

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий

(наименование института полностью)

Кафедра «Прикладная математика и информатика»

(наименование)

02.03.03 Математическое обеспечение и администрирование информационных систем

(код и наименование направления подготовки, специальности)

Технология программирования

(направленность (профиль) / специализация)

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
(БАКАЛАВРСКАЯ РАБОТА)**

на тему «Поиск аномалий на рентгеновских снимках с помощью сверточных нейронных сетей»

Студент

Л.И. Смирнов

(И.О. Фамилия)

(личная подпись)

Руководитель

М.Г. Лисовская

(ученая степень, звание, И.О. Фамилия)

Консультант

О.А. Головач

(ученая степень, звание, И.О. Фамилия)

## **Аннотация**

Тема: «Поиск аномалий на рентгеновских снимках с помощью сверточных нейронных сетей».

**Целью ВКР** является реализация нейронной сети для распознавания патологий грудной клетки.

**Объектом ВКР** является диагностика патологий грудной клетки.

**Предмет исследования** – нейросетевые алгоритмы распознавания и классификации патологий грудной клетки.

В данной выпускной квалификационной работе исследуются патологии грудной клетки, а также различные методы их диагностики.

В работе представлены результаты тестирования и внедрения нейронной сети, обученной на подготовленном наборе данных.

Структура ВКР состоит из введения, трех глав, заключения и списка литературы.

Во введении описывается то, насколько актуально проводимое исследование, а также формулируется цель и ставятся задачи.

В первой главе происходит изучение патологий грудной клетки и архитектур нейронных сетей.

Во второй главе рассматривается процесс разработки нейронной сети.

В третьей главе рассматривается процесс тестирования и внедрения нейронной сети.

Бакалаврская работа состоит из пояснительной записки, включая 33 рисунков и 10 источников на иностранном языке.

## Abstract

The title of the graduation work is “Search for anomalies in x-ray images using convolutional neural networks”.

The goal of the graduation work is an implementation of a neural network for recognizing chest pathologies

The object of the graduation work is the diagnosis of chest pathologies.

The subject of the study is neural network algorithms for recognizing and classifying pathologies of the chest.

In this graduation work, chest pathologies and methods for their diagnosis are studied.

This paper presents the results of testing and implementing a neural network trained on a prepared data set.

The structure of the graduation work consists of an introduction, three chapters, a conclusion, a list of literature.

The introduction describes the relevance of the ongoing research, formulates the goal and sets tasks that need to be addressed.

The first part provides an overview of neural network architectures. The second chapter deals with the process of developing neural network and preparing initial data.

The third chapter examines the process of testing and implementing neural network.

The graduation work consists of an explanatory note, including 33 drawings and 10 sources in a foreign language.

## Оглавление

Введение.....	5
ГЛАВА 1 ИЗУЧЕНИЕ ПАТОЛОГИЙ ГРУДНОЙ КЛЕТКИ И АРХИТЕКТУР НЕЙРОННЫХ СЕТЕЙ .....	6
1.1 Изучение патологий грудной клетки .....	6
1.1.1 Методы диагностики патологий.....	6
1.1.2 Основные легочные патологии.....	11
1.2 Изучение архитектур нейронных сетей.....	13
1.2.1 Определение функционала нейронной сети .....	13
1.2.2 Анализ нейронных сетей .....	13
1.2.3 Описание функций активации .....	17
1.2.4 Описание архитектур нейронных сетей.....	20
1.2.5 Анализ работы сверточной нейронной сети.....	27
ГЛАВА 2 РАЗРАБОТКА НЕЙРОННОЙ СЕТИ .....	33
2.1 Обоснование выбора технических средств .....	33
2.2 Архитектура нейронной сети.....	35
2.3 Подготовка набора данных для обучения .....	37
2.4 Обучение нейронной сети .....	37
ГЛАВА 3 ТЕСТИРОВАНИЕ И ВНЕДРЕНИЕ НЕЙРОННОЙ СЕТИ .....	42
3.1 Тестирование сети.....	42
3.2 Внедрение сети .....	44
Заключение .....	47
Список используемой литературы .....	49
Приложение А Режим доступа к программному обеспечению, которое используется для обучения нейронной сети .....	51

## **Введение**

На сегодняшний день, анализ рентгенограмм, востребован во множестве медицинских учреждений, так как он позволяет поставить достаточно точный диагноз, а также, может наглядно показать динамику развития процесса лечения. Также, при помощи автоматизации процесса обнаружения аномалий на снимках, существенная часть работы была снята со специалистов данной области, а именно - рентгенологов. Б

По оценке специалистов, врач-рентгенолог за день может изучить не более чем 200 рентгенограмм [2]. Это означает, что в густонаселённых районах в эффективность врач-рентгенолога падает, так как ему приходится столкнуться с большим количеством снимков. Из-за этого, снижается качество обработки информации, содержащейся в рентгенограмме.

Одним из решений данной задачи обнаружения аномалий на рентгеновских снимках является применение сверточных нейронных сетей.

**Цель ВКР:** реализация нейронной сети для распознавания патологий грудной клетки.

**Объект ВКР:** диагностика патологий грудной клетки.

**Предмет ВКР:** нейросетевые алгоритмы распознавания патологий грудной клетки.

**Основные задачи ВКР:**

- сделать обзор предметной области;
- выбрать и подготовить набор данных для обучения;
- выбрать архитектуру нейронной сети и реализовать её;
- проанализировать результаты реализации нейронной сети.

# **ГЛАВА 1 ИЗУЧЕНИЕ ПАТОЛОГИЙ ГРУДНОЙ КЛЕТКИ И АРХИТЕКТУР НЕЙРОННЫХ СЕТЕЙ**

## **1.1 Изучение патологий грудной клетки**

### **1.1.1 Методы диагностики патологий**

Благополучный исход заболевания во много зависит от того, насколько быстро было начато лечение. Методы диагностики грудной клетки можно разделить на общеклинические и лабораторные [2].

Методы клинической диагностики предполагают осмотр пациента врачом. Перед началом данного осмотра, врач собирает различную информацию о состоянии пациента. Врач выясняет, на что именно жалуется пациент, спрашивает о времени появления первых симптомов, а также изучает медицинскую карту больного для получения более точной информации о перенесённых и хронических заболеваниях.

К способам общего обследования грудной клетки относят:

- осмотр больного;
- пальпацию;
- перкуссию;
- аускультацию.

При проведении осмотра грудной клетки врач определяет её величину, форму, симметричность, степень участия в дыхательных движениях её обеих половин, частоту, глубину и тип дыхания, соотношение и продолжительность вдоха и выдоха, участие вспомогательной мускулатуры в процессе дыхания.

Во время проведения пальпации, врач выясняет состояние подкожной клетчатки, а также возможные повреждения ребер и болезненные участки пациента. После этого, проводится проверка на голосовое дрожание. Во время данной проверки, пациента просят произнести несколько определённых фраз. В это же время. Врач проверяет симметричность вибраций за грудиной.

Метод перкуссии основан на простукивании различных органов грудной клетки, после чего, у тканей появляются колебательные движения. По характеру такого звука, врач способен определить плотность органов, их воздушность, эластичность и объём. Также, перкуссию можно проводить двумя методами: посредственным и непосредственным.

В свою очередь, метод Он основан на прослушивании различных физиологических звуков органов грудной клетки во время дыхания. При выполнении данного метода используют стетоскоп или фонендоскоп.

Далее, методы лучевой диагностики. Данная диагностика является частью комплексного обследования грудной клетки. В начале, делают обзорные рентгеновский снимок, после чего, проводят дальнейшие исследования, если в этом есть необходимость [1].

Методы лучевой диагностики включают в себя:

- компьютерную томографию;
- магнитно-резонансную томографию;
- УЗИ;
- контрастные методы исследования;
- радионуклидные обследования;
- рентгеноскопию;
- флюорографию;
- рентгенографию.

Компьютерная томография и магнитно-резонансная томография имеют широкое применение благодаря относительной безопасности применения, а также высокому качеству получения снимков.

Результатом выполнения компьютерной томографии является серия послойных снимков органов грудной клетки. Серию снимков анализирует компьютер, после чего, они выводятся на экран монитора.

Метод магнитно-резонансной томографии основан на том, что ткани организма могут излучать электромагнитное поле, когда находятся под

воздействием радиочастотных импульсов. При помощи компьютера, сигналы преобразуют в высококачественные изображения срезов органов.

К контрастным методам лучевой диагностики относят:

- ангиографию;
- пневмомедиастинографию;
- плеврографию;
- фистулографию.

При проведении ангиографии происходит обследование малого круга кровообращения. В организм пациента вводится водорастворимый йодосодержащий препарат, после чего, делают серию снимков, на которых происходит определение артериальной фазы кровотока, а после - венозной. Такая методика позволяет определить, наличие в организме тромбов, аневризм, сужений или же анатомических нарушений строения сосудов.

Пневмомедиастинография применяется в онкологической практике для определения точной локализации опухолей [1].

В свою очередь, при плеврографии контрастный препарат через дренаж вводится непосредственно в плевральную полость.

Фистулографию проводят при наружных свищах грудной клетки. Это необходимо для определения их вида, размера и обнаружения источника гнойного процесса.

В каком-то смысле, радионуклидное исследование органов, расположенных в грудной клетке, похоже на обследование с использованием контраста. Суть этого метода заключается во введении в организм пациента радиоактивных изотопов. В начале, их вдыхают в составе смеси газов, после чего, их вводят внутривенно. В это время, на УЗИ происходит отслеживание распределения изотопов. Данное исследование проводится в целях оценки эффективности проведенного ранее лечения злокачественных новообразований в легких [].



Ещё один, распространённый метод диагностики это – флюорография.

Данный метод заключается в фотографировании органов, при котором происходит проекция данных на экран рентгеновской плёнки. Обычно, такая плёнка имеет размеры 70 или 100 мм, что является стандартом для такого рода исследований.

Для выполнения флюорографии, в медицинском учреждении должен находиться рентгенофлюорографический аппарат.

Преимуществом данного метода является высокая эффективность при необходимости обнаружения таких заболеваний, как:

- туберкулёз;
- патологии бронхолёгочной системы;
- различные онкологические заболевания.

Флюорография является относительно менее вредным методом и поэтому используется при массовом обследовании населения.

Однако, у флюорографии присутствуют существенные недостатки при диагностике некоторых патологий:

- снимок не отличается такой резкостью и контрастностью, как при проведении рентгена;
- сложнее различить патологии лёгких;
- уменьшены размеры грудной клетки на изображении.

Приведённых выше недостатков достаточно для того, чтобы назначать данное обследование только в случае необходимости, в других же случаях, будет проведена флюорография.

Рентгенография является одним из основных методов рентгенологического исследования [1]. Суть этого метода состоит в получении изображения с помощью рентгеновских лучей. Данный метод является сравнительно точным и позволяет уловить малейшие изменения.

Для получения более точного исследования снимки производят в двух проекциях.

Преимуществами метода являются:

- возможность рассмотреть детали изображения;
- меньшая лучевая нагрузка по сравнению с рентгеноскопией.

Недостатки метода:

- большие затраты времени для обработки изображений по сравнению с флюорографией;
- невозможность проводить исследование в разных проекциях без использования большой лучевой нагрузки.

Однако, чтобы сделать рентген применяется доза в 0,1-0,3 мЗВ, а при флюорографии 0,03-0,05 мЗВ, что в десять раз меньше, чем при рентгене.

Из-за достаточно большого влияния рентгена на человека, данное обследование происходит только при определённых показаниях.

При подозрении на лёгочную патологию врач назначит рентген, если у больного есть жалобы на: кашель, продолжающийся не менее недели, выделения мокроты, боль за грудиной, хрипы в лёгких, одышку, харканье кровью, повышенную температуру и жар.

Однако, даже если у пациента определены все вышеописанные признаки, это лишь указывает на появление лёгочных проблем у пациента. Для того, чтобы поставить уверенный диагноз, необходимо проводить рентген.

Среди рассмотренных выше методов наиболее подходящими являются методы рентгенографии и флюорографии, так как они наиболее распространены и обладают достаточной информативностью. Далее будут рассмотрены основные патологии, которые могут быть обнаружены на изображениях.

### 1.1.2 Основные легочные патологии

Отличительной особенностью легочных заболеваний является сложность их диагностики, так как каждая причина отдельной патологии носит сугубо субъективный характер.

Существуют некоторые общие факторы, провоцирующие возникновение и развитие процессов воспаления:

- неблагоприятная экологическая обстановка в регионе;
- наличие различных вредных производств;
- нарушение электролитного баланса в организме;
- обострение других хронических заболеваний;
- наличие различных вредных привычек;
- вирусная, бактериальная или грибковая инфекция;
- генетическая предрасположенность.

Среди основных патологий обычно выделяют: пневмонию, ателектаз, пневмоторакс, эмфизему, инфильтрацию, фиброз, гидроторакс.

Для их обнаружения необходимо выявлять признаки, соответствующие данным патологиям. Такие признаки выражены в виде аномалий на рентгеновских снимках. Следовательно, определив эти признаки, можно предположить, присутствует на снимке та или иная патология.

Далее будет подробнее рассмотрена пневмония.

Пневмонией называют воспаление всего или отдельного участка легочных тканей. Преимущественно, она возникает ещё в детском возрасте, так как организм в это время ещё не имеет сильного иммунитета.

Также, она может показывать себя не только в роли единственного заболевания, но также, может являться осложнением различного вида инфекций и их процессов.

Существует множество причин, по которым может происходить развитие пневмонии. Однако, данные причины можно разделить на несколько групп.

Первичная пневмония развивается в организме детей в том числе и потому, что их организм ещё не знаком с различными возбудителями, а также имеет ослабленный иммунитет.

В то же время, для появления пневмонии у взрослых необходимы некоторые предрасполагающие факторы:

- нахождение в стационаре (в этом случае возникает внутрибольничная пневмония);
- частые переохлаждения;
- нахождение на искусственной вентиляции легких (ИВЛ);
- наличие каких-либо острых и хронических болезней пищеварения, дыхания или сердца и сосудов, почек;
- длительное курение и употребление алкоголя [1].

Признаки пневмонии на рентгеновском снимке зависят от того, на какой стадии находится заболевание, а также от локализации патологического процесса.

Первым признаком пневмонии на рентгеновском снимке является появление различных очагов затенения с неровными контурами в разных частях легкого. Данные очаги могут иметь размер от 3-4-х до 12 мм. Чем темнее будет такое пятно - тем сильнее выражен патологический процесс.

Однако, даже если на снимке видны признаки пневмонии, остановка диагноза при пневмонии должна основываться не только на рентгеновских снимках, но также и на результатах выслушивания грудной клетки, анализов крови и мокроты и других методах обследования.

Если исследование показало признаки пневмонии, то необходимо как можно скорее приступить к лечению, так как запущенный патологический процесс опасен не только для здоровья, но и для жизни человека.

## **1.2 Изучение архитектур нейронных сетей**

### **1.2.1 Определение функционала нейронной сети**

Разработку нейронной сети можно разделить на несколько этапов, а именно:

- выбор архитектуры нейронной сети;
- обучение сети;
- тестирование сети;
- внедрение сети.

Данная сеть, должна будет работать с набором данных, представляющем собой набор рентгеновских снимков, в формате png. Данный набор включает в себя несколько тысяч снимков с тремя видами патологий, а также снимки без патологий.

Так как рентгенограммы обладают набором различных признаков, а также, одна рентгенограмма, может содержать в себе признаки различных классов патологий, нейронной сети необходимо будет определить, присутствуют ли на снимках аномалии, определенные, как признаки патологий.

Для того чтобы достичь максимальной точности, будет рассмотрена задача бинарной классификации патологий. Будет представлено два класса, а именно: с аномалией и нормальный.

Также, для достижения наибольшей точности, будут проанализированы часто используемые архитектуры нейронных сетей, после чего, наиболее подходящая для выполнения поставленной задач архитектура нейронной сети будет реализована.

### **1.2.2 Анализ нейронных сетей**

Для начала, необходимо понять, что из себя представляют нейронные сети. Сами по себе, нейронные сети представляют собой модель, созданную на подобие нейронных сетей мозга, то есть биологических нейронных сетей.

Такие модели, используя небольшие элементы, называемые искусственными нейронами, имитирует свой биологический аналог в сети.

Идея создания данного вида сети появилась при углублении в изучении работы искусственного интеллекта. Для получения большей эффективности и изучения новых методов, было предпринято попытки воссоздать способность тренировки и обучения биологических сетей, а также их способность по исправлению ошибок. Результатом этих исследований стало создание первых нейронных сетей.

На данный момент, нейронные сети активно используются для решения различных задач [4].

Например, в область применения нейронных сетей входят:

- автоматизация процессов распознавания образов;
- прогнозирование;
- адаптивное управление;
- создание экспертных систем;
- организация ассоциативной памяти;
- обработка аналоговых и цифровых сигналов;
- синтез и идентификация электронных цепей и систем [4].

Модели нейронных сетей делятся на аппаратные и программные исполнения.

Нейронная сеть состоит из определённого количества нейронов, которые образуют собой слои в нейронной сети. Как показано на рисунке ниже, нейроны имеют связи между собой на одном слое, однако они связаны с нейронами из других слоёв. Далее, на рисунке 1, представлен пример сети с несколькими слоями, где наглядно показана синаптическая связь между нейронами.

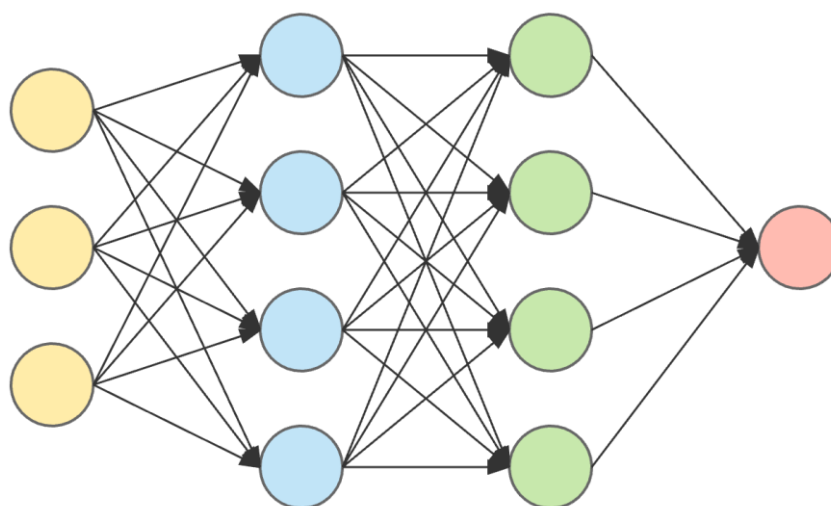


Рисунок 1 – Нейронная сеть

Количество слоёв в нейронной сети определяет точность, с которой она будет решать различного рода задачи. Следовательно, для увеличения точности нейронной сети, можно увеличить количество слоёв или же количество нейронов на них.

Однако, при создании большого количество слоёв или нейронов, скорость решения задачи может значительно уменьшиться, а сложность модели резко повыситься. Поэтому, необходимо правильно проанализировать поставленную задачу для того, чтобы правильно спроектировать нейронную сеть.

Функции биологического нейрона, в нейронной сети, выполняет так называемый искусственный нейрон. Он является неотъемлемой частью любой нейронной сети.

Такой нейрон отвечает за формирование выходного сигнала, получаемого из выходов нейронов, расположенных на других слоях нейронной сети. Данные сигналы передаются между нейронами при помощи синаптических связей, связывающих нейроны между собой.

Чаще всего, сигналы других нейронов, попадающие в нейрон, при помощи синаптических связей, суммируются используя адаптивный сумматор.

Далее, после сложения сигналов, при помощи функции активации определяется состояние нейрона, после чего данный сигнал подаётся на выход нейрона, то есть в точку ветвления.

Пример общего вида искусственного нейрона с наглядно показанными входами, синаптическими связями и выходом представлен на рисунке 2.

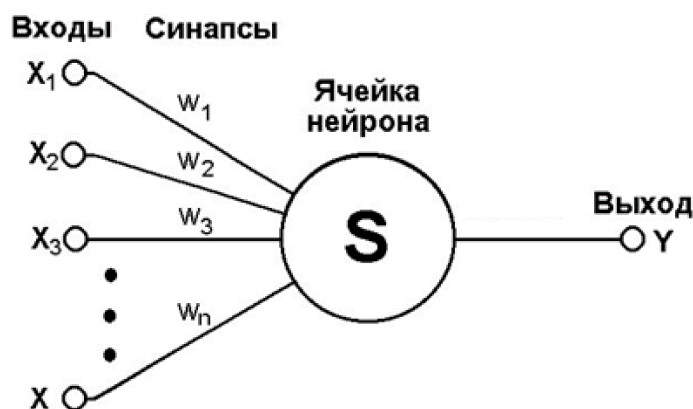


Рисунок 2 – Искусственный нейрон

Для передачи сигнала на выход и в синапсы других нейронов, используется аксон. Он и представляет собой выходную связь нейрона.

У всех синапсов есть свой вес, обозначаемый как  $w_i$ . Также он может называться синаптической связью. Сумма, вычисляемая внутри нейронов, в следствии сложения входных значений, называется взвешенной суммой. Вычисление данной суммы происходит по формуле 1, представленной ниже.

$$S = \sum_{i=1}^n x_i \times w_i \quad (1)$$

где  $S$  – взвешенная сумма;

$x_i$  – сигналы на входах нейрона;

$w_i$  – веса входов.

В свою очередь, выход нейрона есть функция его состояния, представленная на формуле 2.



$$y = f(s) \quad (2)$$

где  $f(s)$  – придаточная функция.

Далее будет представлено описание некоторых видов функции активации, используемых для вычисления состояния нейрона.

### 1.2.3 Описание функций активации

Функцией активации является некая нелинейная функция, используемая внутри нейрона, для определения его выходного сигнала.

Для правильного выбора функции активации, необходимо тщательно изучить поставленную задачу, так как неправильный выбор функции может негативно повлиять на результат работы сети.

От выбранной функции активации будет зависеть то, как будет проходить процесс обучения и тренировки нейронной сети.

Далее будут рассмотрены некоторые виды функций активации.

Линейная функция. Данная функция представляет собой прямую линию и пропорциональна входу, то есть взвешенной сумме на этом нейроне и описывается выражением, представленным на формуле 4.

$$f(s) = k \times s \quad (4).$$

Данная функция состоит из участков, на котором она имеет минимально и максимально допустимые значения, а также из участка, на котором происходит равномерное возрастание функции.

В качестве минимального и максимального выходных значений часто используют пары коэффициентов 0 и 1, -1 и 1. Сама линейная функция представлена ниже, на рисунке 4.

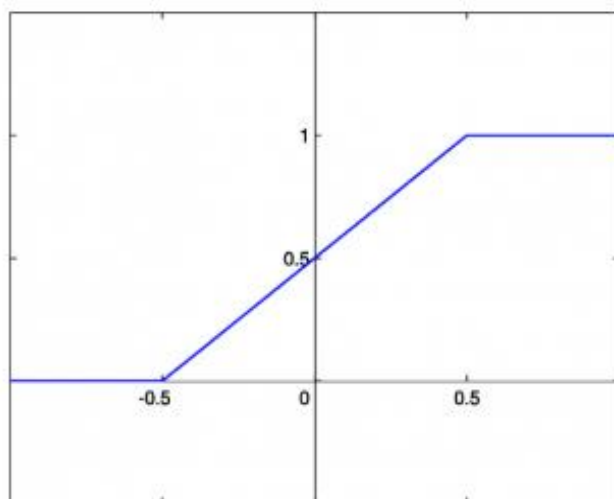


Рисунок 4 – Линейная функция

По рисунку, представленному выше, а также описанию самой линейной функции, можно понять, что данная функция может обеспечить хорошую производительность и перспектива её использования гораздо выше, чем у пороговой функции.

Сигмоидальная функция. Данная функция может напоминать своей формой описанную выше функцию, однако при этом она имеет более сглаженную форму. Сигмоидальная функция описывается выражениями, представленными в формулах 5 и 6, для смешанной и симметричной функции, соответственно.

$$f(s) = \frac{1}{1 + e^{-s}} \quad (5)$$

$$f(s) = \frac{1 - e^{-s}}{1 + e^{-s}} \quad (6)$$

При выборе данной функции необходимо сформулировать конкретную задачу, так как данная функция является непрерывной.

Также, данная функция является:

– непрерывной;

- монотонно возрастающей;
- дифференцируемой [8].

Наибольшее и наименьшее значения данной функции соответствуют значениям линейной функции. Сама сигмоидальная функция активации представлена ниже, на рисунке 5.

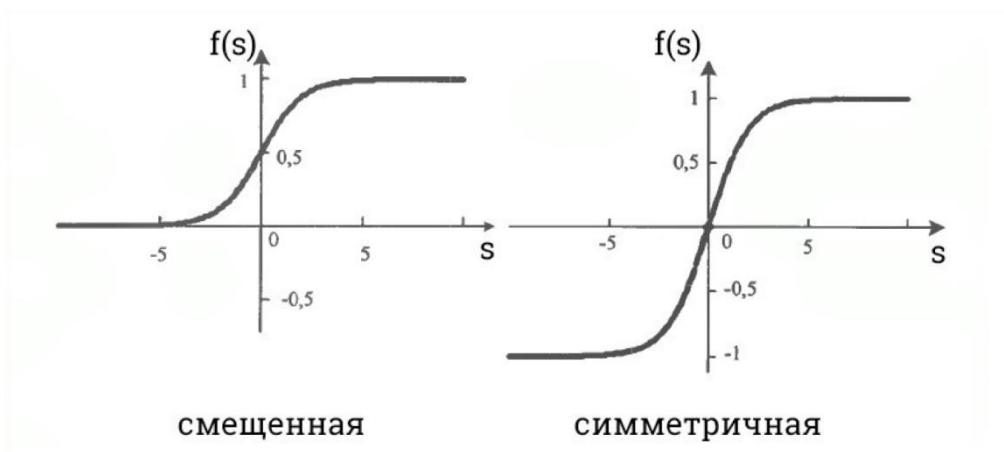


Рисунок 5 – Смещенная и симметричная сигмоидальные функции

Данную функция использует положительные черты других, описанных выше, функций. Это делает данную функцию хорошим выбором для решения поставленных задач.

Далее, функция активации ReLu. Эта функция используется, в основном, для решения проблем с градиентами и представляет собой изменённую версию линейной функции. Описывается данная функция выражением, представленным на формуле 8.

$$f(s) = \max(0, s) \quad (8)$$

Применение данной функции активации значительно повышает стохастического градиентного спуска, в особенности, по сравнению с сигмоидой и гиперболическим тангенсом.

Также, используя пороговое преобразование, можно реализовать данную функцию, в более оптимизированной версии. Такая функция будет являться менее трудоёмкой операцией чем вычисление сигмоиды и гиперболического тангенса. Кроме того, ReLU не подвержен насыщению.

Однако, Основным недостатком функции является – отключение большей части нейронов при неправильном подборе параметров. На рисунке 7, представлена сама функция активации.

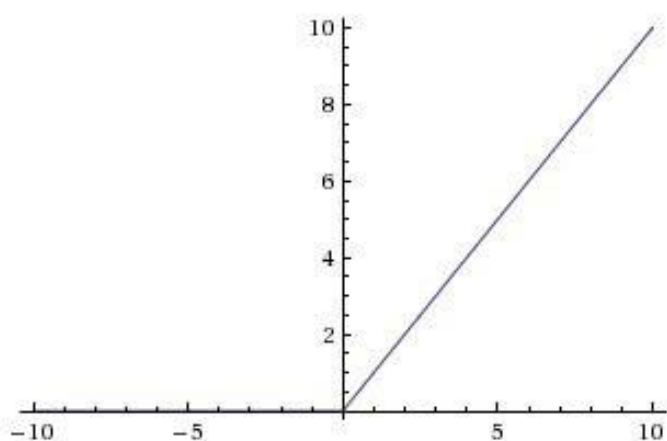


Рисунок 7 – Функция активации ReLu

Для решения задач классификации изображений или же прогнозирования, чаще всего, используется функция ReLu, так как является наиболее подходящим для этого типа задач вариантом. Остальные же функции в большинстве случаев, используются для обучения или же в каких-либо тренировочных заданиях, так как являются наиболее простыми для понимания и наглядно показывают принципы работы функции активации.

Далее будет представлено описание некоторых базовых архитектура нейронных сетей, после чего, будет выбрана наиболее подходящая для решения поставленной задачи.

#### 1.2.4 Описание архитектур нейронных сетей

Нейронные сети представляют собой набор слоёв с различными свойствами, между которыми передаются сигналы. При этом, элементы сети,

находящиеся на одном слое, не имеют между собой связи и связаны только с элементами из соседних слоёв.

Минимальным набором элементов нейронной сети является два входных и один выходной нейрон. Такая структура нейронной сети позволит ей использоваться в качестве логической модели.

Обычно, для обучения сетей такого вида, используется метод обратного распространения ошибок. При использовании данного метода, происходит передвижение по сети большого количества данных разного вида.

Этот процесс называется обучением с учителем. Его отличие от обучения без учителя состоит в том, что в случае обучения без учителя, сеть самостоятельно составляет множество выходных данных [11].

Теоретически, если у сети есть достаточное количество скрытых нейронов, она может моделировать взаимодействие между входным и выходными данными. Возможные формы сети прямого распространения показана на рисунке 8.



Рисунок 8 – Формы нейронной сети прямого распространения

Такой вид нейронной сети используется для объединения с другими видами, для получения новой информации.

Нейронная сеть Хопфилда — это полносвязная сеть, то есть сеть, в которой каждый нейрон соединен с каждым. Особенностью данного вида нейронной сети является состояние нейронов при получении данных, а также изменение их состояния в процессе обучения.

При обучении данной нейронной сети, происходит установка значений для элементов сети, то есть нейронов. После установки, происходит нахождение весов для синапсов, которые больше не будут изменяться.

После завершения процесса обучения на некотором количестве шаблонов, сеть будет сводить к одному из них, однако, не всегда к желаемому.

Система стабилизируется только частично из-за того, что общая “энергия” или “температура” сети во время обучения постепенно понижается.

Нейроны в данной сети обладают определённым порогом активации, соизмеримым с температурой. Это значит, что при превышении этого порога нейрон может перейти из одного состояния в другое. Чаще всего, узлы данной сети последовательно обновляются, в следствии чего, происходит генерация некоторой случайной последовательности, определяющей порядок обновления состояний нейронов. После такого обновления, состояния нейронов больше не будут изменяться, а сеть приходит в стационарное состояние.

Часто, такие сети называют ассоциативной памятью, так как они сходятся к состоянию, наиболее близкому к заданному. Форма нейронной сети Хопфилда представлена на рисунке 9.

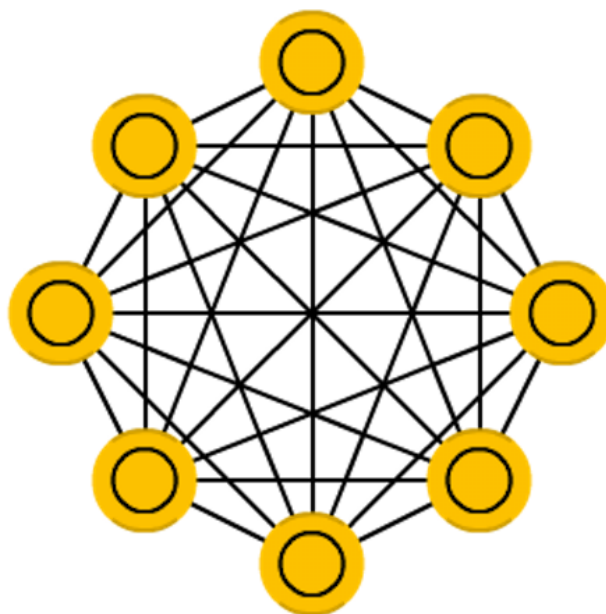


Рисунок 9 – Форма нейронной сети Хопфилда

Машины Больцмана, это нейронные сети, во многом похожи на сети Хопфилда, однако в них некоторые нейроны помечены как входные, а некоторые остаются скрытыми [17]. Когда все нейроны обновляют свои состояния, входные нейроны становятся выходными

В начале, случайным образом, затем происходит обучение методом обратного распространения, однако в последнее время, часто используется алгоритм *contrastive divergence*, при котором градиент вычисляется при помощи марковской цепи.

Так как в процессе обучения задействована цепь Маркова, данная сеть является стохастической. Процесс работы и обучения схож с сетью Хопфилда, описанной выше. То есть цепь начинает функционировать после того, как нейронам будет присвоены определённые начальные состояния. Форма машины Больцмана представлена на рисунке 10.

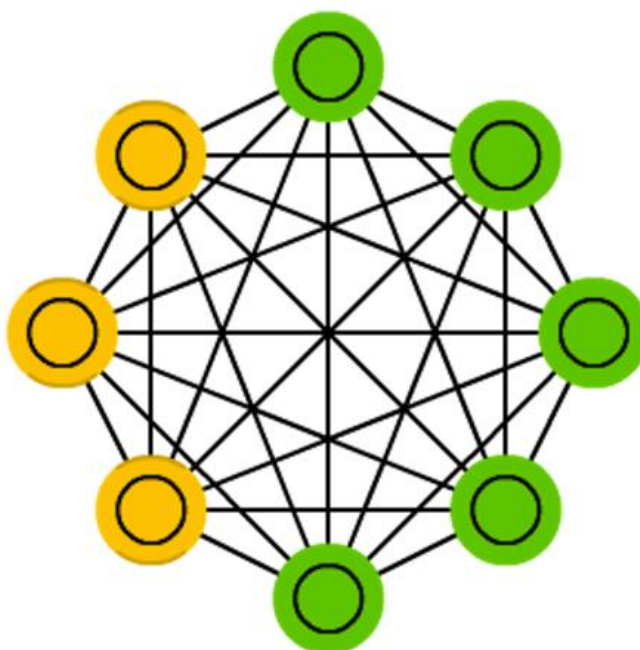


Рисунок 10 – Форма машины Больцмана

Описанные далее автоэнкодеры являются результатом продолжения исследования идей сети прямого распространения. Их основная идея, исходит из их названия и состоит в автоматическом кодировании информации [13].

Данная сеть, своей формой, больше всего напоминает некие песочные часы, так как скрытый слой меньше, чем входной и выходной, при этом. Сеть имеет форму, симметричную относительно средних слоёв.

В средних слоях информация максимально сжата, и они обычно являются самыми маленькими.

До середины расположена кодирующая часть, а выше - декодирующая

Обучение автоэнкодеров происходит при помощи методом обратного распространения ошибки. Также, автоэнкодеры можно построить симметричными и с точки зрения весов, выставляя кодирующие веса равными декодирующим. Форма автоэнкодера показана на рисунке 11.



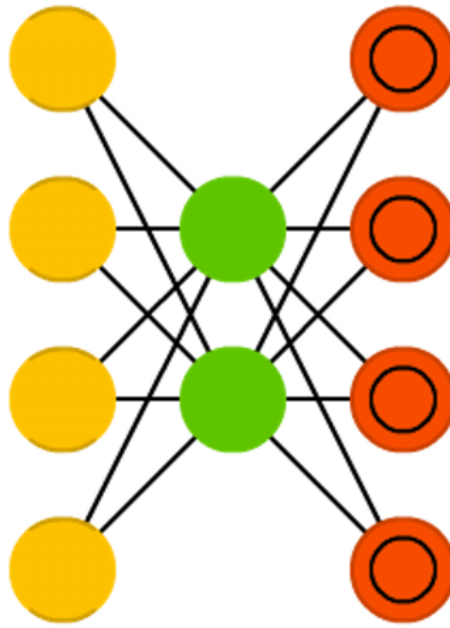


Рисунок 11 – Форма автоэнкодера

Разреженный автоэнкодер, в какой-то степени, представляет собой антипод автоэнкодера.

Вместо того чтобы обучать сеть представлять блоки информации на меньшем “пространстве”, мы кодируем информацию так, чтобы она занимала больше места. Также, вместо сходимости системы в центре, происходит расширение.

Данные сети применяются для извлечения множества маленьких деталей из набора данных.

При попытке обучить разреженный автоэнкодер тем же методом, что и обычный автоэнкодер, в большинстве случаев, получается абсолютно бесполезная сеть, так как на выходе получается ровно то же, что и на входе.

По этому, при обучении разреженного автоэнкодера вместо входных данных на выходе выдаются входные данные и определённый штраф за количество активированных нейронов в скрытом слое. Форма разреженного автоэнкодера представлена на рисунке 12.

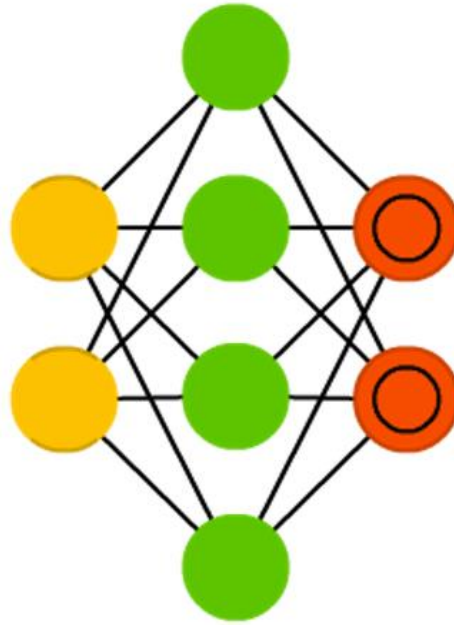


Рисунок 12 – Форма разреженного автоэнкодера

Описанные далее сверточные нейронные сети являются наиболее необычным представителем из описанных сетей. В большинстве случаев, данный вид сетей используется для задач, различным образом, связанных с обработкой изображений.

Примером использования сверточных нейронных сетей является классификации изображений, когда на вход подаётся изображение с животным, а сеть определяет к какой категории животных относится животное на изображении. Форма данной сети показана на рисунке 13.

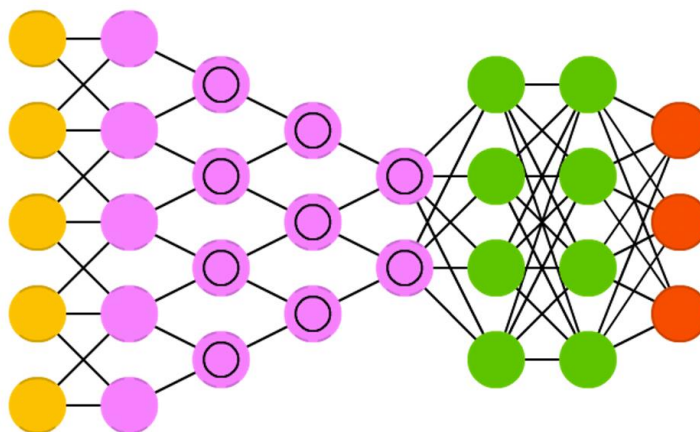


Рисунок 13 – Форма свёрточной нейронной сети

Среди описанных выше архитектур было решено воспользоваться сверточной нейронной сетью, так как данная сеть обладает рядом преимуществ для работы с изображениями. А также, поэтапно обрабатывает изображения с помощью карт свертки и отлично подходит для классификации.

Далее будет подробнее рассмотрена работа сверточной нейронной сети.

### **1.2.5 Анализ работы сверточной нейронной сети**

На данный момент, сети со сверточной архитектурой, считаются одним из наиболее инновационных решений в области задач, решаемых компьютерным зрением. Одной из наиболее востребованных задач, для таких сетей, принято считать бинарную классификацию изображения.

При выполнении задачи бинарной классификации, полученное изображение проходит через нейронную сеть, после чего, происходит вывод класса, наиболее подходящего данному изображению.

Также для того, чтобы классифицировать, данной нейронной сети необходимо найти и определить различные характеристики базового и более продвинутого уровня, каждая из которых определяет принадлежность к классу.

Базовыми характеристиками могут являться небольшие элементы изображения, вроде набора пикселей определённого цвета, однако более сложные характеристики представляют собой наиболее характерные черты изображения.

После того как изображение передаётся в нейронную сеть, она проходит через определённое количество слоёв, заданное при проектировании. Так как данная сеть является сверточной, изображение будет проходить через некоторое количество сверточных слоёв, которые как раз таки и обеспечивают свёртку.

В начале сверточной нейронной сети должен находиться слой свёртки, отвечающий за первый этап свёртки изображения. На это слое, фильтр, движется по рецептивным полям изображения. Сам фильтр является матрицей, с глубиной, такой же что и у изображения.

После того, как фильтр произвёл свертку, происходит умножение его значений на значения пикселей исходного изображения. Далее, происходит суммирование полученных значений, для получения значения, определяющего позицию данного фильтра на следующем слое. Процесс умножения и сложения повторяется, перемещаясь на один пиксель в определённом направлении.

Далее, когда фильтр пройдёт все пиксели, из полученных после сложений значений, будет сформирована новая матрица. Данная матрица также называется картой свойств изображения. Описанный выше процесс показан на рисунке 14.

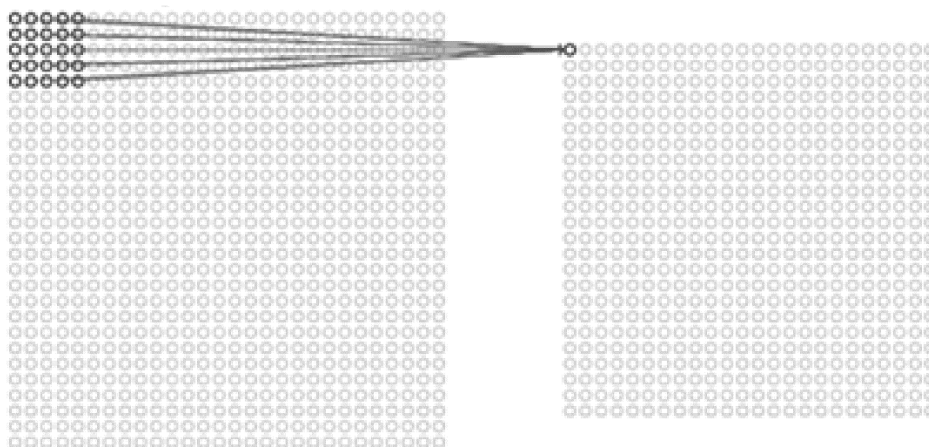


Рисунок 14 – Свертка изображения фильтром

Кроме сверточных слоёв, в структуре данной нейронной сети существуют и другие слои, которые чередуются со сверточными. Классическая архитектура сверточной нейронной сети представлена ниже, на рисунке 15.

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

Рисунок 15 – Классическая архитектура сверточной нейронной сети

После прохождения изображения первого сверточного слоя, результатом которого будет матрица свойств, оно переходит на второй слой и так далее. После каждого из слоев формируется матрицы, определяющие места, на которых определены базовые признаки изображения.

Кроме слоев свертки, важной частью сети являются слои подвыбоки. Суть операции подвыбоки состоит в том, что изображение сжимается, то есть уменьшается в размере, путём сложения значений блоков пикселей.

Для выполнения операции подвыборки, необходимо определиться с размером выборки (размером прямоугольной сетки) и величиной шага. Размер выборки определяет, из какого числа пикселей будет происходить выборка, а шаг определяет количество пикселей, на которые необходимо сдвигать сетку при выполнении операции подвыборки.

После определения размеров выборки и шага, происходит поиск максимального значения пикселя попадающего в сетку. После этого, сетка сдвигается на установленное для шага количество пикселей, а пиксель с максимальным значением переноситься в новое изображение. Данный процесс показан на рисунке 16.



Рисунок 16 – Операция подвыбоки

Далее, на других слоях со сверткой, фильтры, определяющие наиболее сложные фрагменты, будут проверять данное изображение. В зависимости от количества сверточных слоёв, а также от размера самой сети, фильтры будут становиться сложнее, другими словами, они смогут определять более сложные характеристики на изображении.

После нахождения самых сложных свойств, последний слой, который является полносвязным, по полученным данным определяет к какому классу относится данное изображение. Данный слой выводит вектор определённого размера, в зависимости от числа классов, из числа которых будет производиться выбор.

К примеру, в случае определения пневмонии на снимке, у вектора будет только один класс от значения которого будет зависеть, распознала программа пневмонию или нет.

В работе полносвязного слоя присутствует обращение к предшествующим его слоям, для получения результата из работы, которыми должны являться карты со сложными характеристиками. Также, данный слой производит соответствие различных свойств и классов между собой.

Например, если необходимо распознать пневмонию на рентгенограмме, то у данных карт, отображающих сложные характеристики пневмонии, должны быть значения больше.

Полносвязный слой производит соответствие сложных функций и определённых ранее классов, а также их весов. Это означает, что при вычислении произведения между весами и предыдущим слоем, будут получены правильные значения для классов.

Обучение данной нейронной сети происходит при помощи метода обратного распространения ошибок.

Данный метод можно разделить на 4 части, а именно:

- прямое распространение;
- функцию потерь;

- обратное распространение;
- обновление веса [14].

При тренировке нейронной сети, в саму сеть передаётся тренировочное изображение, проходящее через всю структуру сети. При этом, пока тренировочное изображение проходит через сеть, из-за того, что веса и значения для фильтра были составлены случайным образом, сеть не может обосновано определить класс изображения, что ведёт к функции потери.

В основном, функция потери выражается как среднеквадратическая ошибка. Сама формула вычисления среднеквадратической ошибки отображена в формуле 9.

$$E_{total} = \sum \frac{1}{2} (tar - out)^2 \quad (9)$$

где tar – истинное значение переменной;

out – предполагаемое значение переменной.

Так как значения сгенерированы случайным образом, некоторого количества первых входных изображений, потеря может быть высокой, однако, далее это измениться.

Далее, необходимо правильно выбрать веса, так как от них напрямую зависит значение потери, которое необходимо уменьшить. Для того, чтоб сделать это, нужно найти так называемую производную потери, которая находится путём выполнения обратного распространения. При выполнении обратного распространения будет видно, какие веса ответственны за потери, после чего, необходимо будет их отрегулировать, для уменьшения потерь.

Другими словами, необходимо взять фильтровые веса и обновить их так, чтобы они менялись в направлении градиента [15]. Обновление весов происходит по формуле 10, представленной ниже.

$$w = w_i - \eta \frac{dL}{dW} \quad (10)$$

где  $w_i$  – изначальный вес;

$\eta$  – скорость обучения.

При описании нейронной сети, автор сам выбирает скорость обучения своей нейронной сети, так как данный параметр должен соответствовать поставленной задаче.

Например, если выбрать высокую скорость обучения, то при обучении нейронной сети, веса будут изменяться на большие значения вверх или вниз, что может ускорить время обучения модели нейронной сети. При этом, существует риск, что при большой скорости обучения, большие изменения весов могут помешать достижению требуемых показателей.

Описанные выше процессы прямого распространения ошибки, функции потерь, а также обратного распространения и обновления значений весов принято называть одним периодом дискретизации или эпохой [16].



## ГЛАВА 2 РАЗРАБОТКА НЕЙРОННОЙ СЕТИ

### 2.1 Обоснование выбора технических средств

На сегодняшний день, существует множество языков, применяемых для создания нейронной сети.

Далее будут рассмотрены основные языки, для которых созданы библиотеки глубокого обучения.

Язык MATLAB является инструментом, обеспечивающим взаимодействие оператора с различными возможностями для анализа, сбора и представления данных.

У данного языка есть достоинства и недостатки, которые свойственны языку, живущему в замкнутой экосистеме.

Достоинства:

- язык легок для изучения;
- наличие готовых модулей для работы с нейронными сетями;
- частые обновления.

Недостатки:

- перегруженность операторами и функциями;
- дороговизна ПО;
- узко направленность.

Java – это высокоуровневый язык, синтаксис которого основан на C ++, поэтому Java похожа на C. Тем не менее, синтаксис Java проще, что позволяет новичкам быстрее учиться и эффективнее использовать код для достижения конкретных результатов.

Достоинства:

- безопасность;
- независимость от платформы;
- автоматическое управление памятью.

Недостатки:

- многословный и сложный код;

- отсутствие встроенного дизайна.

Python – это высокоуровневый язык программирования, применяемый для различного рода задач [7].

Преимущества:

- легок в изучении;
- большое сообщество;
- наличие различных сервисов и фреймворков.

Недостатки:

- сложный в использовании API;
- низкая производительность.

Для решения поставленной задачи, было решено использовать Python. Так как данный язык имеет ряд значимых преимуществ и отлично подходит для создания сверточной нейронной сети.

Далее будут рассмотрены основные фреймворки, предназначенные для работы с нейронными сетями.

TensorFlow – это открытая библиотека, созданная Google на Python и C++.

Данный фреймворк является хорошим решением для создания сложных проектов. Также, он поддерживается большим сообществом и имеет мощные средства для мониторинга процесса обучения моделей и визуализации.

Torch – это научная вычислительная среда с широкой поддержкой алгоритмов машинного обучения. Он обеспечивает максимальную гибкость и скорость в построении научных алгоритмов, делая данный процесс чрезвычайно простым. Также, данный фреймворк поставляется с большой количеством библиотек, ориентированных на машинное обучение.

Keras – это библиотека, основанная на Python, с возможностью запуска поверх TensorFlow и Theano [17]. Keras поддерживает широкий спектр слоев нейронных сетей, таких как сверточные слои, рекуррентные или плотные.

Для решения поставленной задачи было решено использовать Keras и TensorFlow, так как данные фреймворки отлично подходят для реализации нейронной сети.

## 2.2 Архитектура нейронной сети

В данной нейронной сети используется два слоя свертки, после каждого из них располагается подвыборочный, а в конце, находятся два полносвязных слоя. Данная архитектура была выбрана в ходе тестирования комбинаций различного вида и количества слоёв. В данном случае, увеличение количества слоёв различного типа и изменение размеров входного изображения плохо сказывалось на точности нейронной сети. Построение нейронной сети показано на рисунке 17.

```
cnn = Sequential()

#Convolution
cnn.add(Conv2D(32, (3, 3), activation="relu", input_shape=(64, 64, 3)))

#Pooling
cnn.add(MaxPooling2D(pool_size = (2, 2)))

# 2nd Convolution
cnn.add(Conv2D(32, (3, 3), activation="relu"))

# 2nd Pooling layer
cnn.add(MaxPooling2D(pool_size = (2, 2)))

# Flatten the layer
cnn.add(Flatten())

# Fully Connected Layers
cnn.add(Dense(activation = 'relu', units = 128))
cnn.add(Dense(activation = 'sigmoid', units = 1))
```

Рисунок 17 – Построение нейронной сети

Первый слой является сверточным. В начале, задаётся параметр `filters`, значение которого определяет количество выходных фильтров. Рекомендуется использовать степень двойки для данного значения, так как это позволяет не терять информацию при уменьшении размерности в подвыборочном слое. В данном случае, количество входных фильтров задано как 32, данное значение является своего рода стандартом. Далее, задаётся

параметр `kernel_size`, определяющий размер ядра или фильтра свертки. Он определяет высоту и ширину фильтра. Данный параметр должен быть нечётным числом и на данном слое задан как фильтр размером 3 на 3. После размера фильтра, параметром `activation`, задается функция активации, применяемая после свертки. На данном слое задана функция активации ReLu. И последний параметр данного слоя – `input_shape`. Этот параметр задает формат изображения, попадающего на данный слой. В данном случае, сверточный слой принимает изображения размером 64 на 64 пикселя в RGB, то есть с тремя каналами.

Далее, идёт слой подвыборки подвыборки. Единственным задаваемым параметром данного слоя является параметр, отвечающий за размеры подвыборки. В данном случае, размер подвыборки, то есть размер прямоугольной сетки, задан как 2 на 2.

После этого, описанная структура повторяется, с единственным изменением – отсутствием параметра `input_shape` на сверточном слое.

Далее, идёт выравнивающий слой. Данный слой преобразует 2D данные, в 1D, то есть преобразует множество двумерных карт признаков в одномерный вектор. Этот слой не имеет параметров, так как он занимается только преобразованием данных.

В конце, находятся два полносвязных слоя. Первый слой состоит из 128 нейронов и использует функцию активации ReLu, описанную ранее. Данная функция активации менее затратна в вычислительном отношении, чем тангенс и сигмоидальная функция, так как включает в себя более простые математические операции. За один раз активируются только несколько нейронов, что делает сеть разреженной, а также эффективной и легкой для вычислений.

Второй слой имеет только один нейрон и использует сигмоидальную функцию активации, так как должен определить, присутствует ли на изображении патология или нет.

### 2.3 Подготовка набора данных для обучения

Для обучения нейронных сетей, представленных выше необходимо выбрать набор данных, на котором будет происходить обучение. Данные для обучения нейронной сети были загружены с сервера Kaggle [10]. Данный ресурс предоставляет доступ к различным наборам данных, находящихся в свободном доступе.

Полученный набор данных состоит из необработанных рентгеновских снимков с тремя патологиями, а так же со снимками без патологий. Пример снимков с патологией и без неё представлен на рисунке 18.

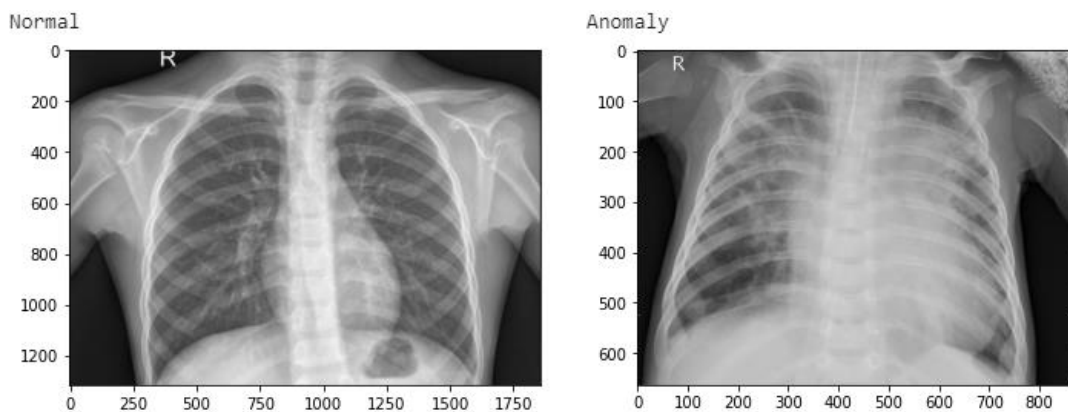


Рисунок 18 – Пример снимков

Данные разделены на три группы, а именно: train, val и test. Train - содержит обучающие данные и изображения для обучения нашей модели, val - содержит изображения, которые используются для проверки модели. Цель этого набора данных состоит в том, чтобы предотвратить чрезмерное приспособление модели. Test - содержит данные, используемые для тестирования модели после того, как она изучила отношения между изображениями и их меткой.

### 2.4 Обучение нейронной сети

Для создания набора данных для обучения будет использоваться метод ImageDataGenerator, реализация которого представлена на рисунке 19.

```

train_datagen = ImageDataGenerator(rescale = 1./255,
                                   shear_range = 0.2,
                                   zoom_range = 0.2,
                                   horizontal_flip = True)

training_set = train_datagen.flow_from_directory('drive/My Drive/data/chest_xray/train',
                                                target_size = (64, 64),
                                                batch_size = 32,
                                                class_mode = 'binary')

```

Рисунок 19 – Создание набора тренировочных данных

В свою очередь, метод `flow_from_directory` формирует поток тренировочных данных из папки `train`.

Для предотвращения переобучения был создан набор данных валидации. Набор формируется из изображений, находящихся в папке `val`. Часть кода, формирующая данный набор представлена на рисунке 20.

```

test_datagen = ImageDataGenerator(rescale = 1./255) #Image normalization.

validation_generator = test_datagen.flow_from_directory('drive/My Drive/data/chest_xray/val/',
                                                       target_size=(64, 64),
                                                       batch_size=32,
                                                       class_mode='binary')

```

Рисунок 20 – Создание набора данных валидации

Далее, рассмотрим часть кода, отвечающую за компиляцию сети. Данная часть кода показана ниже на рисунке 21.

```

cnn.compile(optimizer = 'adam',
            loss = 'binary_crossentropy',
            metrics = ['accuracy'])

```

Рисунок 21 – Компиляцию сети

Функцией активации является `binary_crossentropy`. Такая функция используется для бинарной классификации.

В качестве функции оптимизации выбран adam, являющийся особой модификация стохастического градиентного спуска [19].

Для обучения данной сети использовалась функция fit\_generator, показанная на рисунке 22.

```
cnn_model = cnn.fit_generator(epochs=30,  
                             callbacks=[early_stop,checkpoint, reduce],  
                             shuffle=True,  
                             validation_data=validation_generator,  
                             generator=training_set,  
                             steps_per_epoch=369,  
                             validation_steps=1674,  
                             verbose=2)
```

Рисунок 22 – Функция fit\_generator

Первым параметром в данной функции задано количество эпох, за которое будет проходить обучение модели.

Далее идёт список функций, используемых во время обучения. Early\_stop отвечает за раннюю остановку процесса обучения, если за заданное количество эпох, потери не уменьшается. Реализация данного метода представлена на рисунке 23.

```
early_stop = EarlyStopping(patience=15,  
                           verbose = 2,  
                           monitor='val_loss',  
                           mode='auto')
```

Рисунок 23 – Реализация early\_stop

В данном случае, early\_stop остановит обучение, если потери не уменьшатся за 15 эпох.

Следующая функция отвечает за сохранение набора весов, с наименьшими потерям, при работе с данными валидации и показана на рисунке 24.

```
checkpoint = ModelCheckpoint(  
    filepath='best_model',  
    save_best_only=True,  
    save_weights_only=True,  
    monitor='val_loss',  
    mode='auto',  
    verbose = 1)
```

Рисунок 24 –Реализация checkpoint

Далее идёт функция, отвечающая за уменьшение скорости обучения, при условии, что потери при работе с данными валидации не уменьшаются, в течении установленного количества эпох. Данная функция представлена на рисунке 25.

```
reduce = ReduceLROnPlateau(  
    monitor='val_loss',  
    factor=0.8,  
    patience=5,  
    verbose=1,  
    mode='auto',  
    min_delta=0.0001,  
    cooldown=1,  
    min_lr=0.0001  
)
```

Рисунок 25 – Реализация reduce

Следующий параметр отвечает за перемешивание набора тренировочных данных, перед началом каждой эпохи. В данном случае, значением параметра является true, а значит, данные будут перемешиваться перед началом каждой эпохи.



Далее, передаётся созданный ранее набор с данными валидации, необходимый для предотвращения переобучения, которое возникает при обучении сети на обучающем наборе данных.

После набора с данными валидации, передается набор с данными для тренировки сети, также созданный ранее.

Следующим параметром задается количество шагов, выполняемых генератором, в начале новой эпохи, вычисляемое как количество тренировочных данных деленное на `batch_size`.

И в конце, устанавливается число шагов валидации, равное количеству тестовых данных, и режим детализации.

Далее, будет описан процесс тестирования реализованной архитектуры нейронной сети, на специально подготовленном наборе данных, а также внедрение сети в приложение для работы с рентгенограммами, которое позволяет наглядно показать возможности нейронной сети.

## ГЛАВА 3 ТЕСТИРОВАНИЕ И ВНЕДРЕНИЕ НЕЙРОННОЙ СЕТИ

### 3.1 Тестирование сети

Тестирование нейронной сети происходило на наборе данных, сформированном при помощи метода `flow_from_directory` формирует поток тренировочных данных из папки `test`. Реализация данного метода представлена на рисунке 26.

```
test_set = test_datagen.flow_from_directory('drive/My Drive/data/chest_xray/test',
                                          target_size = (64, 64),
                                          batch_size = 32,
                                          class_mode = 'binary')
```

Рисунок 26 – Создание набора тестовых данных

Оценка работы модели происходит на тестовом наборе данных при помощи метода `evaluate_generator`. Реализация и результат работы данного метода представлены на рисунке 27.

```
test_accu = cnn.evaluate_generator(test_set, steps=363)

print('The testing accuracy is :', test_accu[1]*100, '%')

The testing accuracy is : 78.03091406822205 %
```

Рисунок 27 – Реализация и результат метода `evaluate_generator`

Количество шагов, заданных в методе, определяется путём деления количества изображений на размер выборки. Данный метод показал, что точность данной модели примерно 78%.

Далее, на рисунке 28, будет представлен график, отображающий изменения в точности сети относительно номера эпохи.

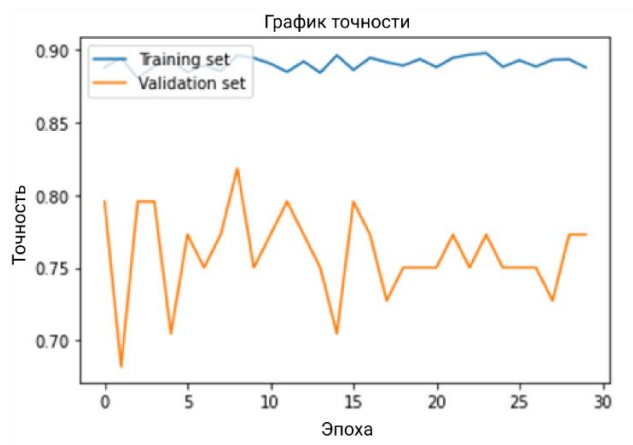


Рисунок 28 – График изменения точности

График, представленный выше, показывает изменения в точности сети, при обучении, в течении 30 эпох, на тренировочном и валидационном наборе данных.

Далее, на рисунке 29, будет представлен график, отображающий изменения в точности сети относительно номера эпохи.

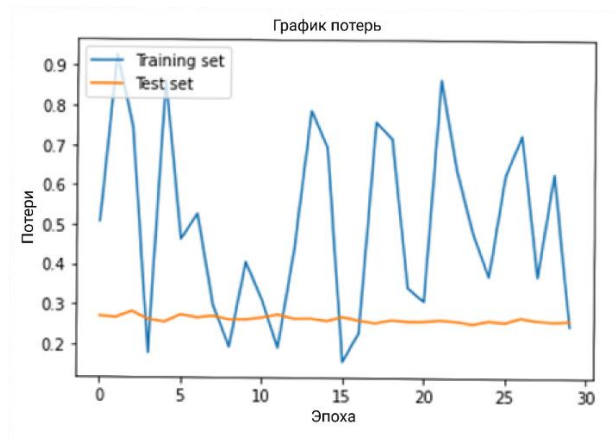


Рисунок 29 – График изменения значения функции потерь

График, показанный на рисунке 27, отображает изменение значения функции потерь, на протяжении 30 эпох обучения сети, на тренировочном и валидационном наборе данных.

В контексте классификации, функция потерь, используется для расчета ошибки между реальными и полученными ответами. Данная функция

показывает, насколько хорошо сеть работает с тренировочным и валидационным набором данных.

### 3.2 Внедрение сети

Для эффективного использования нейронной сети, необходимо создать приложение, использующее возможности распознавания сети. Было решено использовать графическую библиотеку Tkinter, позволяющую создавать программы с оконным интерфейсом.

В данной программе, пользователь загружает изображение рентгенограммы в формате png, после чего, приложение определит, есть ли на данном изображении аномалии. Пример такого изображения представлен на рисунке 30.



Рисунок 30 – Пример входных данных

Далее будет рассмотрено тестирование классификации изображения в данном приложении.

Таблица 1 – Тест-кейс №1 Классификация изображения

Действие	Ожидаемый результат
Запустить приложение	При запуске приложения, должно открыться окно, позволяющее загрузить изображение для классификации.

Загрузить изображение	Выбранное изображение отображается в окне приложения, а также появляется возможность классифицировать его.
Классифицировать изображение	Над изображением выводится результат классификации

Далее, на рисунке 31 будет показан результат работы данной программы.

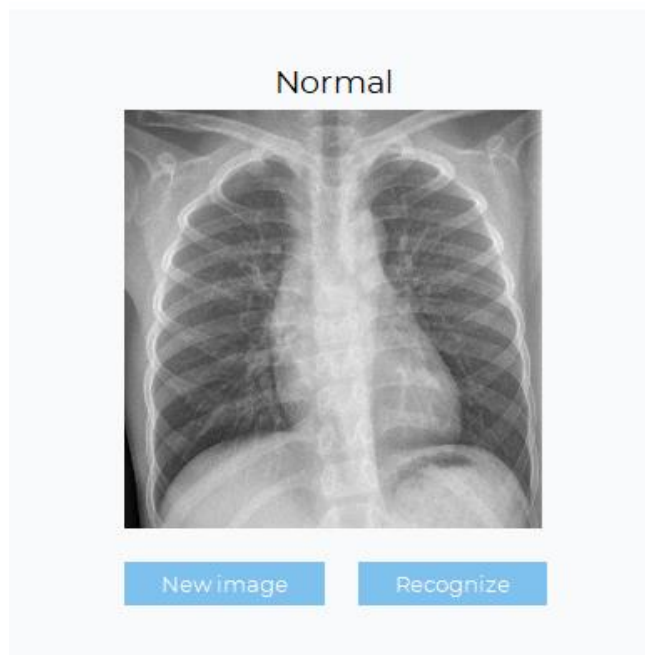


Рисунок 31 – Результат работы программы

После того, как пользователь загрузил изображение, он может классифицировать его или же выбрать другое изображение.

Для классификации используется кнопка “Recognize”, после нажатия которой, вызывается функция, подготавливающая изображение для дальнейшей классификации. Фрагмент кода данной функции представлен на рисунке 32.

```
pixel = 64
img = cv2.cvtColor(cv2.imread(self.path), cv2.COLOR_BGR2RGB)
feed_img = cv2.resize(img, (pixel, pixel))
```

Рисунок 32 – Фрагмент функции, подготавливающей изображение

После того, как изображение преобразовывается в подходящий для классификации вид, оно передаётся в модель нейронной сети, загруженную ранее.

Далее, модель возвращает число, определяющее, к какому классу относится данное изображение. Результат классификации, в зависимости полученного значения выводится в окне приложения, в виде названия класса. Фрагмент кода функции классификации представлен на рисунке 33.

```
model = load_model('cnn_model.h5')  
pr = model.predict_classes(feed_img.reshape(1, pixel, pixel, 3))
```

Рисунок 33 – Фрагмент функции классификации

Классификатор, обученный на снимках, распознает с достаточной точностью. Точность, в свою очередь, зависит не только от объема обучающей выборки, но и от количества информации в обучающей выборке, а также от используемой модели.

В данной главе было рассмотрено тестирование и внедрение нейронной сетей, а также составлен тест-кейс для проверки классификации внутри приложения.

## Заключение

Выпускная квалификационная работа посвящена разработке нейронной сети для обнаружения аномалий на рентгеновских снимках, а также разработке приложения, позволяющего использовать данную нейронную сеть. Целью работы являлось реализация нейронной сети для распознавания патологий грудной клетки.

Перед началом разработки нейронной сети, необходимо было определить соответствие аномалий на снимках с существующими патологиями. Для этого были изучены основные виды патологий, а также их характерные признаки.

Было установлено, что среди методов диагностики оптимальными являются флюорография и рентгенография, так как они не требуют больших временных затрат, и изображения, полученные с помощью этих методов, обладают достаточным набором информативных признаков для классификации патологий. Данные методы широко распространены, что также является весомым преимуществом.

Также, были проанализированы различные виды архитектур нейронных сетей, после чего, был выбран наиболее подходящий для выполняемой задачи вид. Сверточные нейронные сети показали себя лучше всего, при решении задач похожего типа и поэтому были выбраны для решения текущей задачи.

Для получения максимальной эффективности, были изучены различные типы функций активации. Наиболее подходящей для поставленной задачи была выбрана функция активации ReLu, так как она имеет ряд преимуществ и подходит для решения поставленной задачи.

После выбора архитектуры и функции активации, были подготовлены данные, для этапов тренировки, тестирования и валидации. Данные были получены из открытых источников и представляют собой рентгеновские снимки грудной клетки в формате png. Данный набор снимков включает в себя три вида патологий, а также снимки без патологий.

После подготовки данных, были проведены этапы тренировки и тестирования, во время которых, нейронная сеть показала достаточную точность при решении поставленной задачи.

Для использования функционала нейронной сети, был спроектирован и реализован простейший пользовательский интерфейс, наглядно демонстрирующий работу сети. Он включает в себя две кнопки для управления процессами загрузки и анализа изображения, а также места для отображения результата анализа и загруженного для анализа снимка.

В рамках выпускной квалификационной работы все цели и задачи выполнены.

Основные патологии грудной клетки и методы их диагностики рассмотрены.

Разработанная нейронная сеть, с достаточной точностью, решает поставленные задачи, а именно: находит аномалии на рентгеновском снимке, а также увеличивает скорость анализа снимков в целом, тем самым, снимая нагрузку с врачей.



## Список используемой литературы

1. Браженко Н.А. Туберкулез органов дыхания. Издательство: СпецЛит, 2012 г. – 368 стр.
2. Мустафаев А.Г. Применение искусственных нейронных сетей для ранней диагностики заболевания сахарным диабетом // Кибернетика и программирование. 2016. № 2. С. 1-7. DOI: 10.7256/2306-4196.2016.2.17904
3. Стюарт Рассел, Искусственный интеллект. Современный подход. Издательство: Вильямс, 2018 г. – 1408 стр.
4. Мустафаев А.Г. Применение искусственных нейронных сетей для ранней диагностики заболевания сахарным диабетом // Кибернетика и программирование. 2016. № 2. С. 1-7. DOI: 10.7256/2306-4196.2016.2.17904.
5. Назаренко А.В. Компьютерное зрение. Современный подход. Издательство: Вильямс, 2004 г. – 928 стр.
6. Осовский С. Нейронные сети для обработки информации. М.: Финансы и статистика, 2012. 344 с.
7. Петер Флах, Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных. Издательство: ДМК Пресс, 2015 г. – 400 стр.
8. Саймон Хайкин, Нейронные сети. Полный курс// Вильямс. 2016.
9. Уэс Маккинни, Python и анализ данных. Издательство: ДМК Пресс, 2015 г. – 428 стр.
10. Kaggle [Электронный ресурс]: Health National Institutes. NIH Chest X-rays. – Режим доступа: <https://www.kaggle.com/nih-chest-xrays/data>
11. Melin, P. Design of Intelligent Systems Based on Fuzzy Logic, Neural Networks and Nature-Inspired Optimization / Patricia Melin, Oscar Castillo, Janusz Kacprzyk. – Springer International Publishing, 2015. – 637 p.
12. Shanmuganathan, S. Artificial Neural Network Modelling / Subana Shanmuganathan, Sandhya Samarasinghe. – Springer International Publishing, 2016. – 472 p.

13. Chen, M. Automated Segmentation of the Choroid in EDI-OCT Images with Retinal Pathology Using Convolution Neural Networks/ Min Chen, Jiancong Wang, Ipek Oguz, Brian L. VanderBeek, James C. Gee – Springer Cham, 2017 – 643 p.
14. Hai Le, H. Automatic Detection of Singular Points in Fingerprint Images Using Convolution Neural Networks/ Hong Hai Le, authorNgoc, Hoa NguyenTriThanh Nguyen – Springer Cham, 2017 – 342 p.
15. Jiang, H. Fingerprint Minutiae Detection Based on Multi-scale Convolution Neural Networks/ Huinan Jiang, Manhua Liu – Springer Cham, 2017 – 313 p.
16. Autoencoder for words / Cheng-Yuan Liou, Wei-Chen Cheng, Jiun-Wei Liou, Neurocomputing, 2014. – 96 p.
17. Lu, H. Calculate Deep Convolution Neural Network on Cell Unit/ Haofang Lu, Ying Zhou, Zi-Ke Zhang – Springer Singapore, 2017 – 526 p.
18. ArXiv: Deep Residual Learning for Image Recognition -  
Access mode: <https://arxiv.org/pdf/1512.03385.pdf>
19. Xiao, X. A Deep Learning Model of Automatic Detection of Pulmonary Nodules Based on Convolution Neural Networks (CNNs)/ Xiaojiao Xiao, Yan Qiang, Juanjuan Zhao – Springer Singapore, 2016 – 320 p.
20. Kingma Diederik P., Ba Jimmy. Adam: A Method for Stochastic Optimization / CoRR. – 2014.

## **Приложение А**

### **Режим доступа к программному обеспечению, которое используется для обучения нейронной сети**

Исходный код программы доступен для скачивания в электронном виде.

Режим доступа:

<https://drive.google.com/file/d/1LLnq680PtNKjxPkugoo4hl-OGrUbvGX1/view?usp=sharing>