDataGenerator(  
    preprocessing_function=keras.applications.resnet50.preprocess_input,)
```

Обучение проводилось в сервисе Google Colaboratory с сохранением весов предобученной модели. Данные были разделены на две части: тренировочные (train data) и тестовые (test data).

#### Решение

Использовались 2 типа архитектур нейронных сетей:

- Нейронная сеть ResNet-50
- Нейронная сеть Xception

#### Результаты и эксперименты

В процессе работы был проведен ряд экспериментов, включающих в себя изменение таких важных параметров, как: количество эпох (epochs), применение различных оптимизаторов (optimizer) и изменения скорости обучения (lr). Оценивалась точность предсказания обученной модели на тестовых данных – Loss, Accuracy, Val\_loss и Val\_accuracy.

Model	Loss	Accuracy	Val_loss	Val_accuracy	Optimizer/lrate	Epochs
ResNet50	0.1228	0.9800	0.1736	0.9187	sgd/0.0001	5
ResNet50	0.0734	0.9804	0.1390	0.9187	sgd/0.0001	10
ResNet50	0.0160	0.9973	0.0560	0.9688	sgd/0.001	15
ResNet50	0.0089	1.0000	0.0599	0.9688	sgd/0.001	20
Xception	0.9025	0.9432	0.6415	0.9438	sgd/0.0001	5
Xception	0.5342	0.9704	1.3273	0.9125	sgd/0.0001	10
Xception	0.1430	0.9808	0.5191	0.9500	sgd/0.00001	15
Xception	0.1629	0.9817	0.1165	0.9688	sgd/0.00001	20

Для двух архитектур нейронных сетей (ResNet50 и Xception) был получен наилучший результат валидации (Val\_accuracy) при количестве 20 эпох.

Легко заметить, что в случае, когда Xception обучалась 10 эпох со скоростью обучения (lrate) равной 0.0001 было достигнуто переобучение нейронной сети. Это видно, так как при аналогичной скорости обучения на 5-й эпохе была получена значительно более высокая точность. Следовательно, необходимо понижать скорость обучения.

Поскольку архитектура Xception является более сложной, то для достижения результата, совпадающего с ResNet50 (Val\_accuracy = 0.9688) потребовалось использовать более низкую скорость обучения.

#### **Вывод:**

- Для классификации изображений были применены 2 модели сверточных нейронных сетей с использованием библиотек Tensorflow и Keras.
- Была произведена предобработка данных
- Были проведены эксперименты с выбранными нейронными сетями
- По результатам экспериментов были сделаны выводы и была достигнута точность тестирования Val\_accuracy = 0.9688.

#### **СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ: (REFERENCES)**

- [1]. Нейронная сеть // Большая российская энциклопедия : [в 35 т.] / гл. ред. Ю. С. Осипов. — М. : Большая российская энциклопедия, 2004—2017.
- [2]. Mc Culloch W.C. H. A Logical Calculus of Ideas Immanent in Nervous Activity. / W. C. Mc Culloch, W. H. Pitts. // Bulletin of Mathematical Biophysics. – 1943/ - Vol. 5. – P. 115-119.
- [3]. Розенблатт Ф. Принципы нейродинамики: Перцептроны и теория механизмов мозга. / Ф. Розенблатт. – М.: Мир, 1965. —480 с.
- [4]. Москалев Н. С. Виды архитектур нейронных сетей // Молодой ученый. — 2016. — №29. — С. 30-34. — URL <https://moluch.ru/archive/133/37121/> (дата обращения: 17.12.2018).

- [5]. Галушкин А. И. Теория нейронных сетей. — М.: Издательское предприятие редакции журнала «Радиотехника», 2000.
- [6]. Smith L. An Introduction to Neural Networks. Unpublished draft, University of Stirling, 2001. <http://www.cs.stir.ac.uk/~lss/NNIntro/InvSlides.html>
- [7]. Короткий С. Нейронные сети: основные положения. <http://www.gotai.net/documents/doc-nn-002.aspx>.
- [8]. Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard and L. D. Jackel: Backpropagation Applied to Handwritten Zip Code Recognition, Neural Computation, 1(4):541-551, Winter 1989.
- [9]. Convolutional Neural Networks (LeNet) - DeepLearning 0.1 documentation. DeepLearning 0.1. LISA Lab. Дата обращения: 31 августа 2013.
- [10]. Matusugu, Masakazu; Katsuhiko Mori; Yusuke Mitari; Yuji Kaneda. Subject independent facial expression recognition with robust face detection using a convolutional neural network (АНГЛ.) // Neural Networks : journal. — 2003. — Vol. 16, no. 5. — P. 555—559. — doi:10.1016/S0893-6080(03)00115-1.