

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт кибернетики
Направление подготовки 090402 Информационные системы и технологии
Кафедра информационных систем и технологий

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы
Разработка алгоритма и программного обеспечения сегментации и распознавания машиночитаемых бланков на основе сверточной нейронной сети

УДК 004.415.032:004.932.1

Студент

Группа	ФИО	Подпись	Дата
8ИМ5Б	Попова Екатерина Сергеевна		

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Болотова Юлия Александровна	К.Т.Н.		

КОНСУЛЬТАНТЫ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Данков Артем Георгиевич	К.Э.Н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Акулов Петр Анатольевич			

ДОПУСТИТЬ К ЗАЩИТЕ:

Зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
ИСТ	Мальчуков Андрей Николаевич	К.Т.Н., доцент		

Томск – 2017 г.

ЗАПЛАНИРОВАННЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ООП

Код результатов	Результат обучения (выпускник должен быть готов)
Общепрофессиональные компетенции	
P1	Воспринимать и самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте.
P2	Владеть и применять методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе в глобальных компьютерных сетях.
P3	Демонстрировать культуру мышления, способность выстраивать логику рассуждений и высказываний, основанных на интерпретации данных, интегрированных из разных областей науки и техники, выносить суждения на основании неполных данных, анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями.
P4	Анализировать и оценивать уровни своих компетенций в сочетании со способностью и готовностью к саморегулированию дальнейшего образования и профессиональной мобильности. Владеть, по крайней мере, одним из иностранных языков на уровне социального и профессионального общения, применять специальную лексику и профессиональную терминологию языка.
Профессиональные компетенции	
P5	Разрабатывать стратегии и цели проектирования, критерии эффективности и ограничения применимости, новые методы, средства и технологии проектирования геоинформационных систем (ГИС) или промышленного программного обеспечения.
P6	Планировать и проводить теоретические и экспериментальные исследования в области создания интеллектуальных ГИС и ГИС технологии или промышленного программного обеспечения с использованием методов системной инженерии.
P7	Осуществлять авторское сопровождение процессов проектирования, внедрения и сопровождения ГИС и ГИС технологий или промышленного программного обеспечения с использованием методов и средств системной инженерии, осуществлять подготовку и обучение персонала.
P8	Формировать новые конкурентоспособные идеи в области теории и практики ГИС и ГИС технологий или системной инженерии программного обеспечения. Разрабатывать методы решения нестандартных задач и новые методы решения традиционных задач. Организовывать взаимодействие коллективов, принимать управленческие решения, находить компромисс между различными требованиями как при долгосрочном, так и при краткосрочном планировании.
Общекультурные компетенции	
P9	Использовать на практике умения и навыки в организации исследовательских, проектных работ и профессиональной эксплуатации современного оборудования и приборов, в управлении коллективом.

P10	Свободно пользоваться русским и иностранным языками как средством делового общения.
P11	Совершенствовать и развивать свой интеллектуальный и общекультурный уровень. Проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности.
P12	Демонстрировать способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности, способность самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности, способность к педагогической деятельности.

Министерство образования и науки Российской Федерации
 Федеральное государственное бюджетное образовательное учреждение
 высшего профессионального образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
 ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт кибернетики
 Направление подготовки 090402 Информационные системы и технологии
 Кафедра информационных систем и технологий

УТВЕРЖДАЮ:
 Зав. кафедрой
 _____ Мальчуков А.Н.
 (Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

В форме:

Магистерской диссертации

Студенту:

Группа	ФИО
8ИМ5Б	Попова Екатерина Сергеевна

Тема работы:

Разработка алгоритма и программного обеспечения сегментации и распознавания машиночитаемых бланков на основе сверточной нейронной сети	
Утверждена приказом директора (дата, номер)	86/С от 22.02.2017 г.

Срок сдачи студентом выполненной работы:	06.06.2017 г.
--	---------------

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе	<ol style="list-style-type: none"> 1. Примеры Машиночитаемых бланков 2. Обучающая выборка рукописных символов
Перечень подлежащих исследованию, проектированию и разработке вопросов	<ol style="list-style-type: none"> 1. Исследование существующих архитектур нейронных сетей для классификации изображений; 2. Исследование существующих методов сегментации изображений; 3. Выбор оптимальной архитектуры сверточной нейронной сети для решения поставленной задачи; 4. Проектирование и разработка программного продукта, для решения задачи выделения и классификации рукописных символов на машиночитаемых бланках.
Перечень графического материала	Схема основных блоков разрабатываемой системы, UML диаграмма основных классов программы.

Консультанты по разделам выпускной квалификационной работы	
Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Доцент каф. менеджмента, к.и.н., Данков Артем Георгиевич
Социальная ответственность	Ассистент каф. экологии и безопасности жизнедеятельности, Акулов Петр Анатольевич
Раздел на иностранном языке	Старший преподаватель каф иностранного языка института кибернетики, Морозов Василий Сергеевич
Названия разделов, которые должны быть написаны на русском и иностранном языках:	
Объект и методы исследования (Review of recognition and segmentation algorithms)	

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	03.09.2016
---	------------

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Болотова Юлия Александровна	к.т.н		03.09.2016

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ИМ5Б	Попова Екатерина Сергеевна		03.09.2016

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт кибернетики
Направление подготовки 090402 Информационные системы и технологии
Уровень образования магистр
Кафедра информационных систем и технологий
Период выполнения _____ (осенний / весенний семестр 2016/2017 учебного года)

Форма представления работы:

Магистерская диссертация

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:	06.06.2017 г.
--	---------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
20.05.2017 г.	Основная часть	60
26.05.2017 г.	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	15
28.05.2017 г.	Социальная ответственность	10
05.06.2017 г.	Обязательное приложение на иностранном языке	15

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Болотова Юлия Александровна	к.т.н		03.09.2016

СОГЛАСОВАНО:

Зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
ИСТ	Мальчуков Андрей Николаевич	к.т.н., доцент		03.09.2016

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕ-
РЕЖЕНИЕ»**

Студенту:

Группа	ФИО
8ИМ5Б	Попова Екатерина Сергеевна

Институт	Кибернетики	Кафедра	Информационных систем и технологий
Уровень образования	Магистр	Направление/специальность	090402 «Информационные системы и технологии»

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	...
2. <i>Нормы и нормативы расходования ресурсов</i>	...
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	...

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. <i>Оценка коммерческого и инновационного потенциала НТИ</i>	...
2. <i>Определение ресурсной, финансовой, экономической эффективности</i>	...

Перечень графического материала (с точным указанием обязательных чертежей):

1. <i>График проведения и бюджет НТИ</i>
2. <i>Оценка ресурсной, финансовой и экономической эффективности НТИ</i>

Дата выдачи задания для раздела по линейному графику	
---	--

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Данков А. Г.	К.И.Н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ИМ5Б	Попова Екатерина Сергеевна		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»**

Студенту:

Группа	ФИО
8ИМ5Б	Попова Екатерина Сергеевна

Институт	Институт кибернетики	Кафедра	Информационных систем и технологий
Уровень образования	Магистр	Направление/специальность	090402 «Информационные системы и технологии»

Исходные данные к разделу «Социальная ответственность»:

1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	<i>Разработка алгоритма и программного обеспечения сегментации и распознавания машиночитаемых бланков на основе сверточной нейронной сети</i>
--	---

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

<p>1. Производственная безопасность</p> <p>1.1. Анализ выявленных вредных факторов при разработке и эксплуатации проектируемого решения в следующей последовательности:</p> <ul style="list-style-type: none"> – физико-химическая природа вредности, её связь с разрабатываемой темой; – действие фактора на организм человека; – предлагаемые средства защиты; <p>1.2. Анализ выявленных опасных факторов при разработке и эксплуатации проектируемого решения в следующей последовательности:</p> <ul style="list-style-type: none"> – электробезопасность (в т.ч. статическое электричество, молниезащита – источники, средства защиты); – пожаровзрывобезопасность (причины, профилактические мероприятия, первичные средства пожаротушения). 	<p><i>1. Вредные факторы, сопутствующие разработке и использованию программного продукта:</i></p> <p><i>1.1. Вредные факторы:</i></p> <ul style="list-style-type: none"> - микроклимат - недостаточная освещенность рабочей зоны; <p><i>1.2. Опасные факторы:</i></p> <ul style="list-style-type: none"> - опасность поражения электрическим током; - опасность возникновения пожара
<p>2. Экологическая безопасность:</p> <ul style="list-style-type: none"> – защита селитебной зоны – анализ воздействия объекта на литосферу (отходы); – разработать решения по обеспечению экологической безопасности со ссылками на НТД по охране окружающей среды. 	<p><i>2. Программный продукт, разработанный непосредственно в ходе выполнения магистерской диссертации, не наносит вреда окружающей среде.</i></p> <p><i>Средства, необходимые для разработки и эксплуатации программного комплекса могут наносить вред окружающей среде.</i></p> <p><i>Вследствие этого требуется организация утилизации отходов оргтехники..</i></p>

<p>3. Безопасность в чрезвычайных ситуациях:</p> <ul style="list-style-type: none"> – перечень возможных ЧС при разработке и эксплуатации проектируемого решения; – выбор наиболее типичной ЧС; – разработка превентивных мер по предупреждению ЧС; – разработка действий в результате возникшей ЧС и мер по ликвидации её последствий. 	<p>3. Во время разработки программного продукта и его дальнейшего использования типичной ЧС является:</p> <ul style="list-style-type: none"> -пожар; -Действия и меры, направленные на ликвидацию последствий пожара.
<p>4. Правовые и организационные вопросы обеспечения безопасности:</p> <ul style="list-style-type: none"> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны. 	<p>4. Правовые и организационные вопросы для работника, использующего программный продукт.</p> <ul style="list-style-type: none"> -Требования к помещениям для работы с ПЭВМ регламентируются в соответствии с СанПиН 2.2.2/2.4.1340-03. <p>Влияние разработанного программного обеспечения на рабочий процесс:</p> <ul style="list-style-type: none"> -Сокращение времени обработки бланков, что уменьшает нагрузки специалиста, способствует увеличению частоты смены вида деятельности работника, как оператора ПЭВМ, и тем самым снижает влияние психофизических вредных факторов; -Сокращение умственных и зрительных нагрузок на специалиста за счет автоматизации обработки бланков; -Снижение общего времени нагрузок на специалиста, что уменьшает влияние вредных и опасных факторов производства.

Дата выдачи задания для раздела по линейному графику	
---	--

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Акулов Пётр Анатольевич			

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ИМ5Б	Попова Екатерина Сергеевна		

РЕФЕРАТ

Магистерская диссертация содержит 113 с., 26 рис., 34 табл., 32 источников.

Ключевые слова: информационная система, искусственные нейронные сети, сверточные нейронные сети, алгоритм обратного распространения ошибки, тестовая выборка, обучающая выборка, формальный нейрон.

Объект исследования: Алгоритмы сегментации и распознавания объектов на изображении.

Цель работы: Разработка алгоритма и программного обеспечения сегментации и распознавания машиночитаемых бланков на основе сверточной нейронной сети.

В процессе работы проводились исследования архитектур нейронных сетей, анализ алгоритмов обучения. Были изучены алгоритмы сегментации и распознавания изображений рукописных символов.

В результате исследования была выделена наиболее подходящая нейросетевая архитектура для реализации поставленной задачи, была спроектирована архитектура программы, реализованы классы и методы необходимые для обучения и функционирования сверточной сети. Были реализованы алгоритмы сегментации текстовых изображений выбранные для решения поставленной задачи.

Область применения: распознавание бланков ЕГЭ.

ОПРЕДЕЛЕНИЯ

Искусственная нейронная сеть – математическая модель эмитирующая организацию и функционирование биологических нейронных сетей.

Формальный нейрон – математическая модель биологической нервной клетки.

Функция активации – функция, вычисляющая выходной сигнал искусственного нейрона.

Обучающая выборка (training set) – набор данных, который используется для обучения сети. Представляет собой набор объектов, для каждого из которых априорно известно, к которому из нескольких классов он принадлежит.

Проверочная выборка (validation set) – набор данных, который используется в процессе обучения для оценки качества обучения.

Тестовая выборка (test set) – набор данных, который используется для оценки качества работы сети после завершения обучения.

ЕГЭ – единый государственный экзамен.

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

ОС – Операционная система.

ПО – программное обеспечение.

БД – База данных.

ООП – Объектно-ориентированное программирование.

MNIST database (Mixed National Institute of Standards and Technology database) – база изображений рукописных цифр.

FCNN – (Fully connected neural network) полносвязная нейронная сеть.

АЕ – (Autoencoder) автоэнкодер.

RBM – (Restricted Boltzmann machine) машина Больцмана.

CNN – (Convolutional neural network) сверточная нейронная сеть.

RELU – (Rectified linear unit) выпрямленная линейная функция активации.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	16
1. ОБЗОР АЛГОРИТМОВ РАСПОЗНАВАНИЯ И СЕГМЕНТАЦИИ.....	18
1.1. Искусственные нейронные сети	18
1.1.1. Модель формального нейрона	20
1.1.2. Функция активации.....	21
1.1.3. Функция потерь	25
1.1.4. Алгоритм обучения нейронной сети.....	25
1.2. Сверточные нейронные сети.....	28
1.2.1. Архитектура сверточной нейронной сети	29
1.2.2. Сверточный слой.....	30
1.2.3. Подвыборочный слой	31
1.2.4. Полносвязанный слой.....	32
1.2.5. DropOut слой.....	33
1.2.6. Преимущества и недостатки сверточных сетей.....	33
1.3. Алгоритмы сегментации.....	34
1.3.1. Выделение строк	34
1.3.2. Выделение символов.....	36
2. ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ.....	40
2.1. Обучающая выборка MNIST.....	40
2.2. Обучающая выборка рукописных символов русского алфавита	41
2.3. Подбор архитектуры сверточной нейронной сети.....	44
2.3.1. Обзор библиотек машинного обучения.....	44
2.3.2. Реализация простой нейронной сети на Keras	46
2.3.3. Реализация сверточной нейронной сети на Keras	52

2.4.	Обзор основных блоков разрабатываемой системы.....	58
2.5.	Проектирование логической модели системы	60
3.	РЕЗУЛЬТАТЫ ПРОВЕДЕННОГО ИССЛЕДОВАНИЯ.....	63
4.	ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ.....	65
4.1.	Организация и планирование работ	65
4.1.1.	Продолжительность этапов работ	66
4.1.2.	Расчет нарастания технической готовности работ	69
4.2.	Расчет сметы затрат на разработку ПО и проведение эксперимента ...	71
4.2.1.	Расчет затрат на материалы	71
4.2.2.	Расчет основной заработной платы	71
4.2.3.	Расчет отчислений от заработной платы	72
4.2.4.	Расчет затрат на электроэнергию	73
4.2.5.	Расчет амортизационных расходов	74
4.2.6.	Расчет прочих расходов.....	74
4.2.7.	Расчет общей себестоимости разработки	74
4.2.8.	Прибыль	75
4.2.9.	НДС	75
4.2.10.	Цена разработки НИР	75
4.3.	Оценка экономической эффективности проекта	75
4.3.1.	Оценка научно-технического уровня НИР	76
5.	СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ	81
5.1.	Введение.....	81
5.2.	Производственная безопасность	81
5.2.1	Вредные производственные факторы	82

5.2.1.1 Повышенная или пониженная температура воздуха рабочей среды	82
5.2.1.2 Недостаточная освещенность рабочей зоны	84
5.2.2 Опасные производственные факторы	86
5.2.2.1 Опасность поражения электрическим током	86
5.2.2.2 Опасность возникновения пожара.....	87
5.2.3 Мероприятия и рекомендации по устранению и минимизации.....	87
5.3 Экологическая безопасность.....	89
5.4 Безопасность в чрезвычайных ситуациях.....	90
5.5 Правовые и организационные вопросы обеспечения безопасности	93
5.5.1 Правовые нормы трудового законодательства для рабочей зоны оператора ПЭВМ.....	93
5.5.2 Влияние разработанного программного обеспечения на рабочий процесс	94
ЗАКЛЮЧЕНИЕ	96
СПИСОК ПУБЛИКАЦИЙ СТУДЕНТА.....	97
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ.....	98
Приложение А	103

ВВЕДЕНИЕ

На сегодняшний день существует множество направлений науки и техники, которые в значительной степени ориентированы на развитие систем, анализирующих информацию, представленную в виде изображений. Задача обработки и распознавания изображений относится к разряду трудно формализуемых задач, и является одной из наиболее важных на сегодняшний день.

Для решения данной задачи в последнее время стали активно разрабатываются различные архитектуры нейронных сетей, которые дают значительно более точные результаты распознавания по сравнению с другими алгоритмами.

В рамках данной работы была выбрана архитектура сверточных нейронных сетей, как одна из наиболее подходящих для решения поставленной задачи. Данная архитектура нацелена на эффективное распознавание изображений, она хорошо зарекомендовала себя в решении подобных задач. Также эта архитектура нейронных сетей входит в состав технологий глубокого обучения.

Целью данной работы является разработка алгоритма и программного обеспечения сегментации и распознавания машиночитаемых рукописных бланков на основе сверточной нейронной сети.

Для решения поставленной задачи необходимо решить следующие задачи:

- Исследование существующих архитектур нейронных сетей для классификации изображений;
- Исследование существующих методов сегментации изображений;
- Выбор оптимальной архитектуры сверточной нейронной сети для решения поставленной задачи;
- Определение оптимальных параметров сети;

- Проектирование и разработка программного продукта для решения задачи выделения и классификации рукописных символов на машиночитаемых бланках.

Объектом исследования в данной работе выступают алгоритмы сегментации и распознавания объектов на изображениях, применяемые для решения задачи выделения и распознавания рукописных символов.

Предметом исследования является задача разработки программного продукта, реализующего алгоритм сегментации и распознавания на основе сверточной нейронной сети для решения задачи классификации рукописных символов.

1. ОБЗОР АЛГОРИТМОВ РАСПОЗНАВАНИЯ И СЕГМЕНТАЦИИ

Распознавание машиночитаемых бланков на сегодняшний день является актуальным направлением как научной, так и технической деятельности, и широко освещается в специализированной литературе. В литературных источниках [1, 4, 2] затронуты основные моменты проектирования нейронных сетей для различных задач науки и техники.

В литературных источниках [10, 14] приводится обоснование актуальности использования сверточных нейронных сетей для распознавания изображений. Ключевым моментом в разработке нейронных сетей является выбор алгоритма обучения. В [4, 5, 6] рассматривается метод обучения сетей с помощью обратного распространения ошибки и способы его организации по отношению к разным архитектурам нейронных сетей.

В литературе [3, 7, 8] доступна информация об основных методах цифровой обработки изображений, с целью повышения эффективности последующей сегментации.

В источниках [15, 16], приводится обзор основных алгоритмов сегментации для разных типов изображений и способы их реализации.

1.1. Искусственные нейронные сети

Попытки воспроизвести способность биологической нервной системы обучаться и исправлять ошибки привели к созданию искусственных нейронных сетей. Искусственные нейронные сети представляют собой семейство моделей, построенных на основе организации и функционирования биологических нейронных сетей.

На сегодняшний день нейронные сети успешно применяются для решения таких задач, как классификация и кластеризация многомерных данных, машинное зрение, распознавание речи и др.

Несмотря на наличие различий в архитектурах нейронных сетей, все они обладают некоторыми общими чертами. Во-первых, основу каждой нейронной

сети составляют относительно простые и однотипные элементы (формальные нейроны), имитирующие работу нейронов головного мозга.

Во-вторых, общей чертой для всех нейронных сетей является возможность параллельной обработки сигналов, достигающейся за счет объединения нейронов в слои. На рисунке 1 представлена структура связей однослойной полносвязанной нейронной сети, где каждый нейрон из предыдущего слоя связан с каждым нейроном из следующего слоя.

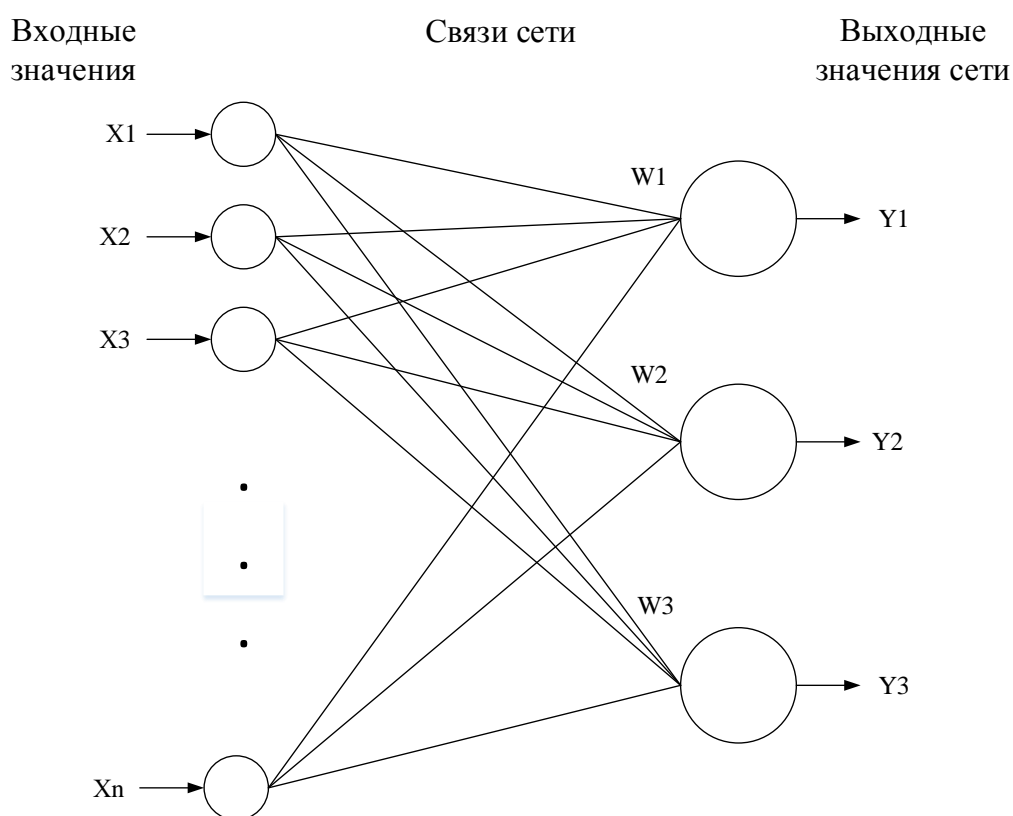


Рисунок 1 – Структура связей полносвязанной нейронной сети

В теории, количество слоев и нейронов в слое может быть выбрано произвольным образом, но на практике оно ограничивается ресурсами компьютера, на котором будут производиться вычисления и временем, за которое сеть должна производить необходимые вычисления. Однако, чем сложнее нейронная сеть, тем более масштабные задачи она может решать.

Процесс функционирования нейронной сети зависит от величин синаптических связей (весов сети), поэтому, после определения структуры

нейронной сети для решения какой-либо задачи необходимо найти оптимальные значения всех настраиваемых параметров. Этот процесс называется обучением нейронной сети, от того, насколько точно будут подобраны значения параметров (весов), зависит способность сети решать поставленные перед ней задачи.

1.1.1. Модель формального нейрона

В основе любой нейронной сети лежит модель формального нейрона (рис. 2), который является основной функциональной единицей нейронной сети.

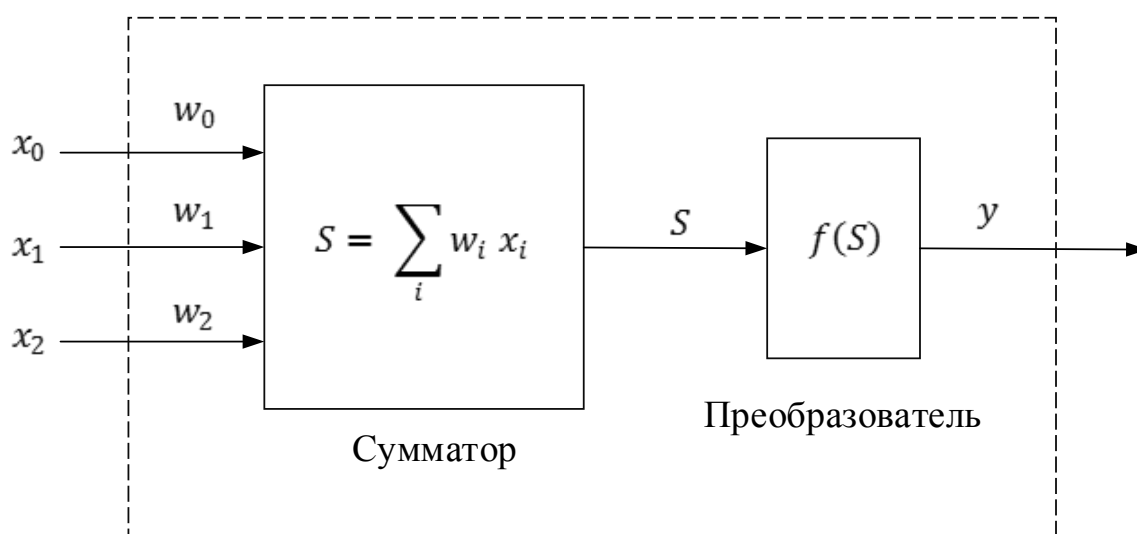


Рисунок 2 – Модель формального нейрона

В этой модели нейрон получает входные сигналы ($x_0 \dots x_n$), которые проходят через связи-синапсы, вес которых ($w_0 \dots w_n$ соответственно) эмулируют различную пропускную способность синапсов в естественных нейронах. После прохождения входных сигналов по синапсам они обрабатываются сумматором, задачей которого является получение линейной комбинации всех входных сигналов, в свою очередь полученная линейная комбинация служит аргументом для функции-преобразователя. Именно эта функция определяет значение выходного сигнала нейрона, который затем посылается на единственный выход нейрона – аксон. Таким образом искусственные нейроны объединяют в сети путем соединения выходов одних нейронов с входами других.

Следовательно, формальный нейрон состоит из элементов 3 типов [4]:

- умножители (синапсы, веса), которые характеризуют силу связи между двумя нейронами;
- сумматор, который выполняет сложение входных сигналов нейрона, перед этим умноженных на соответствующие веса связей.
- преобразователь, который реализует функцию одного аргумента. Эта функция называется функцией активации или передаточной функцией нейрона.

Исходя из полученного описания, можно представить математическую модель нейрона следующим образом:

$$y = f(S), \quad (1)$$

$$S = \sum_{i=1}^n w_i x_i + b, \quad (2)$$

где x – элементы вектора входных сигналов, w – значения весов связей нейрона, b – смещение нейрона, а y – выходной сигнал нейрона.

1.1.2. Функция активации

Функция активации, представленная в формуле (1) как $f(S)$, определяет выходной сигнал нейрона в зависимости от взвешенной суммы входов S .

Ниже приведены основные виды функций активации [1, 4]:

1. **Пороговая функция активации.** Эта функция описывается следующим образом:

$$f(S) = \begin{cases} 0, & \text{при } S < 0 \\ 1, & \text{при } S \geq 0 \end{cases} .$$

2. **Линейная функция.** Линейная функция описывается следующим образом:

$$f(S) = S$$

3. **Кусочно-линейная функция.** Кусочно-линейная функция описывается следующим выражением:

$$f(S) = \begin{cases} 1, & S \geq \frac{1}{2} \\ |S|, & +\frac{1}{2} > S > -\frac{1}{2} \\ 0, & S \leq -\frac{1}{2} \end{cases} .$$

4. **Сигмоидальная логистическая функция.** Сигмоидальная функция является одной из самых распространенных функций, применяемых в искусственных нейронных сетях. Примером сигмоидальной функции может служить логсигмоидная функция, описываемая следующим выражением:

$$f(S) = \frac{1}{1 + e^{(-aS)}} ,$$

где, a – параметр наклона, изменение которого позволит построить функцию с различной крутизной наклона. Область значений сигмоидальной функции лежит в диапазоне от 0 до 1.

Однако иногда требуется функции активация, имеющая область значений от -1 до +1, в этом случае функция активации должна быть симметрично относительно начала координат. Тогда пороговую функцию можно определить следующим образом:

$$f(S) = \begin{cases} 1, & S > 0 \\ 0, & S = 0 \\ -1, & S < 0 \end{cases} .$$

Данная функция называется *сигнум*, в непрерывном дифференцируемом виде она будет иметь форму *гиперболического тангенса*:

$$f(S) = \frac{e^{(aS)} - e^{(-aS)}}{e^{(aS)} + e^{(-aS)}} = \text{Atanh}(aS) ,$$

где $f(S)$ – искомое значение элемента, S – взвешенная сумма входов, a – параметры функции.

Ниже описаны используемые в данной работе функции активации ReLU и Softmax, на данный момент они имеют лучшие результаты при обучении нейронных сетей.

4. Выпрямленная линейная функция активации (rectified linear unit, ReLU). Известно, что нейронные сети способны приблизить сколь угодно сложную функцию, если в них достаточно слоев, и функция активации является нелинейной. Функции активации типа сигмоидной или тангенциальной являются нелинейными, но приводят к проблемам с затуханием или увеличением градиентов в процессе обучения. Однако можно использовать и гораздо более простой вариант – выпрямленную линейную функцию активации. Функция ReLU является выпрямленной линейной функцией и на данный момент считается гораздо более простым и эффективным с точки зрения вычислительной сложности вариантом передаточной функции, она является одним из последних успехов в области методов настройки глубоких нейронных сетей:

$$f(S) = \max(0, S) = \begin{cases} 0, & \text{при } S < 0 \\ S, & \text{при } S \geq 0 \end{cases} .$$

На рисунке 3 представлен график функции ReLU и ее первой производной.

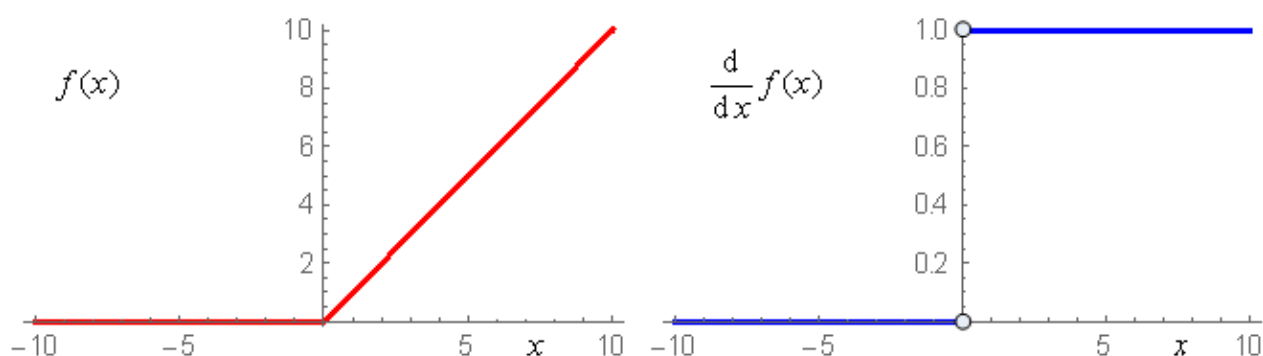


Рисунок 3 – График функции ReLU

Ее производная равна либо единице, либо нулю, и поэтому не может произойти разрастания или затухания градиентов. Более того, использование данной функции приводит к прореживанию весов.

На сегодняшний день существует семейство различных модификаций ReLU, решающих проблемы надёжности этой передаточной функции при прохождении через нейрон больших градиентов: Leaky ReLU, Parametric ReLU, Randomized ReLU.

5. Функция активации *Softmax*. Эта функция активации разработана, чтобы превратить любой вектор с реальными значениями в вектор вероятностей и определяется для i -ого нейрона следующим образом:

$$z_i = \frac{\exp(y_i)}{\sum_{j=1}^n \exp(y_j)} , \quad (3)$$

где z_i – искомое значение выхода i -ого нейрона, y_i – исходное значение выхода i -ого нейрона.

Из формулы (3) видно, что значение каждого выходного нейрона зависит от суммы всех остальных нейронов. Преимущество данной функции заключается в том, что частная производная i -ого нейрона равна:

$$\frac{\partial z_i}{\partial y_i} = z_i(1 - z_i) ,$$

На рисунке 4 представлен график функции *Softmax*.

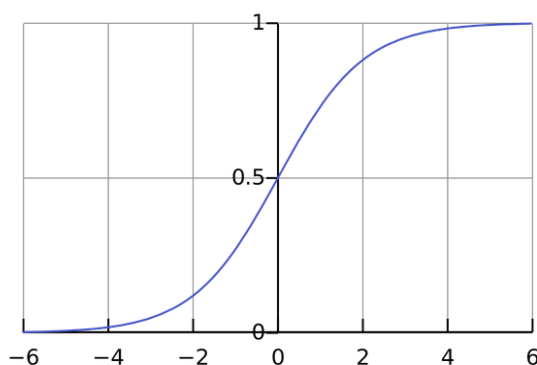


Рисунок 4 – График функции *Softmax*

1.1.3. Функция потерь

В данном разделе будет рассмотрена одна из наиболее часто встречающихся функций потерь, используемая в данной работе – перекрестная энтропия.

Так как в качестве функции активации в выходном слое будет использоваться функция активации softmax, которая преобразует любой входной вектор в вектор вероятностей, то для сравнения двух вероятностных распределений необходимо выбрать корректную меру. В качестве такой меры будет использоваться перекрестная энтропия [10]:

$$C = - \sum_{j=1}^n t_j \log(y_j) ,$$

где t_j – требуемый выход для текущего обучающего примера, y_j – реальный выход нейронной сети.

1.1.4. Алгоритм обучения нейронной сети

Наиболее распространенным алгоритмом обучения нейронных сетей является метод градиентного спуска или как его еще называют метод обратного распространения ошибки, и его модификации. Данный алгоритм относится к методам обучения с учителем [4, 5, 6].

В процессе обучения нейронной сети с применением алгоритма обратного распространения ошибки, на входы сети подаются множества примеров из обучающей выборки. Один цикл предъявления всего набора примеров из обучающей выборки называют эпохой. Процесс обучения происходит от эпохи к эпохе, пока веса и уровни порога (смещения) сети не стабилизируются, а ошибка сети на всем обучающем множестве не сойдется к некоторому заданному минимальному значению.

Алгоритм обратного распространения ошибки может быть реализован тремя способами:

Последовательный режим. Последовательный режим обучения также иногда называют стохастическим градиентным спуском. В этом режиме

изменение весов связей происходит после подачи каждого примера из обучающей выборки.

Пакетный режим. В пакетном режиме обучения корректировка весов связей происходит после подачи на вход сети всех обучающих примеров одной эпохи обучения.

Mini-batch. Между этими двумя видами метода обратного распространения ошибки (градиентного спуска) существует компромисс, называемый иногда «mini-batch». В этом случае корректировка синоптических весов сети происходит после небольшого количества обучающих образцов.

С точки зрения производительности, последовательный режим обучения является более предпочтительным, чем пакетный, так как для хранения каждой синоптической связи требуется меньший объем внутренней памяти. Помимо этого, предъявление обучающих примеров в случайном порядке в процессе обучения для последовательно режима, делает поиск в пространстве весов стохастическим. Таким образом уменьшая возможность остановки алгоритма в точке какого-либо локального минимума.

Алгоритм работы метода обратного распространения ошибки для пакетного режима обучения

Данный метод обучения нейронной сети называют обобщенным дельта-правилом или правилом Error backpropagation. Он был предложен в 1986 г. Румельхартом, Макклеландом и Вильямсом.

Этот алгоритм используется для минимизации отклонения реальных значений выходных сигналов нейронной сети от требуемых. В качестве функции оценки работы ИНС будем использовать метод наименьших квадратов (квадратичная ошибка) [4]:

$$E(w) = \frac{1}{2} \sum_{i,k} (f_{i,k} - y_{i,k}^{(T)})^2 ,$$

где $f_{i,k}$ – значение выходного сигнала k -го выходного нейрона сети при подаче на её входы i -го набора обучающих данных, $y_{j,k}^{(T)}$ – требуемое значение выходного

сигнала k -го выходного нейрона для i -го набора данных для обучения. Суммирование ведется по всем нейронам выходного слоя.

В качестве метода минимизации функции $E(w)$, будем использовать методом градиентного спуска, который обеспечивает коррекцию весовых коэффициентов следующим образом:

$$\Delta w_{ij}^{(q)} = \eta \frac{\partial E}{\partial w_{ij}} ,$$

где $\Delta w_{ij}^{(q)}$ – величина изменения веса связи, соединяющей i -й нейрон $(q-1)$ слоя с j -м нейроном слоя q ; η – коэффициент скорости обучения, $0 < \eta < 1$. Таким образом, вес связи изменяется пропорционально её вкладу в значение ошибки нейрона, для которого эта связь является входной, т.к. частная производная по весу $\partial E / \partial w_{ij}$ показывает зависимость скорости изменения функции ошибки E от изменения этого веса.

Изменение веса связи определяется следующим образом:

$$\Delta w_{ij}^{(q)} = \eta \delta_j x_i ,$$

где δ_j – значение ошибки j -го нейрона в слое q , x_i – значение i -го входного сигнала для j -го нейрона слоя q . Данная формула применима и для настройки смещений нейронов, только вместо x_i необходимо подставить «1».

Значение ошибки нейрона определяется в зависимости от его положения в сети. Для нейронов выходного слоя:

$$\delta_j = (f_{i,k}(S))' (f_{i,k} - y_{i,k}) ,$$

где $y_{i,k}$ – требуемое, а $f_{i,k}$ – фактическое значение выходного сигнала k -го нейрона для i -го набора данных из обучающей выборки, $(f_{i,k}(S))'$ – значение производной активационной функции k -го нейрона для i -го набора обучающих данных. Если нейрон принадлежит одному из скрытых слоев, то

$$\delta_i^{(q)} = (f_i^{(q)}(S))' \sum_j w_{ij} \delta_i^{(q+1)} ,$$

где $\delta_i^{(q)}$ – ошибка i -го нейрона в слое q , $\delta_i^{(q+1)}$ – ошибка j -го нейрона в $(q+1)$ слое, w_{ij} – вес связи, соединяющей эти нейроны, $(f_{i,k}(S))'$ – значение производной активационной функции i -го нейрона слоя q .

Из этого следует, что влияние каждого нейрона на величину ошибок нейронов следующего слоя, пропорционально значению ошибки этого нейрона.

1.2. Сверточные нейронные сети

С появлением больших объемов данных и больших вычислительных возможностей стали активно использоваться нейронные сети. Особую популярность получили сверточные нейронные сети.

Свёрточная нейронная сеть – архитектура искусственных нейронных сетей, предложенная Яном Лекуном, предназначенная для эффективного распознавания изображений.

Данная разновидность нейронных сетей использует некоторые особенности зрительной части коры головного мозга, где были открыты простые клетки, которые реагируют на прямые линии на изображении, расположенных под разными углами, и сложные клетки, реагирующих на активацию определённого набора простых клеток.

Свое название сверточные сети получила из-за присутствия операции свёртки, суть которой заключается в вычисление нового значения текущего пикселя, учитывая значения соседних пикселей. Для вычисления значения используется ядром свертки. Во время вычисления нового значения выбранного пикселя на него накладывается ядро свертки (матрица свертки), соседние пиксели так же накрываются ядром. Далее подсчитывается сумма, где слагаемыми являются произведения значений пикселей на значения ячейки ядра, накрывшей данный пиксель. Получившийся результат суммируется и записывается в аналогичную позицию выходного изображения.

В архитектуру сети заложены априорные знания из предметной области компьютерного зрения:

- пиксель изображения сильнее связан с соседним (локальная корреляция);
- объект на изображении может встретиться в любой части изображения.

Особое внимание сверточные нейронные сети получили после конкурса ImageNet в октябре 2012 года, который был посвящен распознаванию объектов на фотографиях. Победитель данного конкурса - Алекс Крижевский, используя сверточную нейронную сеть, значительно превзошел остальных участников.

1.2.1. Архитектура сверточной нейронной сети

Архитектура сверточной нейронной сети представляет из себя чередование сверточных слоев, субдискретизирующих слоев (подвыборочный слой) и полносвязных слоев (fully-connected layer) на выходе. Все перечисленные виды слоев могут располагаться в нейронной сети произвольным образом.

В свою очередь, сверточный слой представляет собой набор плоскостей, образующийся за счет того, что изображение предыдущего слоя сканируется небольшим окном и пропускается сквозь набор весов, а результат отображается на соответствующий нейрон текущего слоя. Таким образом получается, что каждый нейрон выполняет свертку некоторой области предыдущего слоя. Следовательно, набор плоскостей представляет собой карты особенностей входного изображения, и каждая плоскость находит соответствующие признаки в любом месте предыдущего слоя, который содержит некоторое изображение.

На рисунке 5 изображена стандартная структура сверточной нейронной сети.

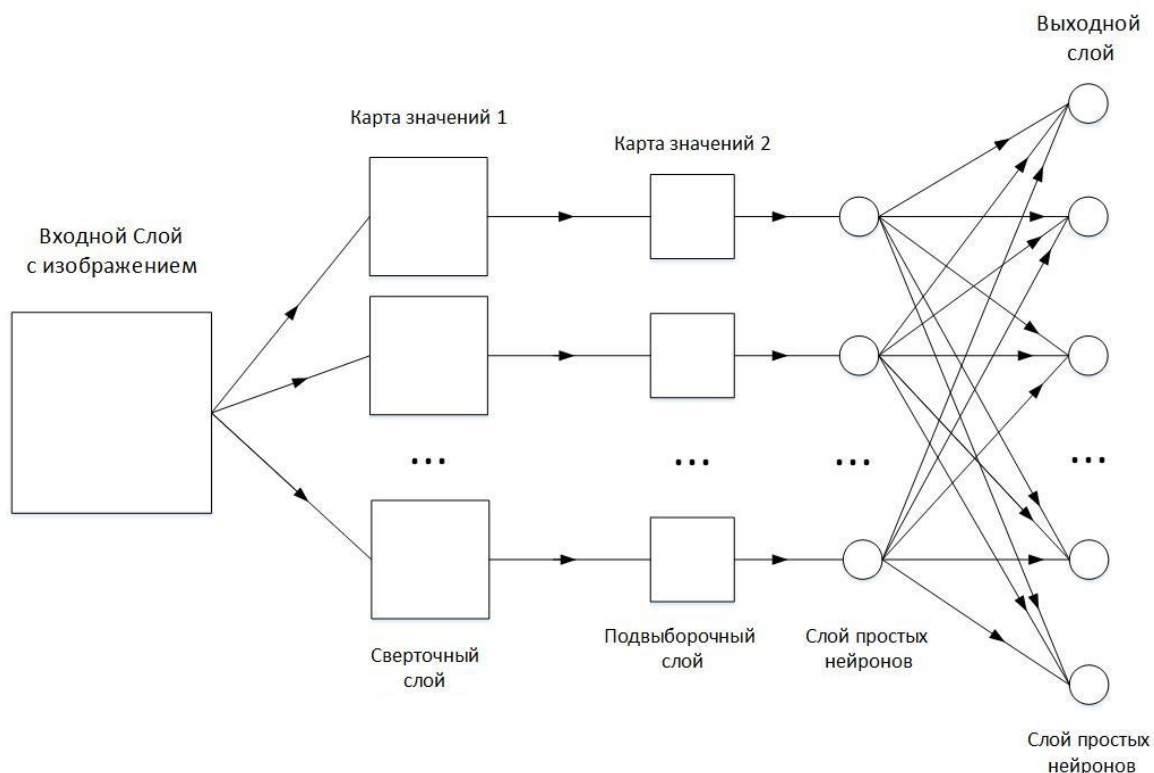


Рисунок 5 – Структура простой сверточной нейронной сети

1.2.2. Сверточный слой

Каждый нейрон находящийся в плоскости сверточного слоя получает свои входные значения от некоторой области предыдущего слоя, то есть изображение, которое подается на вход просматривается небольшим окном и пропускается сквозь набор весов, полученный результат записывается в определенный нейрон сверточного слоя.

Принцип работы сверточного слоя представлен на рисунке 6.

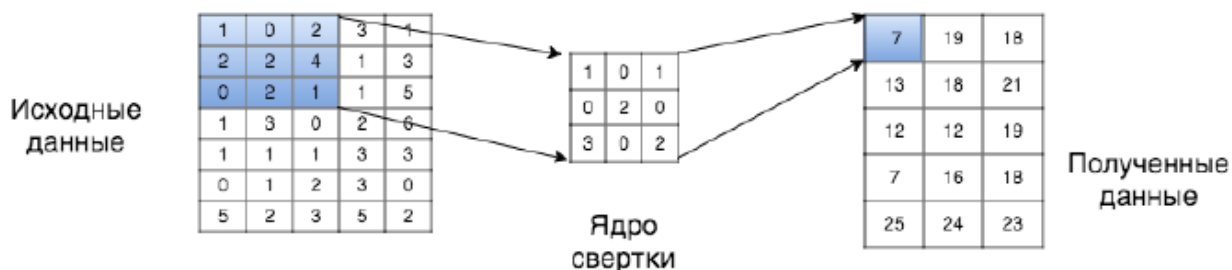


Рисунок 6 – Принцип работы сверточного слоя с ядром свертки 3×3

Таким образом, процесс функционирования нейрона свёрточного слоя $y_k^{(i,j)}$ задается выражением [14]:

$$y_k^{(i,j)} = b_k + \sum_{s=1}^K \sum_{t=1}^K w_{k,s,t} x^{((i-1)+s,(j+t))} ,$$

где, $y_k^{(i,j)}$ – нейрон k -ой плоскости свёрточного слоя, b_k – нейронной смещение k -ой плоскости, K - размер рецептивной области нейрона, $W_{k,s,t}$ – матрица синоптических коэффициентов, x – выходы нейронов предыдущего слоя.

Затем полученный результат свертки каждой локальной области, подается на активационную функцию:

$$out_k^{(i,j)} = f(y_k^{(i,j)}) .$$

Отличительной особенностью свертчного слоя, является то что он немного уменьшает исходное изображение за счет краевых эффектов.

1.2.3. Подвыборочный слой

В современных ИНС используются субдискретизирующие (subsampling) слои, выполняющие уменьшение размерности входной карты признаков. Это можно делать разными способами, но чаще всего, для этого используется метод выбора максимального элемента (maxpooling) в окрестности текущего элемента. Вся карта признаков разделяется на ячейки, из которых выбираются максимальные по значению. Использование maxpooling позволяет сделать сеть инвариантной к масштабным преобразованиям.

На рисунке 7 показан пример субдискретизирующего слоя с методом выбора максимального элемента и размером окна 2×2 .



Рисунок 7 – Принцип работы нейронов подвыборочного слоя

Этот слой обычно идет сразу за свёрточным, он также состоит из плоскостей, как правило, имеет такое же количество плоскостей, что и в предыдущем сверточном слое.

1.2.4. Полносвязанный слой

Слой, в котором каждый нейрон соединен со всеми нейронами на предыдущем уровне, причем каждая связь имеет свой весовой коэффициент. На рисунке 8 показан пример полносвязного слоя.

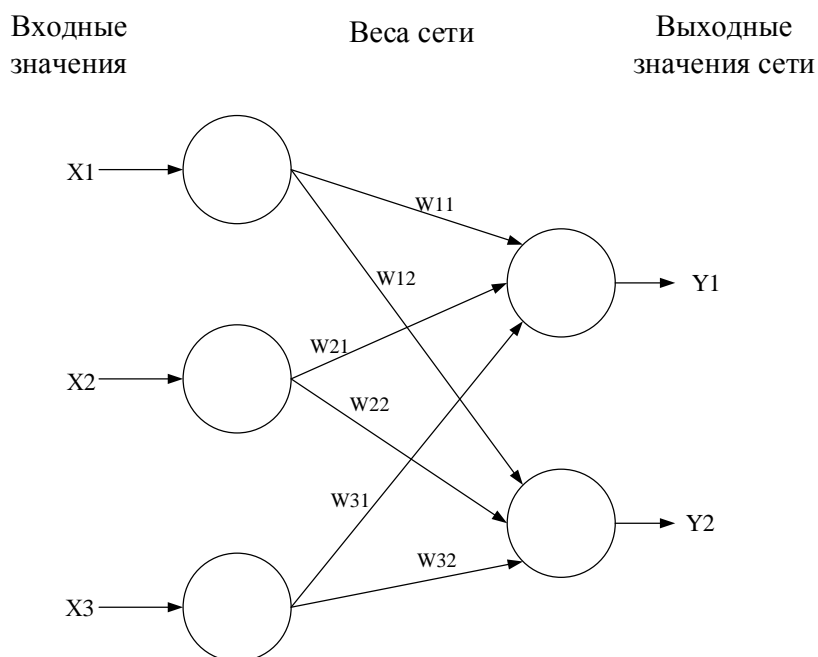


Рисунок 8 – Полносвязный слой

где w_{ij} – весовые коэффициенты связей, x_i – входные значения сигналов, y_i – выходные значения сети.

1.2.5. DropOut слой

Dropout является простым и эффективным методом регуляризации и заключающийся в том, что в процессе обучения сети из её совокупной топологии многократно случайным образом выделяется подсеть, и очередное обновление весов происходит только в рамках выделенной подсети. Каждый нейрон исключается из совокупной сети с некоторой вероятностью, которая называется коэффициентом dropout.

1.2.6. Преимущества и недостатки сверточных сетей

Преимущества [5, 10]:

- хорошо справляется с задачей классификации и распознаванию изображений;
- меньшее количество настраиваемых параметров, по сравнению с полносвязной нейронной сетью;
- более высокая устойчивость к повороту и сдвигу распознаваемого изображения, по сравнению полносвязными нейронными сетями (типа персептрона).

К недостатком сверточной нейронной сети можно отнести большое количество настраиваемых гиперпараметров, к которым относятся:

- количество сверточных и подвыборочных слоев;
- количество карт признаков в сверточных и подвыборочных слоях;
- размерность матрицы свёртки;
- функция активации нейронов;
- скорость обучения сети.

Все выше перечисленные параметры существенно влияют на результат работы сверточной нейронной сети, и как правило выбираются эмпирически. Для распространенных задач существуют некоторые выверенные и прекрасно

работающие конфигурации сетей, но для новых задач подбор гиперпараметров осуществляется опытным путем, так как не существует общего правила для их определения.

1.3. Алгоритмы сегментации

Сегментация строки на символы является одним из важнейших этапов в процессе оптического распознавания символов, в частности, при оптическом распознавании изображений документов. Сегментацией строки называется декомпозиция изображения, содержащего последовательность символов, на фрагменты, содержащие отдельные символы.

Важность сегментации обусловлена тем обстоятельством, что в основе большинства современных систем оптического распознавания текста лежат классификаторы (в том числе — нейросетевые) отдельных символов, а не слов или фрагментов текста. В таких системах ошибки неправильного проставления разрезов между символами как правило являются причиной львиной доли ошибок конечного распознавания.

В данной работе сегментирование текстового изображения разделен на несколько этапов, с целью улучшения производительности разрабатываемой системы и повышения точности определения областей расположения отдельных символов.

Сегментацию изображения текста будем проводить в два этапа:

- выделение строк – исходное изображение текста необходимо "разрезать" на полосы-строки нужной ширины.
- сегментация символов – в изображении слова проводим границы символов.

1.3.1. Выделение строк

В данном случае полагается, что исходное изображение текста правильно ориентированно, т.е. строки ровные и картинка не повёрнута относительно наблюдателя. Такое предположение сделано на том основании, что оцифровка

исходного документа производилась с помощью сканирующего устройства, следовательно это позволяет избежать непреднамеренного наклона изображения и разности освещенности, в отличии от использования ручной съемки с помощью фотокамеры.

В связи с этим, для выделения на исходном изображении строки был выбран пороговый алгоритм сегментации, основанный на использовании гистограммы [12, 15].

Данные алгоритмы эффективны, если сравнивать с другими методами сегментации, из-за того, что они делают всего один проход по пикселям. В этом методе гистограмма вычисляется по всем пикселям изображения и её минимумы и максимумы используются, чтобы найти кластеры на изображении. Цвет или яркость пикселя могут быть использованы при сравнении.

В данном случаи метод гистограмм используется для сегментации изображения на строки.

Данный метод предполагает построение гистограммы для черных пикселей бинарного изображения. В таком случаи по оси Y будет располагаться шкала распределения пикселей по количеству, а на оси X размешены номера строк.

Алгоритм основан на том что количество черных писклей в межстрочных интервалах существенно меньше чем в текстовых строках. Основываясь на этом предположении, определим каким должно быть наименьшее количество черных пикселей в строке, чтобы отнести ее к текстовой строке. Рассчитаем значение по формуле.

Сначала для всех пикселей строки исходного изображения находим количество черных пикселей:

$$s_i = s_i(B) = \sum_{j=1}^n b_{ij}$$

Затем определяем среднее количество черных пикселей для всего изображения:

$$s(B) = \frac{1}{m} \sum_{j=1}^m s_j(B)$$

Теперь исходя из полученного значения определим минимальный порог количества черных пикселей в текстовой строке:

$$N = k * s(B)$$

где k – коэффициент, принятый за 0.1.

Следовательно, используя найденный порог разделим все изображение на строки.

Работа алгоритма сегментации строк заключается в последовательном просмотре массива содержащего количество черных пикселей для каждой строки сравнение их с минимальным количеством N и выявлении множества пар индексов (s_i^1, s_i^2) строк, соответствующих границам печатных строк.

На рисунке 9 представлены результаты работы программы после построения гистограммы для фрагмента текста.



Рисунок 9 – Построение гистограмма для фрагмента изображения

На данном изображении явна выделяются участки гистограммы, где количество черных пикселей значительно больше чем в других областях. Такие участки будут расценены программой как текстовые строки.

1.3.2. Выделение символов

Для выделения одиночных символов на изображении, был выбран метод связанных компонент [12], как наиболее подходящий для решения поставленной задачи.

Под выделением связанных компонент понимают присвоение уникальной метки каждому объекту изображения. При следующем анализе данные метки

служат в качестве идентификаторов при обращении к объектам. Это делает операцию выделения связанных компонент неотъемлемой частью почти всех приложений распознавания образов и компьютерного зрения.

Поэтому, перед тем как классификатор сможет определить какой объект находится на изображении, смежные пиксели одного объекта должны быть идентифицированы и промаркированы. Каждая выделенная группа пикселей соответствует одному объекту на изображении.

Метод связанных компонент хорошо подходит для сегментирования слов на отдельные символы, так как он предполагает выделение на исходных изображениях непрерывных участков с близкими по некоторым свойствам наборами точек.

Основные понятия:

Связанная область (участок) – набор точек на изображении, в котором любые две точки соединены друг с другом через последовательность соседей.

Виды связности:

- 4-связность - соседями точки считаются точки, имеющие с данной точкой одну общую сторону.
- 8-связность – соседями точки считаются точки, имеющие с данной точкой общую сторону или общий угол.

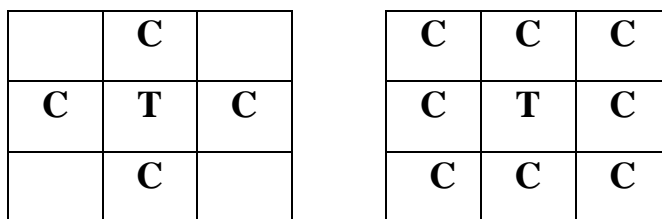


Рисунок 10 – «4-связанность» и «8-связанность»

где С – сосед, Т – текущая (данная) точка.

Алгоритмы выделения связанных компонент можно разделить на следующие виды:

- многопроходные;
- двухпроходные;

- однопроходные;
- алгоритмы с использованием иерархических структур представления изображения
- параллельные алгоритмы.

В данной работе будет рассмотрен двухпроходной алгоритм как наиболее подходящий для обработки монохромных изображений и наименее затратным по времени из всех выше предложенных алгоритмов.

Для описания алгоритма введем некоторые понятия. Обозначим через I матрицу изображения. Если $I(i, j)=0$, то пиксель является фоновым. Если $I(i, j)=1$, то пиксель принадлежит объекту. Через L обозначим двумерную матрицу, скан-маска, которая используется для хранения информации о метках и имеющую размеры, равные размерам изображения.

Двухпроходной алгоритм последовательно сканирует изображение. Первый проход по изображению осуществляется из верхнего левого угла, слева на право и сверху вниз. На внешнем цикле – перебор строк, на внутреннем – перебор столбцов строки, и анализируются только соседи сверху и слева. Тесть при первом проходе мы будем анализировать уже известные значения пикселей, относительно текущего пикселя.

Каждый раз при обнаружении черного пикселя его соседи, принадлежащие скан-маске, исследуются для определения метки, которая будет присвоена рассматриваемому пикселю. Если в скан-маске содержатся только белые пиксели, то текущий пиксель получает новую промежуточную метку. Если скан-маска содержит только один черный пиксель, то текущий пиксель получает метку соседа. Если скан-маска содержит несколько черных пикселей, то их промежуточные метки являются эквивалентными, среди них выбирается метка с наименьшим значением и пикселю присваивается значение выбранной метки.

После окончания первого прохода каждый объектный пиксель получает временную метку, на втором проходе значение метки уточняется. Второй скан

проход осуществляется в противоположенном направлении, снизу-вверх и с права на лево. Во время второго прохода, алгоритм анализирует не только соседей сверху и слева, а все пиксели, имеющие с данным общую сторону или общий угол.

Не смотря на то что метод связанных компонент не является универсальным алгоритмом выделения символов в строке, в большинстве случаев он будет справляться с поставленной задачей.

После выделения, связанных компонент могут образоваться мелкие области, не представляющие интереса, к примеру остаточные шумы на изображении. Для устранения таких объектов, определим минимальный размер элемента – эталон, в случаи если объект меньше эталона, он не засчитывается за символ. Определение минимального размера происходит по формуле:

$$S_{\min} = S_{\max} * 0.2 ,$$

где $S_{\max} = a * b$, a и b соответственно ширина и высота самого большого объекта.

Данный алгоритм хорошо подходит для сегментации рукопечатного текста, который в отличии от рукописного не имеет связей между буквами и так как в данная работа нацелена на обработку именно таких текстов, он хорошо подходит для решения поставленной задачи и в большинстве случаев дает корректные результаты сегментации, которые играют важную роль в последующем распознавании символов.

2. ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ

В качестве программной среды выбран продукт компании Microsoft Visual Studio 2015 и язык программирования C#.

Дополнительным средством для проведения экспериментов были выбраны язык программирования Python, так как это высокоуровневый язык программирования общего назначения, который имеет хорошую совместимость с различными фреймворками машинного обучения.

2.1. Обучающая выборка MNIST

В качестве одного из наборов, данных для обучения и тестирования сети будет использоваться база изображений рукописных цифр MNIST [11]. Изображения в данной базе имеют разрешение 28x28 пикселей и хранятся в формате оттенков серого, следовательно, каждое значение пикселя лежит в диапазоне от 0 (представляет белый цвет) до 255 (представляет черный цвет). Цифры отцентрированы на изображении. Данный набор разбит на две части: тренировочную и тестовую выборки состоящих из 60000 и 10000 изображений соответственно. На рисунке 9 показаны первые восемь изображений в обучающем наборе.

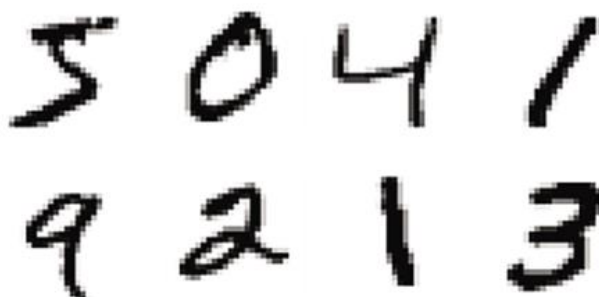


Рисунок 11 – Первые восемь обучающих изображений MNIST

Формат файла обучающих пиксельных данных MNIST имеет начальное целочисленное значение (32 бита), равное 2051, за которым следует количество изображений (целочисленное значение), затем количество строк и количество

столбцов (целочисленные значения), потом 60 000 изображений по 28×28 пикселей = 47 040 000 байтовых значений.

Обучающие, и тестовые данные хранятся в двух файлах, один файл содержит значения пикселей для изображений, а другой – метки изображений (0–9). Каждый из четырех файлов также содержит заголовочную информацию, и все они хранятся в двоичном формате, сжатом в формате gzip.

2.2. Обучающая выборка рукописных символов русского алфавита

Приведенный выше набор данных MNIST, не содержит в себе рукописных букв, которые необходимы для обучения сети распознавать бланки ЕГЭ. Для этих целей будет использоваться обучающая выборка, содержащая 146324 изображений рукописных символов, включающая в себя цифры и буквы русского алфавита.



Рисунок 12 – Примеры элементов из обучающей выборки

Ниже приведена таблица с частой встречаемостью каждого символа в наборе данных рукописных букв русского алфавита.

Таблица 1. Частота распределения символов

Символ	Кол-во	Символ	Кол-во	Символ	Кол-во	Символ	Кол-во
0	252	Б	2343	Л	3062	Ц	132
1	12803	В	8483	М	2276	Ч	1197
2	4939	Г	2096	Н	4299	Ш	887
3	6028	Д	5594	О	11834	Щ	55
4	6533	Е	11978	П	5484	Ъ	72
5	1852	Ё	12	Р	8776	Ы	3278
6	1152	Ж	806	С	4825	Ь	3744

7	1168	З	1779	Т	8284	Э	149
8	940	И	8291	У	829	Ю	467
9	456	Й	1688	Ф	15	Я	1908
А	6114	К	2934	Х	136		

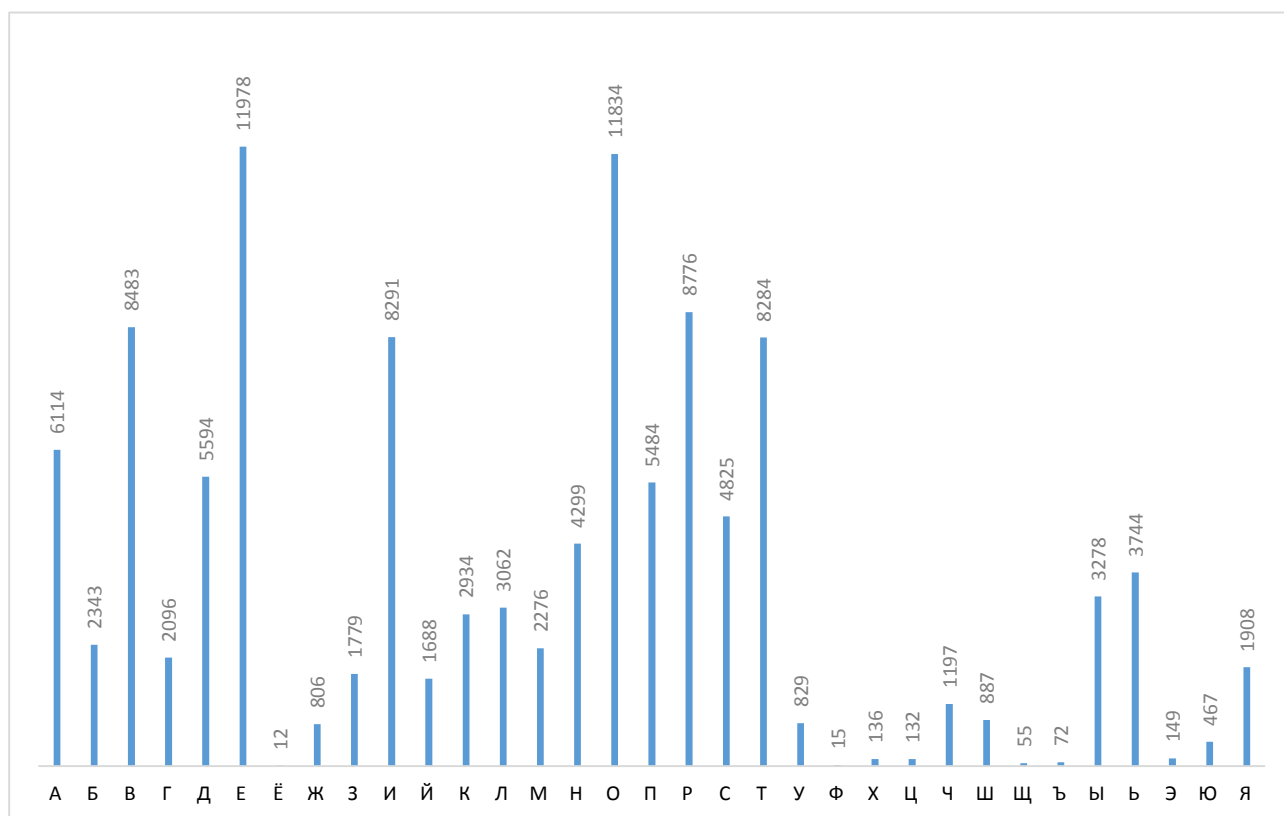


Рисунок 13 – Гистограмма распределения рукописных символов в выборке

Как можно видеть из приведённого выше графика, не смотря на большое количество обучающих примеров в исходном наборе, количество букв в выборке распределено не равномерно, следовательно, было принято решение создать обучающую выборку, содержащую около 100 – 200 примеров для каждого символа. Символы, которые не имели такого количества изображений были размножены за счет имеющихся изображений этих символов. Таблица 2, содержит количество изображений для каждого символа в получившейся выборке.

Таблица 2. Количество символов в сбалансированной выборке

Символ	Кол-во	Метка	Символ	Кол-во	Метка	Символ	Кол-во	Метка
А	209	0	Л	213	11	Ц	119	22
Б	210	1	М	207	12	Ч	197	23
В	209	2	Н	210	13	Ш	210	24
Г	212	3	О	200	14	Щ	100	25
Д	210	4	П	228	15	Ъ	100	26
Е	210	5	Р	211	16	Ы	210	27
Ж	225	6	С	210	17	Ь	210	28
З	211	7	Т	210	18	Э	149	29
И	209	8	У	210	19	Ю	210	30
Й	209	9	Ф	100	20	Я	210	31
К	204	10	Х	122	21			

Размер сбалансированной выборки составил – 6155 изображений, далее выборка была разделена на два набора обучающую и тестовую, в соотношении 9:1 соответственно. В результате размер обучающей выборки составил 5532 примеров, тестовой – 662 примеров.

Изображения в данной базе имеют размер 32x32 пикселей и хранятся в формате оттенков серого, следовательно, каждое значение пикселя лежит в диапазоне от 0 (черный цвет) до 255 (белый цвет), символы центрированы на изображении. Для ускорения работы сети необходимо инвертировать значения пикселей следующим образом 0 – белый цвет, 255 – черный цвет, по следующей формуле:

$$y_i = 255 - x_i$$



Рисунок 14 – Инвертированное изображение

Для ускорения сходимости обучения сети значения входных пикселей изображений нормализуются по формуле:

$$y_i = \frac{x_i}{255}$$

где x_i – значение i -го пикселя изображения из базы, y_i – значение, подаваемое на вход сети.

2.3. Подбор архитектуры сверточной нейронной сети

Большое количество обучающих и тестовых примеров требует достаточно большого времени обучения, поэтому, для подбора наилучшей конфигурации сети было решено воспользоваться готовыми библиотекам, а затем перейти к реализации собственной сверточной сети выбранной архитектуры.

2.3.1. Обзор библиотек машинного обучения

Для подбора наиболее удачных параметров разрабатываемой сети, был произведен обзор существующих библиотек машинного обучения. На основе выбранной библиотеки спроектирована нейронная сеть с различными параметрами, для выявления наиболее подходящей конфигурации сети для решения поставленной задачи.

Существует множество программных средств для решения задач машинного обучения, в таблице 2 приведены наиболее популярные из них, они предоставляют возможности для создания полностью связанных нейросетей (FC NN), сверточных нейронных сетей (CNN), автокодировщиков (AE) и ограниченных машин Больцмана (RBM).

Таблица 3. Обзор библиотек машинного обучения

Название	Языки	ОС	FC NN	CNN	AE	RBM
DeepLearnToolbox	Matlab	Windows, Linux	+	+	+	+

Theano	Python	Windows, Linux, Mac	+	+	+	+
Pylearn2	Python	Linux, Vagrant	+	+	+	+
Deepnet	Python	Linux	+	+	+	+
Torch	Lua, C	Linux, Mac OS X, iOS, Android	+	+	+	+
Darch	R	Windows, Linux	+	–	+	+
Caffe	C++, Python, Matlab	Linux, OS X	+	+	–	–
nnForge	C++	Linux	+	+	–	–
CXXNET	C++	Linux	+	+	–	–
Cuda-convnet	C++	Linux, Windows	+	+	–	–
Cuda CNN	Matlab	Linux, Windows	+	+	–	–

Из ходя из таблицы 3, была выбрана библиотека машинного обучения Theano, как наиболее оптимальная. Библиотека реализована на языке Python, поддерживается на операционных системах Windows, Linux и Mac OS. В состав Theano входит компилятор, который переводит математические выражения, написанные на языке Python в эффективный код на C или CUDA.

Theano предоставляет базовый набор инструментов для конфигурации нейросетей и их обучения. Возможна реализация многослойных полностью связанных сетей (Multi-Layer Perceptron), сверточных нейросетей (CNN), рекуррентных нейронных сетей (Recurrent Neural Networks, RNN),

автокодировщиков и ограниченных машин Больцмана. Также предусмотрены различные функции активации, в частности, сигмоидальная, softmax-функция, RELU, кросс-энтропия.

Для работы с библиотекой машинного обучения Theano библиотека Keras.

Keras – это библиотека, позволяющая на более высоком уровне работать с нейросетями. Она упрощает множество задач, используется в быстрых экспериментах и сильно уменьшает количество однообразного кода. В качестве бекендной библиотеки для вычислений keras может использовать theano и tensorflow, в данной работе в качестве бекенда использовалась библиотека машинного обучения theano.

2.3.2. Реализация простой нейронной сети на Keras

Действительно ли нам нужна такая сложная модель, как сверточные нейронные сети, для получения высокой точности распознавания рукописных символов. В начале реализуем очень простую модель нейронной сети с одним скрытым слоем.

Спроектируем простую полносвязную многослойную модель перцептрона, которая достигает точности распознавания около 90%. Полученные результаты будут использованы в качестве основы для сравнения с более сложной архитектурой сверточных нейронных сетей.

Для задачи распознавания рукописных символов из обучающей выборки MNIST с помощью многослойного перцептрона элементы матрицы входного изображения записываются по строкам, в результате получается вектор x длины $28^2 = 784$ — признаковое описание объекта.

Ниже приведена реализация классической нейронной сети с 3 слоями для распознавания рукописных символов из обучающей выборки MNIST.

Таблица 4: Конфигурация классической нейронной сети

Слой	1	2	3
Тип слоя	Входной	Скрытый	Выходной
Функция активации	ReLU	ReLU	Softmax
Число нейронов	784	784	10
Число связей	–	615 440	7 850
Общее кол-во настраиваемых параметров	623 290		

На рисунке 15 визуализированная архитектура простой нейронной сети для распознавания рукописных цифр из обучающей выборки MNIST.

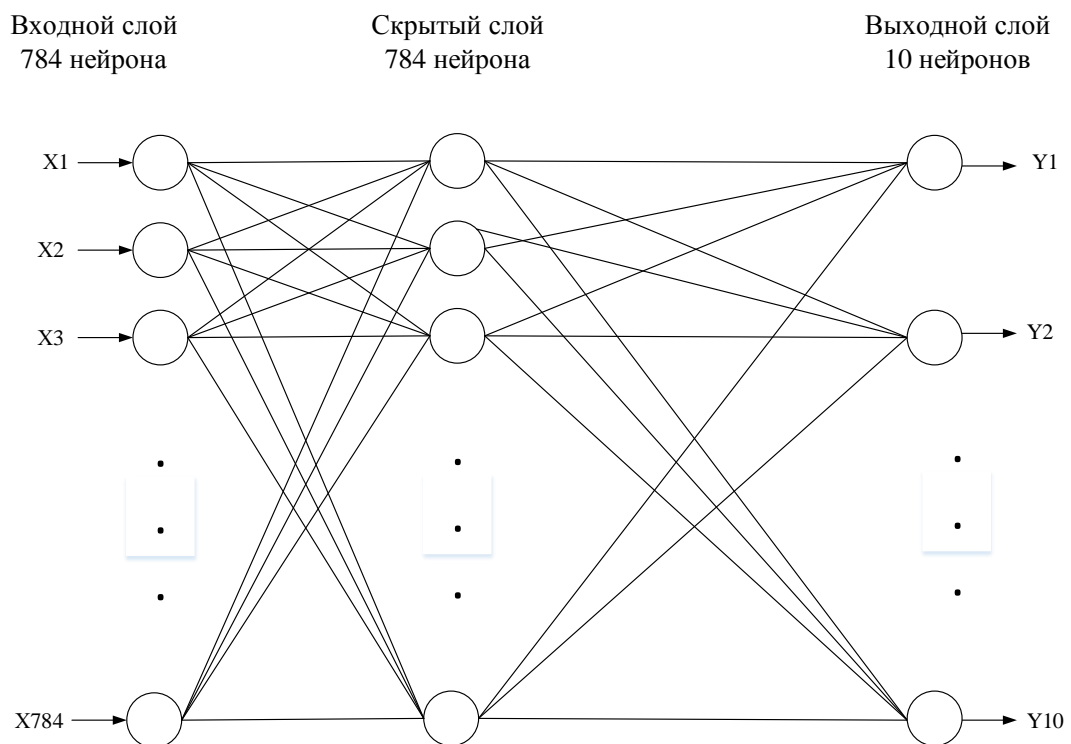


Рисунок 15 – Архитектура нейронной сети для распознавания рукописных цифр

В таблице 5 приведены результаты тестирования классической однослойной нейронной сети для распознавания рукописных символов из обучающей выборки MNIST.

Для обучения сети использовались различные модификации метода градиентного спуска, коррекция весов производилась после 200 элементов из обучающей выборки, такой способ обучения сети, является оптимальным вариант между последовательным (стохастическим) и пакетным обучением.

Таблица 5: Влияние количества эпох на точность распознавания

Размер мини выборки = 200				
Количество эпох	Алгоритмы оптимизации			
	SGD	Adam	AdaGrad	AdaDelta
5	90.80%	98.03%	97.00%	97.00%
10	92.06%	98.23%	98.08%	97.74%
50	93.27%	98.38%	98.21%	98.35%

Таблица 6: Влияние размера мини выборки на точность распознавания

Количество эпох обучения = 5				
Размер мини выборки	Алгоритмы оптимизации			
	SGD	Adam	AdaGrad	AdaDelta
25	94.98%	98.01%	98.95%	98.23%
50	93.58%	97.88%	97.88%	97.72%
100	92.12%	97.85%	97.83%	97.52%
200	90.80%	98.03%	97.00%	97.00%

Дальнейшее увеличение количества эпох обучения не имеет смысла, так как точность на проверочной выборке после 25 эпохи стала колебаться ± 0.02 , это свидетельствует о переобучении сети.

В таблице 7 приведены результаты тестирования описанной выше архитектуры нейронной сети для набора данных рукописных букв. Архитектура сети была изменена следующим образом, входной слой и скрытый слой содержит $32^2 = 1024$ нейронов, выходной слой содержит 32 нейрона – по 1 нейрону на каждый класс (32 буквы – 32 класса).

Таблица 7: Конфигурация классической нейронной сети

Слой	1	2	3
Тип слоя	Входной	Скрытый	Выходной
Функция активации	ReLU	ReLU	Softmax
Число нейронов	1 024	1 024	32
Число связей	–	1 049 600	32 800
Общее кол-во настраиваемых параметров	1 082 400		

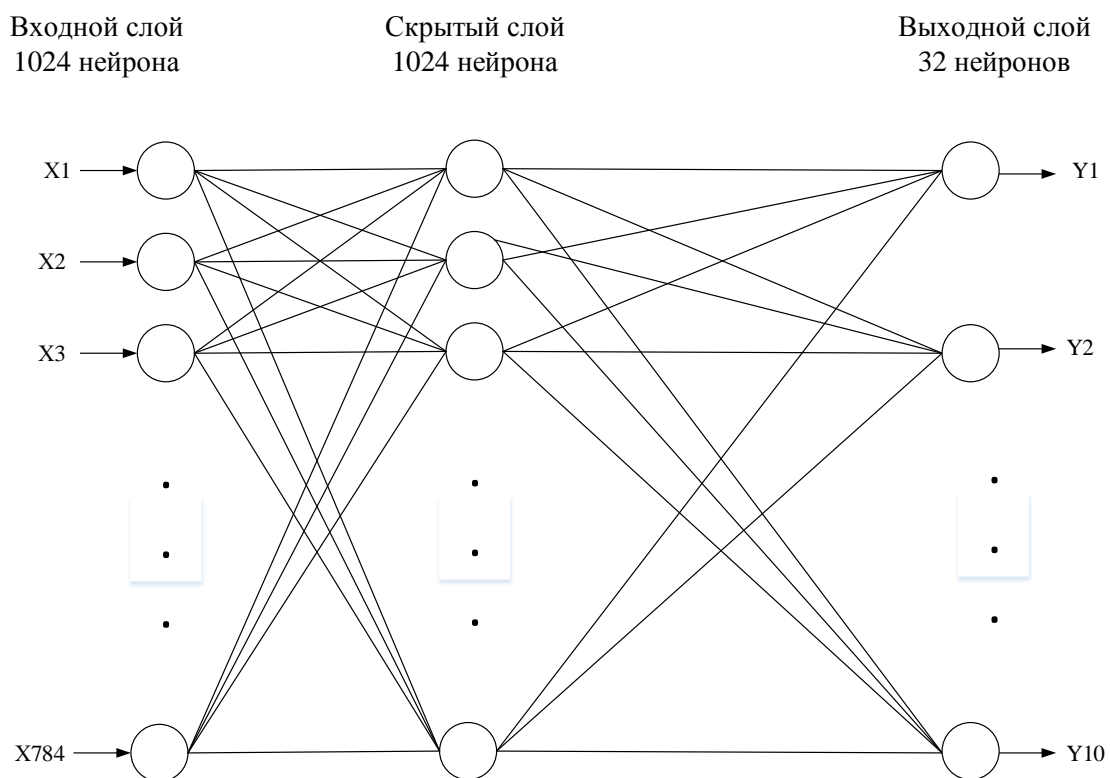


Рисунок 16– Архитектура нейронной сети для распознавания букв

Таблица 8: Влияние количества эпох на точность распознавания

Размер мини выборки = 200				
Количество эпох	Алгоритмы оптимизации			
	SGD	Adam	AdaGrad	AdaDelta
5	67.20%	92.12%	91.96%	89.87%
10	78.14%	92.93%	92.44%	90.51%
50	87.46%	93.57%	93.09%	93.41%

Визуализируем изменение точности распознавания от эпохи к эпохе для проверочной выборки. И графика видно, что после 15 эпохи значение точности начинает колебаться в пределах ± 0.2 , это свидетельствует о переобучение сити.

Следовательно, количество эпох равное 20, является избыточным и следует ограничить их количество в пределах 10-15 эпох.

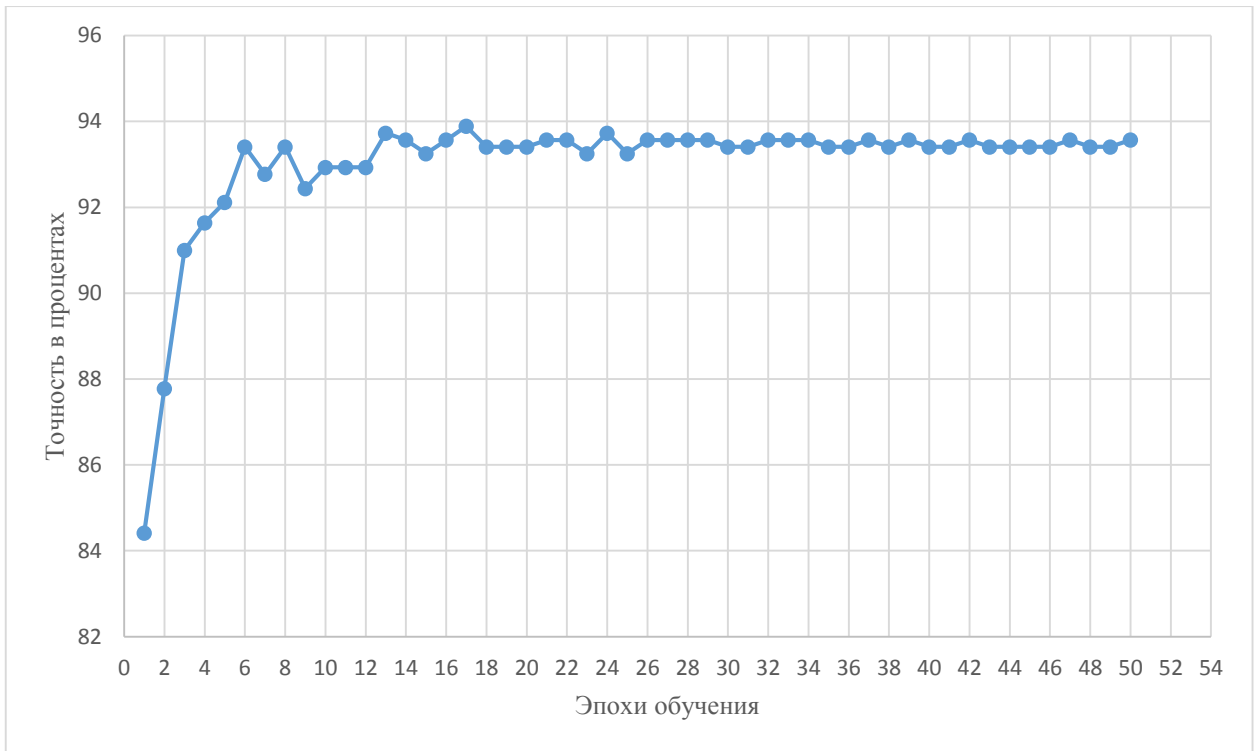


Рисунок 17 – Изменение точности распознавания для тестовой выборки

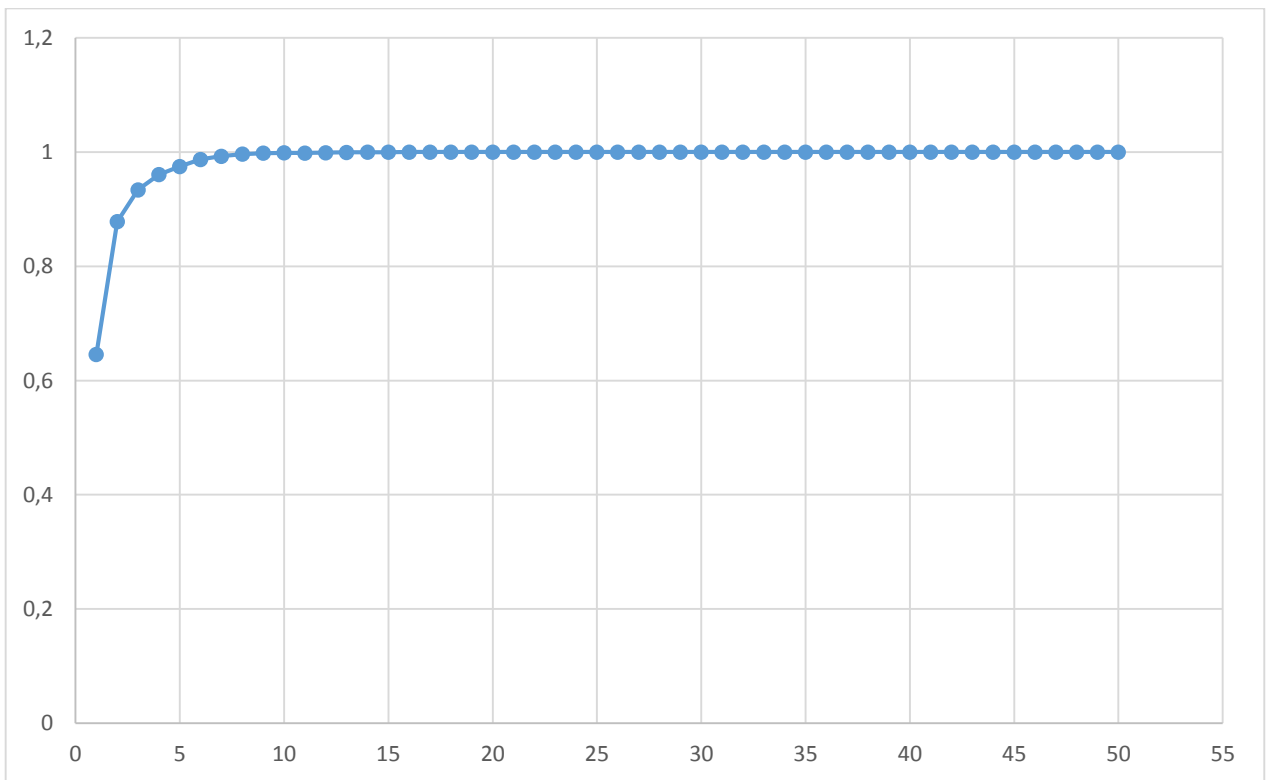


Рисунок 18 – Изменение точности распознавания для обучающей выборки

В таблице 9 приведены результаты обучения сети, для разного количества эпох. Исходя из полученных данных, можно сделать вывод, что в данном случае изменение количества эпох не значительно влияет на точность распознавания.

Таблица 9: Влияние размера мини выборки на точность распознавания

Количество эпох обучения = 15				
Размер мини выборки	Алгоритмы оптимизации			
	SGD	Adam	AdaGrad	AdaDelta
50	88.10%	94.05%	93.89%	93.41%
100	84.73%	93.73%	93.41%	92.44%
200	81.03%	93.25%	92.60%	90.84%

Лучшая точность распознавания, которой удалось добиться используя простую многослойную нейронную сеть с одним скрытым слоем для классификации рукописных букв русского алфавита составила 94.05%. Данный результат несколько ниже точности полученной для данных из обучающей выборки MNIST, с похожей архитектурой сети.

Можно сделать вывод что такая разница в точности связана с тем, что в наборе данных MNIST значительно больше обучающих примеров и меньше количество классов по сравнению с сформированной выборкой рукописных букв русского алфавита, где количество классов равно 32 и на каждый класс в среднем приходится 100 – 200 обучающих примеров.

2.3.3. Реализация сверточной нейронной сети на Keras

В данном разделе спроектируем сверточную нейронную сеть для распознавания рукописных букв. В таблице 8 представлена конфигурация сети с одним сверточным и подвыборочным слоем, размер матрицы свертки 5×5, размер окна подвыборки 2×2 (maxpooling).

Таблица 10: Конфигурация сверточной нейронной сети

Слой	1	2	3	4
Тип слоя	Слой свертки, Кол-во карт признаков: 32	Слой подвыборки Кол-во карт признаков: 32	Полностью-связанный слой, Кол-во нейронов: 128	Полностью-связанный слой, Кол-во нейронов: 32
Функция активации	ReLU	–	ReLU	Softmax
Количество настраиваемых параметров (весов)	832	–	802944	4128
Общее кол-во настраиваемых параметров	807 904			

На рисунке 19 визуализирована схема описанной в таблице 8 архитектуры сверточной нейронной сети для распознавания рукописных букв русского алфавита.

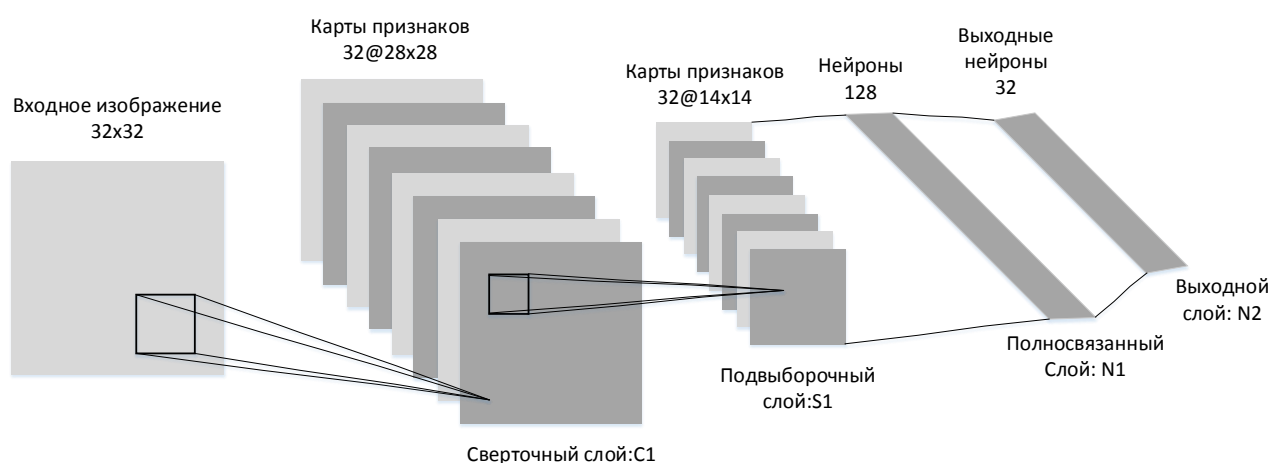


Рисунок 19 – Архитектура тестируемой сверточной нейронной сети

Ниже приведены результаты тестирования сверточной сети для набора данных рукописных букв. Точность распознавания на тестовой выборке представлена в таблицах 11 и 12.

Таблица 11: Влияние количества эпох на точность распознавания

Размер мини выборки = 100				
Количество эпох	Алгоритмы оптимизации			
	SGD	Adam	AdaGrad	AdaDelta
5	81.03%	94.53%	96.14%	93.09%
10	84.41%	95.18%	96.30%	94.21%
20	88.42%	95.98%	96.46%	95.82%

На рисунке 12, показан график изменение точности распознавания сети на проверочной выборки. В качестве анализируемых данных были выбраны показатели для лучшей точности из таблицы 9, которая оставила 96.46%.

Проанализировав данный график можно сделать вывод что количество эпох равное 20 является избыточным, так как значительное изменение точности происходит с 1 по 10 эпоху, а после 10 эпохи значение точности начинают колебаться, что свидетельствует о переобучении сети.

Следовательно, для дальнейших экспериментов следует ограничить количество эпох обучение, не более 10.

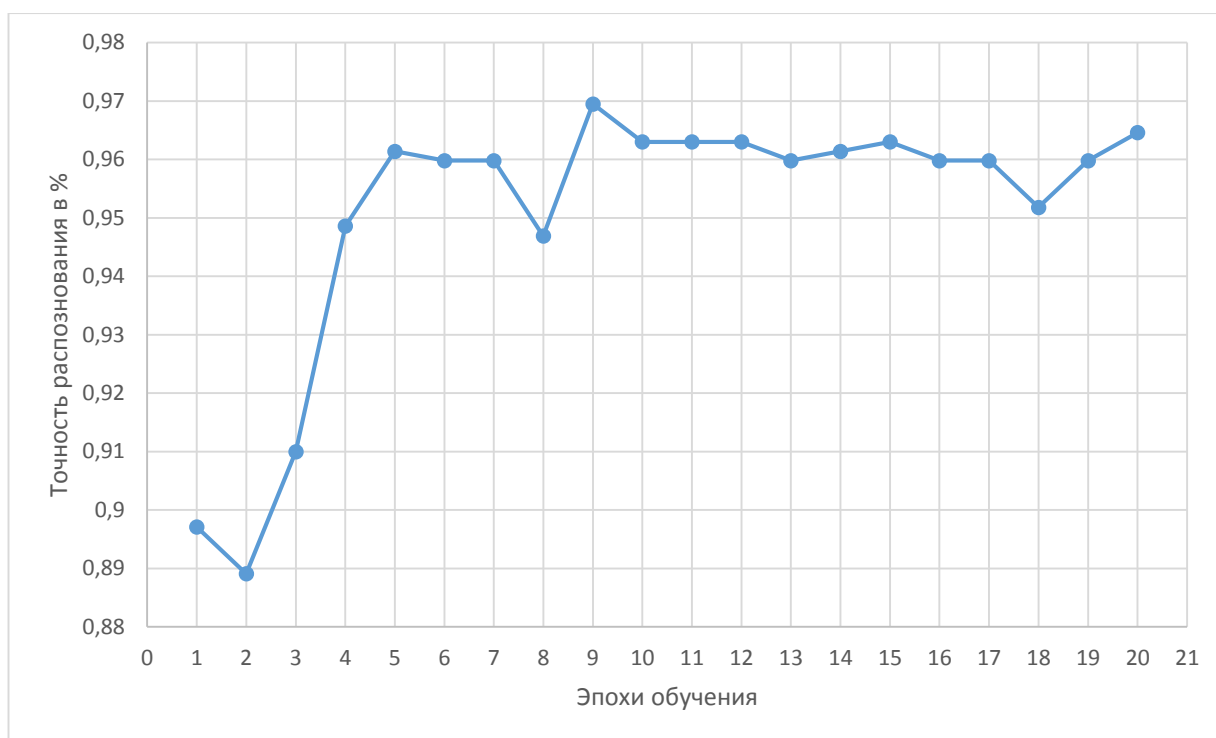


Рисунок 20 – Изменения точности распознавания для тестовой выборки

Таблица 12: Влияние размера минивыборки на точность распознавания

Количество эпох = 10				
Размер мини выборки	Алгоритмы оптимизации			
	SGD	Adam	AdaGrad	AdaDelta
50	89.23%	95.98%	96.46%	95.18%
100	84.41%	95.18%	96.30%	94.21%
200	83.92%	94.53%	96.46%	94.21%

В таблице 11 приведены значения точности для разного количества эпох, лучшие результаты были получены для размера минивыборки 50 и 200, учитывая, что данные значения равны, в дальнейшем, при обучении предпочтительней использовать размер минивыборки = 200, так как он требует меньших затрат на обучение.

Исходя из данных таблиц 11 и 12, можно сделать вывод что уменьшение размера мини выборки и увеличения количества эпох незначительно влияет на точность распознавания. Следовательно, дальнейшая точность распознавания может быть увеличена либо за счёт увеличения карт признаков в сверточном слое, либо за счет увеличения глубины сети.

Таблица 13: Влияние количества карт признаков на точность распознавания

Количество эпох = 10, размер минивыборки = 100				
Количество карт признаков	Алгоритмы оптимизации			
	SGD	Adam	AdaGrad	AdaDelta
32	84.41%	95.18%	96.30%	94.21%
64	85.21%	95.34%	96.78%	95.98%

Увеличение карт признаков в сверточном слое не привело к значительному изменению точности распознавания, следовательно, можно попробовать увеличить точность за счет добавления сверточных и подвыборочных слоев. Архитектура сверточной сети с двумя слоями свертки описана в таблице 14.

Таблица 14: Конфигурация сверточной нейронной сети

Слой	1	2	3	4	5	6
Тип слоя	Слой свертки, Кол-во карт признаков: 16, 5×5	Слой подвыборки Кол-во карт признаков: 16	Слой свертки, Кол-во карт признаков: 32	Слой подвыборки Кол-во карт признаков: 32, 5×5	Полно-связанный слой, Кол-во нейронов: 128	Полно-связанный слой, Кол-во нейронов: 32

Функция активации	ReLU	–	ReLU	–	ReLU	Softmax
Количество настраиваемых параметров (весов)	416	–	12832	–	102528	4128
Общее количество настр. параметров	119 904					

В таблице 15 приведены результаты тестирования описанной выше архитектуры сверточной сети.

Таблица 15: Влияние размера минивыборки на точность распознавания

Количество эпох = 10				
Размер минивыборки	Алгоритмы оптимизации			
	SGD	Adam	AdaGrad	AdaDelta
100	84.89%	96.46	96.80%	95.96%
200	56.27	95.03	95.78%	94.80%

Максимальная точность работы на тестовом наборе данных составила 96.80%, что практически идентично результатам, полученным с помощью сети с одним сверточным слоем, исходя из этого можно сделать вывод, что в условиях небольшого количества обучающих данных достигнутая точность в 96.8% является максимально доступной для данного объема обучающего набора.

Так же было принято решения для дальнейшей работы использовать архитектуру с одним сверточным слоем, так как разница в точности по сравнению с сетью с двумя сверточными слоями незначительна, но реализации более простая архитектура дает преимущество в скорости обучения сети и дальнейшей ее эксплуатации.

Следовательно, оптимальной была выбрана архитектура описанной выше сверточной сети с одним скрытым слое и параметрами обучения:

- Алгоритм оптимизации: AdaGrad
- Количество эпох обучения: 10
- Размер минивыборки: 200

2.4. Обзор основных блоков разрабатываемой системы

Блок сегментации изображения. Данный блок отвечает за определение области распознавания машиночитаемого бланка, выделение строк и символов на изображение. Так же он включает в себя этап бинаризации исходного изображения, так как используемые в работе алгоритмы сегментации работают с бинарными изображениями.

Масштабирование и центрирование символа. Данный блок отвечает за масштабирование изображения символа полученного после сегментации к размеру, на который ориентирована используемая нейронная сеть.

Инвертирование изображения. Данный блок выполняет инвертирование цветов входного изображения символа, в диапазоне от 0 до 255, где 0 – черный пиксель, 255 – белый.

Нормализация значений. Исходные значения пикселей входного изображения символа в диапазоне от 0 до 255, нормализуются в диапазоне от 0 до 1 с целью повышения эффективности работы сети.

Распознавание. На данном этапе происходит распознавание изображения с помощью, обученной сверточной нейронной сети.

На рисунке 21 представлена схематическая последовательность основных блоков разрабатываемой системы, в виде их логической последовательности применения к входному изображению.

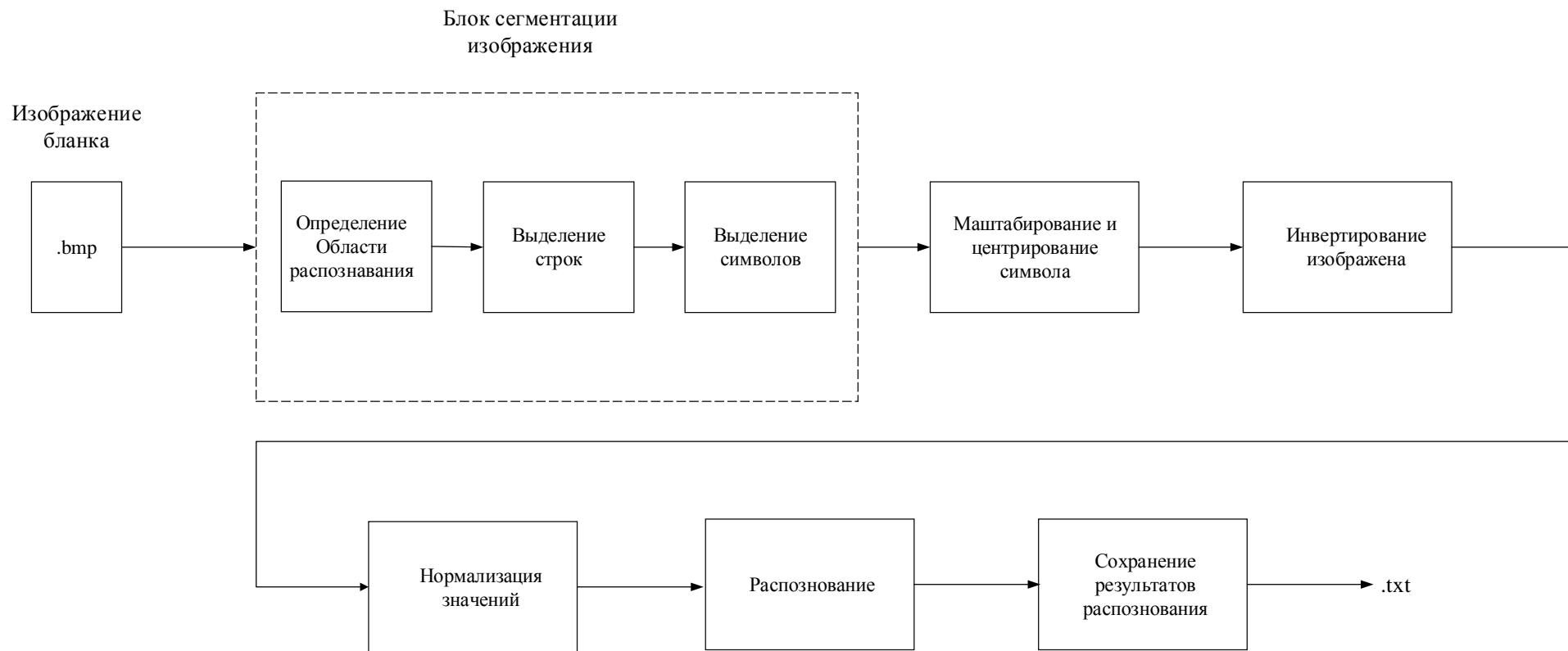


Рисунок 21 – Схема основных блоков системы

2.5. Проектирование логической модели системы

Построение логической модели начинается с определения классов и методов разрабатываемой системы. На рисунке представлены основные классы системы в виде UML диаграммы, для двух основных блоков: сегментации и распознавания.

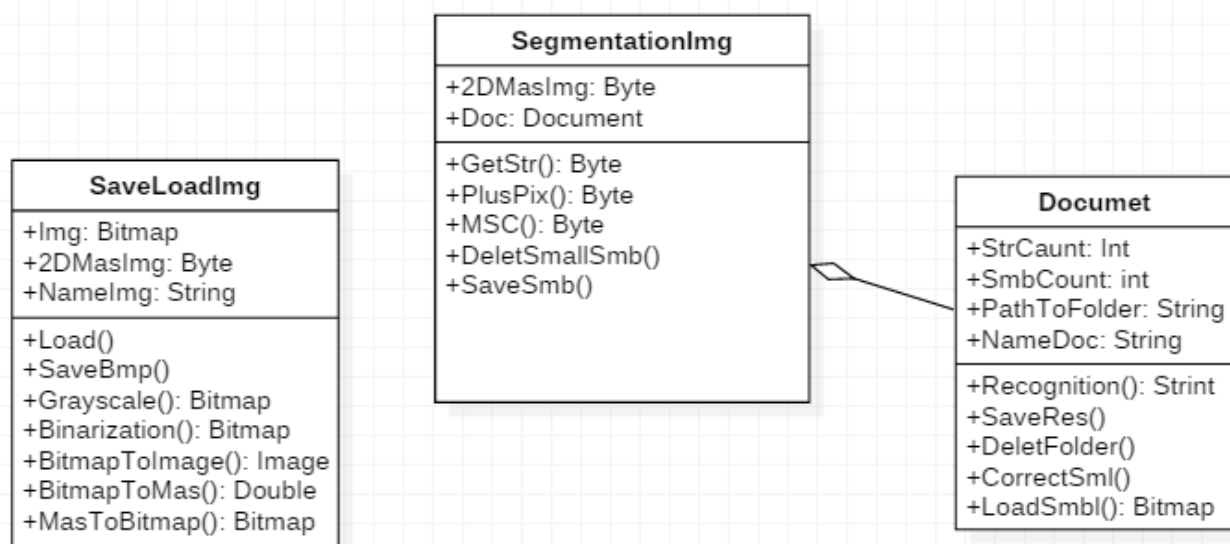


Рисунок 22 – UML диаграммы классов для блока сегментации

Класс `SaveLoadImg` – реализует основной функционал для загрузки исходного изображения и предобработки сходного изображения, для его последующий сегментации.

Класс `SegmentationImg` – отвечает за выделение строк и символов на исходном изображении, и сохраняет выделенные символы на диске.

Класс `Document` – содержит информацию о количестве строк и символов в распознаваемом документе. Основная задача этого класса – распознавание сегментированных символов с помощью обученной нейронной сети, которая подключается к проекту как сторонняя библиотека.

Так как разработанная нейронная сеть подключается к основному проекту в качестве библиотеки, ниже приведено описание ее основных классов. Исходя из архитектуры сети, были определены следующие классы:

Абстрактный класс `BaseObject` – описывает механизмы работы всех типов объектов проектируемой нейронной сети (нейроны, сверточные слои,

подвыборочные слои), объявляет методы, ответственные за режимы обучения и коррекции весов;

Класс `ConvolutionalPlane` – реализует функционал сверточного слоя сети;

Класс `SubsamplingPlane` – реализует функционал подвыборочного слоя сети;

Класс `Neuron` – реализует функционал для работы нейронов полносвязанного слоя;

Класс `Layer` – описывает механизмы работы слоев нейронной сети.

Абстрактный класс `InputData` – описывает механизмы обработки входных данных нейронной сети (в нашем случае изображений рукописных символов).

Класс `MNIST` – описывает механизмы работы с базой данных рукописных символов `MNIST` или с любой другой базой изображений, имеющих аналогичную структуру хранения.

Класс `NeuralNet` – содержит в себе основные настраиваемые параметры сети, сведения о ее общей структуре и значения результатов распознавания.

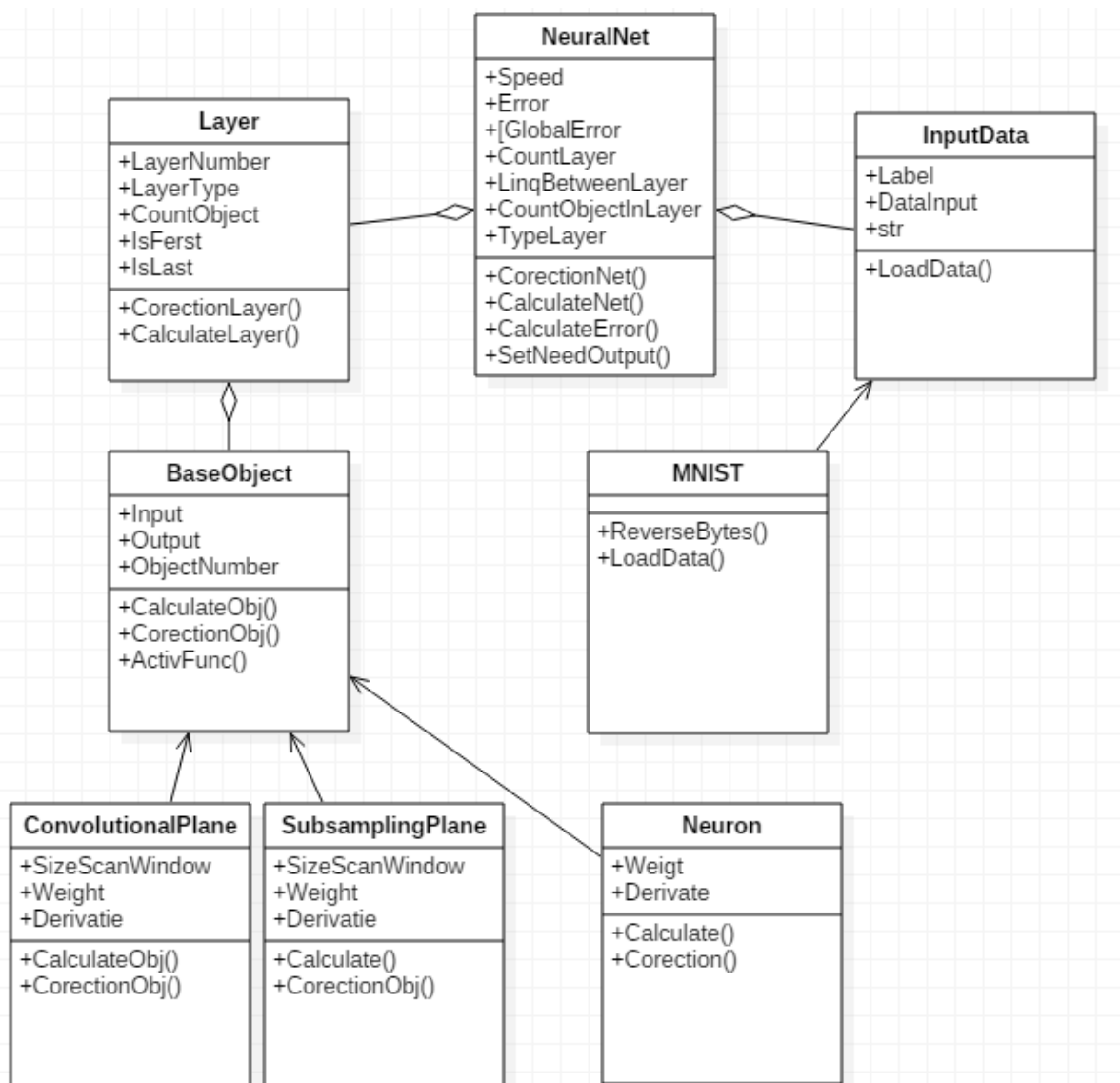


Рисунок 23 – Модель сверточной нейронной сети в виде UML диаграммы классов.

На данном этапе был определен основной функционал программы, были реализованы классы и методы, необходимые для работы выбранных алгоритмов сегментации и распознавания.

3. РЕЗУЛЬТАТЫ ПРОВЕДЕННОГО ИССЛЕДОВАНИЯ

Для оценивания результатов классификации была посчитана доля верно классифицированных объектов к общему количеству объектов. На данный момент лучшая точность распознавания, которой удалось добиться при реализации выбранной модели составляет 94.1%. Значение точность вычислялось по формуле:

$$R = \frac{n}{N} = \frac{623}{662} = 0.941 ,$$

где R – точность распознавания по всему набору тестовой выборки, n – количество правильно распознанных символов из тестовой выборки, N – количество элементов в тестовой выборки.

Достигнутая точность несколько отличается от той, которой удалось достигнуть при тестировании модели с помощью библиотеки машинного обучения. Данное различие связано с различной настройкой некоторых гиперпараметров в ходе реализации модели.

На рисунке 19 представлены некоторые примеры не верного распознавания символов различных категорий.

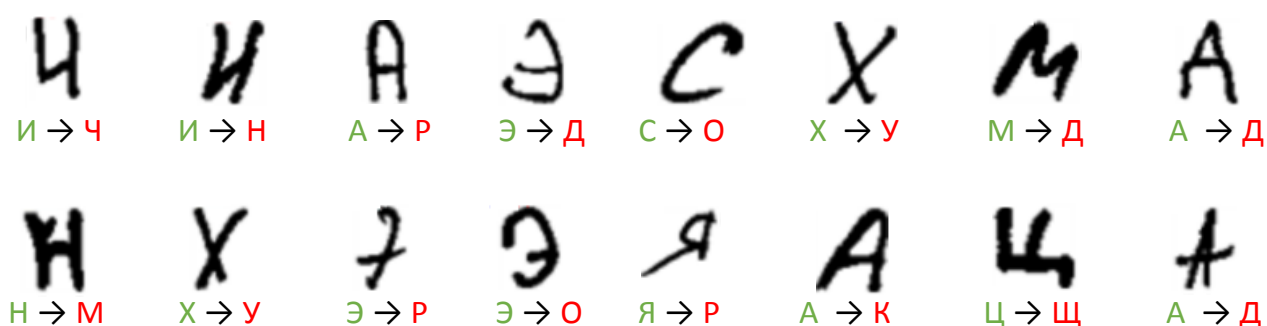


Рисунок 24 – Примеры не верно распознанных символов

На рисунке 15 и 16 визуализированы выходы сверточного и подвыборочного слоя соответственно, для буквы «Я».

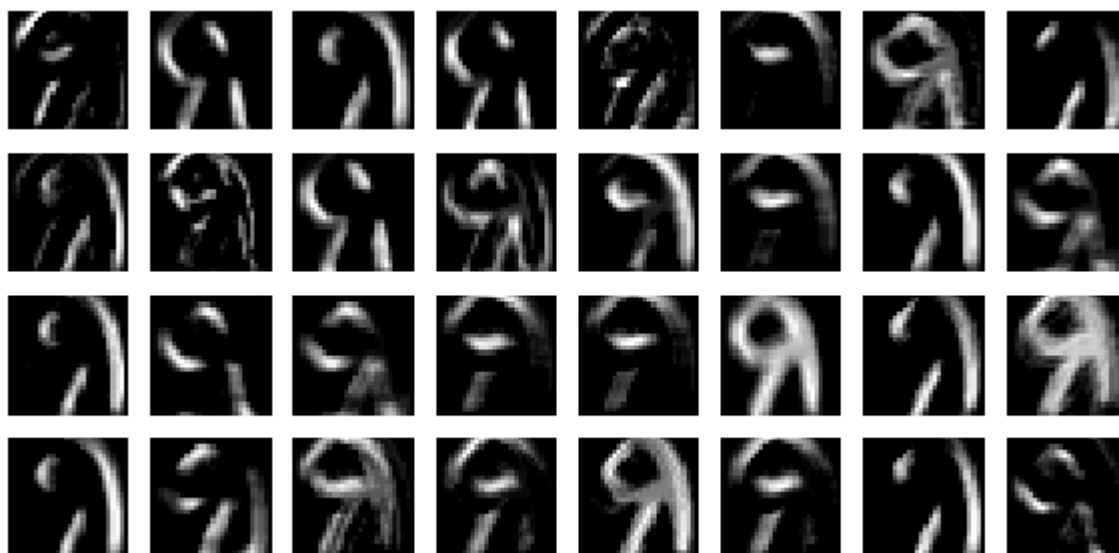


Рисунок 25 – Визуализация выходов сверточного слоя сети



Рисунок 26 – Визуализация выходов подвыборочного слоя сети

Полученные изображения, отражают принцип работы сверточной нейронной сети. Первый сверточный слой выделяет на изображении некоторые признаки с помощью матрицы свертки (чем больше карт в сверточном слое, тем больше признаков может выделить сеть), а следующий за сверточным – подвыборочный слой уменьшает исходное изображение в два раза делая сеть устойчивой к масштабированию и инвариантности.

4. ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ

Технико-экономическое обоснование научно-исследовательских работ проводится с целью определения и анализа трудовых и денежных затрат, направленных на их реализацию, а также уровня их научно-технической результативности.

4.1. Организация и планирование работ

При организации процесса реализации конкретного проекта необходимо оптимально планировать занятость каждого из его участников и сроки проведения отдельных работ.

На данном этапе составляется полный перечень проводимых работ, и определяются их исполнители и оптимальная продолжительность. Результатом планирования работ является сетевой, либо реализации проекта. Наиболее удобным, простым и наглядным способом для этих целей является использование линейного графика. Для его построения составим перечень работ и соответствие работ своим исполнителям, продолжительность выполнения этих работ и сведем их в таблицу 17.

Таблица 17 – Перечень работ и продолжительность их выполнения

Этапы работы	Исполнители	Загрузка исполнителей
Постановка целей и задач, получение исходных данных	НР, И	НР – 60% И – 40%
Подбор и изучение материалов по тематике	НР, И	НР – 30% И – 100%
Разработка календарного плана	НР, И	НР – 40% И – 60%
Обсуждение литературы	НР, И	НР – 20% И – 100%
Выбор алгоритмов для решения поставленной задачи	НР, И	НР – 20% И – 100%
Планирование архитектуры программного обеспечения	НР, И	НР – 10% И – 100%
Разработка программного обеспечения	И	И – 100%
Тестирование и отладка программного обеспечения	И	И – 100%
Оформление расчетно-пояснительной записки	И	И – 100%
Подведение итогов	НР, И	НР – 70% И – 100%

НР – научный руководитель.

И – исполнитель

4.1.1. Продолжительность этапов работ

Расчет продолжительности этапов работ осуществляется двумя методами:

- технико-экономическим;
- опытно-статистическим.

В данном случае используется опытно-статистический метод, который реализуется двумя способами:

- аналоговый;
- экспертный (вероятностный).

Для определения ожидаемого значения продолжительности работ $t_{ож}$ применяется вероятностный метод – метод двух оценок t_{min} и t_{max} .

$$t_{ож} = \frac{3 \cdot t_{min} + 2 \cdot t_{max}}{5}, \quad (1)$$

где t_{min} – минимальная трудоемкость работ, чел/дн., t_{max} – максимальная трудоемкость работ, чел/дн.

Для выполнения перечисленных в таблице 1 работ требуются специалисты:

- инженер – в его роли действует исполнитель НИР (ВКР);
- научный руководитель.

Для построения линейного графика необходимо рассчитать длительность этапов в рабочих днях, а затем перевести в календарные дни. Расчет продолжительности выполнения каждого этапа ведется по формуле:

$$T_{РД} = \frac{t_{ож}}{K_{ВН}} \cdot K_{Д},$$

где $t_{ож}$ – трудоемкость работы, чел/дн., $K_{ВН}$ – коэффициент выполнения работ ($K_{ВН} = 1$), $K_{Д}$ – коэффициент, учитывающий дополнительное время на компенсации и согласование работ ($K_{Д} = 1.2$).

Расчет продолжительности этапа в календарных днях ведется по формуле:

$$T_{КД} = T_{РД} \cdot T_{К},$$

где $T_{РД}$ – продолжительность выполнения этапа в рабочих днях, $T_{КД}$ – продолжительность выполнения этапа в календарных днях, $T_{К}$ – коэффициент календарности.

Коэффициент календарности рассчитывается по формуле:

$$T_{К} = \frac{T_{КАЛ}}{T_{КАЛ} - T_{ВД} - T_{ПД}},$$

где $T_{КАЛ}$ – календарные дни ($T_{КАЛ} = 365$), $T_{ВД}$ – выходные дни ($T_{ВД} = 52$), $T_{ПД}$ – праздничные дни ($T_{ПД} = 10$).

$$T_{К} = \frac{365}{365 - 52 - 10} = 1,205.$$

Таблица 18 – Трудозатраты на выполнение проекта

Этап	Исполнители	Продолжительность работ, дни			Длительность работ, чел/дн.			
		t_{min}	t_{max}	$t_{ож}$	$T_{рд}$		$T_{кд}$	
					НР	И	НР	И
Постановка целей и задач, получение данных	НР, И	1	3	1,8	1,296	0,864	1,568	1,04
Подбор и изучение материалов по тематике	НР, И	10	15	12	4,32	14,4	5,2	17,35
Разработка календарного плана	НР, И	2	4	2,8	1,34	2,016	1,61	2,429
Обсуждение литературы	НР, И	2	5	3,2	0,768	3,84	0,925	4,627
Выбор алгоритмов для решения поставленных задач	НР, И	10	17	12,8	3,072	15,36	3,7	18,5
Планирование архитектуры программного обеспечения	НР, И	15	22	17,8	2,136	21,36	2,57	25,738
Разработка программного обеспечения	И	30	50	38	0,00	45,6	0	54,948
Тестирование и отладка программного обеспечения	И	20	25	22	0,00	26,4	0	31,812
Оформление расчетно-пояснительной записки	И	5	7	5,8	0,00	6,96	0	8,3868
Подведение итогов	НР, И	2	4	2,8	2,35	3,36	2,83	4,0488
Итого:				119	15,282	140,16	18,41	168,89

4.1.2. Расчет нарастания технической готовности работ

Величина нарастания технической готовности работы показывает процент выполнения работы на каждом этапе. Расчетная формула:

$$H_i = \frac{t_{Hi}}{t_0} \cdot 100\%, \quad (5)$$

где t_{Hi} – нарастающая трудоемкость с момента начала работы i -го этапа, t_0 – общая трудоемкость.

Общая трудоемкость вычисляется по формуле:

$$t_0 = \sum_{i=1}^n t_{OЖi}, \quad (6)$$

где $t_{OЖi}$ – ожидаемая продолжительность i -го этапа.

Удельный вес каждого этапа Y_i определяется по формуле:

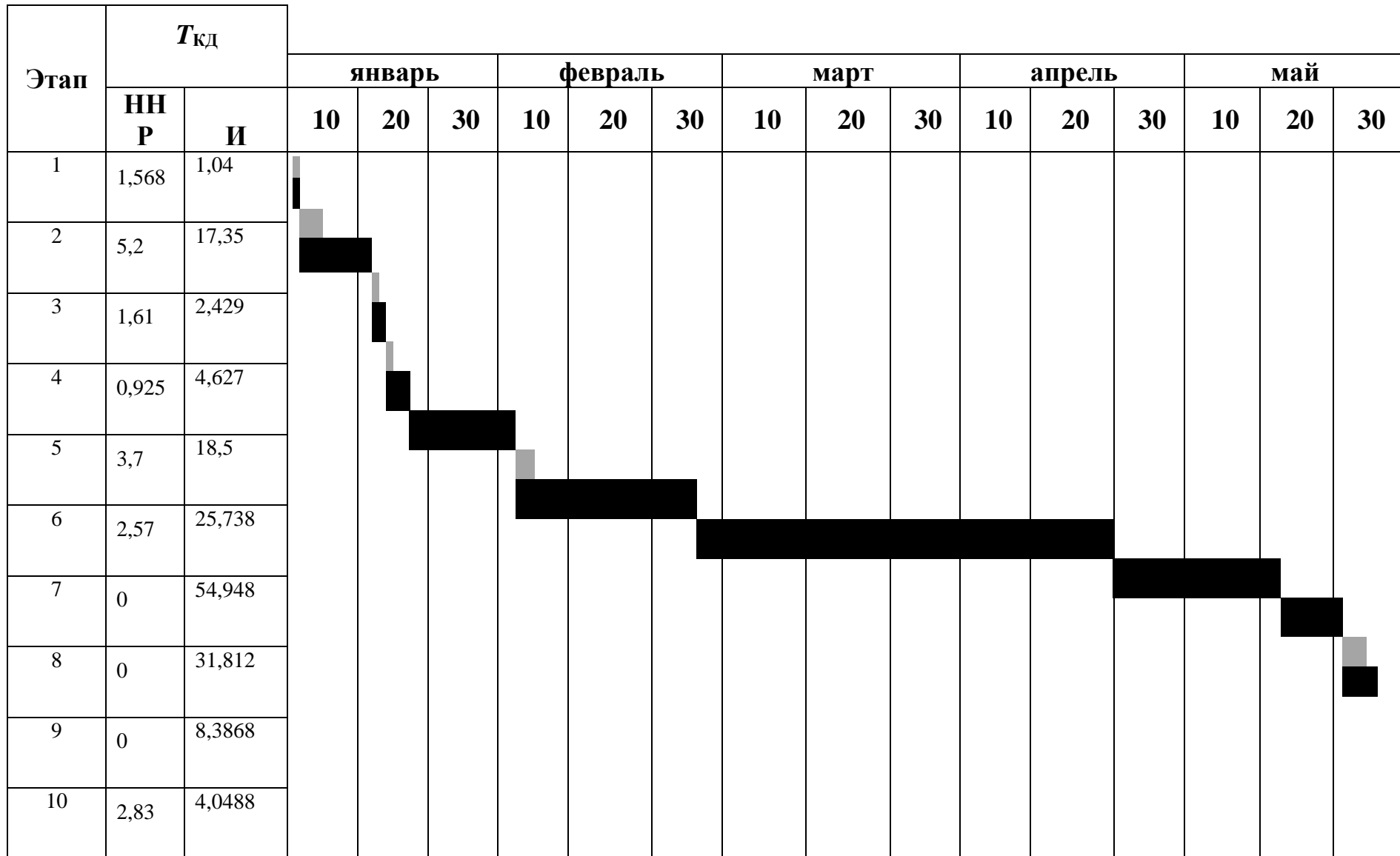
$$Y_i = \frac{t_{OЖi}}{t_0} \cdot 100\% \quad (7)$$

Результаты вычислений H_i и Y_i отражены в таблице 19.

Таблица 19 – Нарастание технической готовности и удельный вес этапов

Этап	$H_i, \%$	$Y_i, \%$
Постановка целей и задач, получение исходных данных	1,51	1,51
Подбор и изучение материалов по тематике	11,59	10,08
Разработка календарного плана	13,949	2,35
Обсуждение литературы	16,63	2,689
Выбор алгоритмов для решения поставленных задач	27,39	10,756
Планирование архитектуры программного обеспечения	42,35	14,958
Разработка программного обеспечения	74,285	31,93
Тестирование и отладка программного обеспечения	92,77	18,487
Оформление расчетно-пояснительной записки	97,647	4,87
Подведение итогов	100	2,35

Таблица 20 – Линейный график работ



НР  И 

4.2. Расчет сметы затрат на разработку ПО и проведение эксперимента

В состав затрат на разработку ПО и проведение эксперимента включается стоимость всех расходов, необходимых для реализации комплекса работ, составляющих содержание данной разработки. Расчет сметной стоимости на выполнение данной разработки производится по следующим статьям затрат:

- материалы и покупные изделия;
- основная заработная плата;
- отчисления в социальные фонды;
- расходы на электроэнергию;
- амортизационные отчисления;
- работы, выполняемые сторонними организациями;
- прочие расходы.

4.2.1. Расчет затрат на материалы

К данной статье расходов относится стоимость материалов, покупных изделий, полуфабрикатов и других материальных ценностей, расходуемых непосредственно в процессе выполнения работ. Так как основным рабочим инструментом при разработке данного проекта выступал персональный компьютер, затраты на материалы и оборудование составляют 0 рублей в связи с отсутствием необходимости в расходных материалах.

4.2.2. Расчет основной заработной платы

Данная статья расходов включает заработную плату научного руководителя и инженера, а также премии, входящие в фонд заработной платы. Расчет основной заработной платы выполняется на основе трудоемкости выполнения каждого этапа и величины месячного оклада исполнителя.

Среднедневная заработная плата рассчитывается по формуле:

$$\text{Дневная з/плата} = \frac{\text{Месячный оклад}}{25,17 \text{ дней}}$$

Расчеты затрат на основную заработную плату приведены в таблице 6. При расчете учитывалось, что в году 302 рабочих дня и, следовательно, в месяце 25,17 рабочих дня. Затраты времени на выполнение работы по каждому исполнителю брались из таблицы 21. Также был принят во внимание коэффициент, учитывающий коэффициент по премиям $K_{\text{доп.ЗП}} = 1,188$, $K_{\text{ПР}} = 1,1$ и районный коэффициент $K_{\text{РК}} = 1,3$ ($K = 1 * K_{\text{ПР}} * K_{\text{РК}} * K_{\text{РК}} = 1 * 1,1 * 1,3 * 1,188 = 1,699$).

Таблица 21– Затраты на основную заработную плату

Исполнитель	Оклад, руб./мес.	Среднедневная ставка, руб./день	Затраты времени, дни	Коэффициент	Фонд з/платы, руб.
НР	25000	993,25	15,282	1,699	25788,9
И	15000	595,95	140,16	1,699	141915
Итого:					167703,9

4.2.3. Расчет отчислений от заработной платы

Затраты по этой статье составляют отчисления в социальные фонды. Отчисления по заработной плате определяются по следующей формуле:

$$C_{\text{соц}} = K_{\text{соц}} * C_{\text{зп}},$$

где $K_{\text{соц}}$ – коэффициент, учитывающий размер отчислений из заработной платы. Данный коэффициент составляет 30,2% от затрат на заработную плату и включает в себя:

- отчисления в пенсионный фонд;
- на социальное страхование;
- на медицинское страхование.

Итак, отчисления из заработной платы составили:

$$C_{\text{соц}} = 0,302 * 167703,9 = 50646,6 \text{ руб.}$$

4.2.4. Расчет затрат на электроэнергию

Данный вид расходов включает в себя затраты на электроэнергию при работе оборудования. Затраты на электроэнергию при работе оборудования для технологических целей рассчитываются по формуле:

$$\text{Э}_{\text{ОБ}} = P_{\text{ОБ}} \cdot \text{Ц}_{\text{Э}} \cdot t_{\text{ОБ}} ,$$

где $\text{Э}_{\text{ОБ}}$ – затраты на электроэнергию, потребляемую оборудованием, руб., $P_{\text{ОБ}}$ – мощность, потребляемая оборудованием, кВт, $\text{Ц}_{\text{Э}}$ – тарифная цена за 1 кВт·час, $\text{Ц}_{\text{Э}} = 1,8$ руб/кВт·час, $t_{\text{ОБ}}$ – время работы оборудования, час.

Время работы оборудования вычисляется на основе данных для $T_{\text{РД}}$ таблицы 18 для инженера из расчета, что продолжительность рабочего дня равна 8 часов.

Мощность, потребляемая оборудованием, определяется по формуле:

$$P_{\text{ОБ}} = P_{\text{УСТ.ОБ}} \cdot K_{\text{С}} ,$$

где $P_{\text{УСТ.ОБ}}$ – установленная мощность оборудования, кВт, $K_{\text{С}}$ – коэффициент загрузки, зависящий от средней степени использования номинальной мощности. Для технологического оборудования малой мощности $K_{\text{С}} = 1$.

Затраты на электроэнергию для технологических целей приведены в таблице 6.

Таблица 22 – Затраты на электроэнергию для технологических целей

Наименование оборудования	Время работы оборудования $t_{\text{ОБ}}$, час	Потребляемая мощность $P_{\text{ОБ}}$, кВт	Затраты $\text{Э}_{\text{ОБ}}$, руб.
Персональный компьютер	977,6	0,3	527,9
Терминал Мобис-Т	977,6	0,005	8,8
Итого:			536,7

4.2.5. Расчет амортизационных расходов

В статье «Амортизационные отчисления» от используемого оборудования рассчитывается амортизация за время выполнения работы для оборудования, которое имеется в наличии.

Амортизационные отчисления рассчитываются на время использования ЭВМ по формуле:

$$C_{AM} = \frac{N_A \cdot Ц_{OB}}{F_D} \cdot t_{BT} \cdot n,$$

где N_A – годовая норма амортизации, $N_A = 25\%$, $Ц_{OB}$ – цена оборудования, $Ц_{OB} = 50000$ руб., F_D – действительный годовой фонд рабочего времени, $F_D = 2416$ часов, t_{BT} – время работы вычислительной техники при создании программного продукта, $t_{BT} = 984$ часа, n – число задействованных ПЭВМ, $n = 1$.

Итак, затраты на амортизационные отчисления составили:

$$C_{ам} = 5091,06 \text{ руб}$$

4.2.6. Расчет прочих расходов

В статье «Прочие расходы» отражены расходы на разработку проекта, которые не учтены в предыдущих статьях.

Прочие расходы составляют 10 % от единовременных затрат на выполнение технического продукта и проводятся по формуле:

$$C_{проч} = (C_{зп} + C_{соц} + C_{эл.об.} + C_{ам}) * 0,1$$

$$C_{проч} = (167703,9 + 50646,6 + 536,7 + 5091,06) * 0,1 = 22397,83$$

4.2.7. Расчет общей себестоимости разработки

Проведя расчет сметы затрат на разработку, можно определить общую стоимость разработки проекта «Разработка алгоритма сегментации и распознавания машиночитаемых бланков на основе сверточных нейронных сетей».

Таблица 23 – Смета затрат на разработку проекта

Статья затрат	Условное обозначение	Сумма, руб.
1 Материалы и покупные изделия	$C_{\text{МАТ}}$	0
2 Основная заработная плата	$C_{\text{ЗП}}$	167703,9
3 Отчисления в социальные фонды	$C_{\text{СОЦ}}$	50646,6
4 Расходы на электроэнергию	Э	536,7
5 Амортизационные отчисления	$C_{\text{АМ}}$	5091,06
6 Работы, выполняемые сторонними организациями	$C_{\text{СТОП}}$	0
7 Прочие расходы	$C_{\text{ПРОЧ}}$	22397,83
Итого:		223978,3

Таким образом, расходы на разработку составили $C = 223978,3$ руб.

4.2.8. Прибыль

Прибыль составляет 44795,65 рублей (20 %) от расходов на разработку.

4.2.9. НДС

НДС составляет 18% от суммы затрат на разработку. Сумма НДС составила 40316,09 руб.

4.2.10. Цена разработки НИР

Цена равна сумме полной себестоимости, прибыли и НДС:

$$C_{\text{НИР(КР)}} = 223978,3 + 44795,65 + 40316,09 = 309091,04 \text{ руб.}$$

4.3. Оценка экономической эффективности проекта

Выполнение научно-исследовательских работ оценивается уровнями достижения экономического, научного, научно-технического и социального эффектов.

Научный эффект характеризует получение новых научных знаний и отображает прирост информации, предназначенной для внутринаучного потребления. Научно-технический эффект характеризует возможность использования результатов в других проектах и обеспечивает получение информации, необходимой для создания новой техники. Экономический эффект характеризуется в частности выраженной в стоимостных показателях экономией живого общественного труда, а также в возможности применения полученных знаний для создания новых разработок. Социальный эффект проявляется в улучшении условий труда.

Экономическая эффективность проекта обусловлена возрастающей необходимостью в системах автоматизации процессов обработки машиночитаемых бланков. Конечная система в виде автоподключаемого модуля или библиотеки с возможностью сегментирования и распознавания рукописной части машиночитаемых бланков является востребованной и актуальной на сегодняшний день. Использованной подобной системы в коммерческой и социальной сфере позволяет решать такие задачи, как оперативная обработка рукописной части бланков, а так же систематизация и проверка полученных результатов.

Подводя итог вышесказанному, экономический эффект от реализации проекта может быть выражен в снижении затрат на покупку подобных систем от сторонних разработчиков, а так же повышение эффективности обработки бланков за счет экономии времени при замене ручной обработки на автоматизированную.

4.3.1. Оценка научно-технического уровня НИР

Научно-технический уровень характеризует, в какой мере выполнены работы и обеспечивается научно-технический прогресс в данной области. Для оценки научной ценности, технической значимости и эффективности, планируемых и выполняемых НИР, используется метод бальных оценок. Бальная оценка заключается в том, что каждому фактору по принятой шкале

присваивается определенное количество баллов. Обобщенную оценку проводят по сумме баллов по всем показателям или рассчитывают по формуле. На этой основе делается вывод о целесообразности НИР.

Сущность метода заключается в том, что на основе оценок признаков работы определяется коэффициент ее научно-технического уровня по формуле:

$$K_{\text{НТУ}} = \sum_{i=1}^3 R_i \cdot n_i ,$$

где $K_{\text{НТУ}}$ – коэффициент научно-технического уровня, R_i – весовой коэффициент i -го признака научно-технического эффекта, n_i – количественная оценка i -го признака научно-технического эффекта, в баллах.

Таблица 24 – Весовые коэффициенты признаков НТУ

Признак НТУ	Примерное значение весового коэф-та n_i	
1 Уровень новизны	Систематизируются и обобщаются сведения, определяются пути дальнейших исследований	0,4
2 Теоретический уровень	Разработка способа (алгоритм, программа мероприятий, устройство, вещество и т.п.)	0,1
3 Возможность реализации	Время реализации в течение первых лет	0,5

Таблица 25– Баллы для оценки уровня новизны

Уровень новизны	Характеристика уровня новизны	Баллы
Принципиально новая	Новое направление в науке и технике, новые факты и закономерности, новая теория, вещество, способ	8 – 10
Новая	По-новому объясняются те же факты, закономерности, новые понятия дополняют ранее полученные результаты	5 – 7
Относительно новая	Систематизируются, обобщаются имеющиеся сведения, новые связи между известными факторами	2 – 4
Не обладает новизной	Результат, который ранее был известен	0

Таблица 26 – Баллы значимости теоретических уровней

Теоретический уровень полученных результатов	Баллы
1 Установка закона, разработка новой теории	10
2 Глубокая разработка проблемы, многоспектральный анализ, взаимодействия между факторами с наличием объяснений	8
3 Разработка способа (алгоритм, программа и т. д.)	6
4 Элементарный анализ связей между фактами (наличие гипотезы, объяснения версии, практических рекомендаций)	2
5 Описание отдельных элементарных факторов, изложение наблюдений, опыта, результатов измерений	0,5

Таблица 27 – Возможность реализации научных, теоретических результатов по времени и масштабам

Время реализации	Баллы
В течение первых лет	10
От 5 до 10 лет	4
Свыше 10 лет	2

Результаты оценок признаков научно-технического уровня приведены в таблице 28.

Таблица 28 – Количественная оценка признаков НИОКР

Признак научно-технического эффекта НИР	Характеристика признака НИОКР	R_i
Уровень новизны	Систематизируются, обобщаются имеющиеся сведения, новые связи между известными факторами	00,4
Теоретический уровень	Разработка способа (алгоритм, программа мероприятий, устройство, вещество и т.п.)	00,1
Возможность реализации	Время реализации 5-10 лет	00,5

Обоснование оценки признаков НИОКР приводится в таблице 29.

Таблица 29 – Оценки научно-технического уровня НИР

Фактор НТУ	Значимость	Уровень фактора	Выбранный балл	Обоснование выбранного балла
Уровень новизны	0,4	Относительно новая	4	Облегчит обработку машиночитаемых бланков
Теоретический уровень	0,1	Разработка способа	6	Разработана и реализован алгоритм, решающий поставленные в рамках задания задачи
Возможность реализации	0,5	В течение первых лет	7	Полученный продукт находится на стадии тестирования и внедрения

Исходя из оценки признаков НИОКР, показатель научно-технического уровня для данного проекта составил:

$$K_{\text{НТУ}} = 0,4*4 + 0,1*6 + 0,5*7 = 1,6 + 0,6 + 3,5 = 5,7$$

Таблица 30 – Оценка уровня научно-технического эффекта

Уровень НТЭ	Показатель НТЭ
Низкий	1-4
Средний	4-7
Высокий	8-10

Таким образом, исходя из данных в таблице 30, проект «Разработка алгоритма сегментации и распознавания машиночитаемых бланков с помощью сверточных нейронных сетей» имеет средний уровень научно-технического эффекта.

5. СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

5.1. Введение

Разработанный в рамках магистерской диссертации проект является программным обеспечением, предназначенным для сегментации и распознавания машиночитаемых бланков. Разработка программы велась исключительно при помощи компьютера. Сферы применения разработки очень широки: от нужд обычных пользователей до специализированных задач. Одним из наиболее перспективных применений является разработка на основе описываемого в данной работе метода программной системы обработки бланков ЕГЭ. Независимо от конкретного применения, взаимодействие пользователя с разработанной программой в любом случае производится с помощью программных и аппаратных средств ПЭВМ, а также с помощью периферийных устройств, подключенных к ПЭВМ.

Данный раздел посвящен анализу вредных и опасных факторов производственной среды для операторов ПЭВМ а так же разработке программ по минимизации воздействия вредоносного и опасного влияния выявленных факторов, а также программ по снижению вредных воздействий на окружающую среду, экономии невозполнимых ресурсов и защите в чрезвычайных ситуациях.

5.2. Производственная безопасность

Для обеспечения производственной безопасности необходимо проанализировать воздействия на человека вредных и опасных производственных факторов, которые могут возникать при разработке или эксплуатации проекта.

Производственный фактор считается вредным, если воздействие этого фактора на работника может привести к его заболеванию. Производственный фактор считается опасным, если его воздействие на работника может привести к его травме [16].

Все производственные факторы классифицируются по группам элементов: физические, химические, биологические и психофизические. Для данной работы целесообразно рассмотреть физические и психофизические вредные и опасные факторы производства, характерные как для рабочей зоны программиста, как разработчика рассматриваемой в данной работе системы, так и для рабочей зоны пользователя готового продукта. Выявленные факторы представлены в таблице 3 **Ошибка! Источник ссылки не найден.**

Таблица 31 – Вредные и опасные производственные факторы при выполнении работ за ПЭВМ [18]

Источник фактора, наименование видов работ	Факторы (по ГОСТ 12.0.003-74)		Нормативные документы
	Вредные	Опасные	
1) Работа за ПЭВМ	1) Повышенная или пониженная температура воздуха рабочей зоны; 2) Недостаточная освещенность рабочей зоны.	1) Опасность поражения электрическим током; 2) Опасность возникновения пожара.	1) СанПиН 2.2.4.548-96; 2) СанПиН 2.2.2/2.4.1340-03; 3) СП 52.13330.2011; 4) ГОСТ Р 12.1.019-2009 ССБТ; 5) СНиП 21-01-97.

5.2.1 Вредные производственные факторы

5.2.1.1 Повышенная или пониженная температура воздуха рабочей среды

Данный фактор является вредным производственным фактором и является фактором микроклимата рабочей среды, параметры которого регулируются СанПиН 2.2.4.548-96. Он больше характерен для рабочей среды

программиста-разработчика системы. К параметрам, характеризующим микроклимат в производственных помещениях, относятся:

- Температура воздуха (t , °С);
- Температура поверхностей (t , °С);
- Относительная влажность воздуха (φ , %);
- Скорость движения воздуха (v , м/с);
- Интенсивность теплового облучения (I , Вт/м²).

В производственных помещениях для работы с ПЭВМ происходит постоянное выделение тепла самой вычислительной техникой, вспомогательными приборами и средствами освещения. Поскольку оператор расположен в непосредственной близости с источниками выделения тепла, то данный фактор является одним из важнейших вредных факторов производственной среды оператора ПЭВМ, а высокая температура воздуха способствует быстрому перегреву организма и быстрой утомляемости [18].

Влажность оказывает большое влияние на терморегуляцию организма. Так, например, высокие показатели относительной влажности (более 85 %) затрудняют терморегуляцию снижая возможность испарения пота, низкие показатели влажности (менее 20 %) вызывают пересыхание слизистых оболочек человека [19].

Санитарные нормы устанавливают оптимальные и допустимые значения величин показателей микроклимата рабочих мест для различных категорий работ в теплый и холодный периоды года. Для программиста или оператора ПЭВМ категория работ является лёгкой (1а), т.к. работа проводится сидя, без систематических физических нагрузок. Оптимальные параметры микроклимата в офисных помещениях приведены в таблице 32.

Таблица 32 – Оптимальные параметры микроклимата производственных помещений оператора ПЭВМ

Период года	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность, %	Скорость движения воздуха, м/с
-------------	-------------------------	------------------------------	----------------------------	--------------------------------

Холодный	22–24	21–25	60–40	0,1
Теплый	23–25	22–26	60–40	0,1

Холодный период года – среднесуточная температура воздуха 10 °С и ниже, теплый период года – среднесуточная температура воздуха выше 10 °С.

В таблице 333 приведены допустимые показатели микроклимата для офисных помещений.

Таблица 33 – Допустимые показатели микроклимата производственных помещений оператора ПЭВМ [20]

Период года	Температура воздуха, °С		Температура поверхностей, °С	Относительная влажность, %	Скорость движения воздуха, м/с, для диапазона температур воздуха	
	ниже оптимальных величин	выше оптимальных величин			ниже оптимальных величин, не более	выше оптимальных величин, не более
Холодный	20,0–21,9	24,1–25,0	19–26	15–75	0,1	0,1
Теплый	21,0–22,9	25,1–28,0	20–29	15–75	0,1	0,2

5.2.1.2 Недостаточная освещенность рабочей зоны

Недостаточная освещенность рабочей зоны является вредным производственным фактором, возникающим при работе с ПЭВМ, уровни которого регламентируются СП 52.13330.2011.

Причиной недостаточной освещенности являются недостаточность естественного освещения, недостаточность искусственного освещения, пониженная контрастность.

Работа с компьютером подразумевает постоянный зрительный контакт с дисплеем ПЭВМ и занимает от 80 % рабочего времени. Недостаточность освещения снижает производительность труда, увеличивает утомляемость и количество допускаемых ошибок, а также может привести к появлению профессиональных болезней зрения.

Разряд зрительных работ программиста и оператора ПЭВМ относится к разряду III и подразряду г (работы высокой точности). В таблице 34 представлены нормативные показатели искусственного освещения при работах заданной точности.

Таблица 34– Требования к освещению помещений промышленных предприятий для операторов ПЭВМ

Характеристика зрительной работы	Наименьший или эквивалентный размер объекта различения, мм	Разряд зрительной работы	Подразряд зрительной работы	Контраст объекта с фоном	Характеристика фона	Искусственное освещение		
						Освещённость, лк		
						При системе комбинированного освещения		При системе общего освещения
						всего	в том числе от общего	
Высокой точности	От 0,3 до 0,5	III	г	Средний, большой	Светлый, средний	400	200	200

5.2.2 Опасные производственные факторы

5.2.2.1 Опасность поражения электрическим током

Поражение электрическим током является опасным производственным фактором и, поскольку оператор ПЭВМ имеет дело с электрооборудованием, то вопросам электробезопасности на его рабочем месте должно уделяться много внимания. Нормы электробезопасности на рабочем месте регламентируются СанПиН 2.2.2/2.4.1340-03, вопросы требований к защите от поражения электрическим током освещены в ГОСТ Р 12.1.019-2009 ССБТ.

Электробезопасность – система организационных и технических мероприятий и средств, обеспечивающих защиту людей от вредного и опасного воздействия электрического тока, электрической дуги, электромагнитного поля и статического электричества.

Опасность поражения электрическим током усугубляется тем, что человек не в состоянии без специальных приборов обнаружить напряжение дистанционно.

Помещение, где расположено рабочее место оператора ПЭВМ, относится к помещениям без повышенной опасности ввиду отсутствия следующих факторов: сырость, токопроводящая пыль, токопроводящие полы, высокая температура, возможность одновременного прикосновения человека к имеющим соединение с землей металлоконструкциям зданий, технологическим аппаратам, механизмам и металлическим корпусам электрооборудования.

Для оператора ПЭВМ при работе с электрическим оборудованием обязательны следующие меры предосторожности:

- Перед началом работы нужно убедиться, что выключатели и розетка закреплены и не имеют оголённых токоведущих частей;
- При обнаружении неисправности оборудования и приборов необходимо, не делая никаких самостоятельных исправлений, сообщить человеку, ответственному за оборудование [22, 24].

5.2.2.2 Опасность возникновения пожара

Возникновение пожара является опасным производственным фактором, т.к. пожар на предприятии наносит большой материальный ущерб, а также часто сопровождается травмами и несчастными случаями. Регулирование пожаробезопасности производится СНиП 21-01-97.

В помещениях с ПЭВМ повышен риск возникновения пожара из-за присутствия множества факторов: наличие большого количества электронных схем, устройств электропитания, устройств кондиционирования воздуха; возможные неисправности электрооборудования, освещения, или неправильная их эксплуатация может послужить причиной пожара.

Возможные виды источников воспламенения:

- Искра при разряде статического электричества;
- Искры от электрооборудования;
- Искры от удара и трения;
- Открытое пламя [25].

5.2.3 Мероприятия и рекомендации по устранению и минимизации

Для поддержания нормальных значений параметров микроклимата на рабочих местах рекомендуется оснащать их системами отопления, вентиляции и кондиционирования воздуха. Также, в некоторых случаях, целесообразно обеспечить питьевое водоснабжение. В помещениях для работы с ПЭВМ должна производиться ежедневная влажная уборка, а также систематическое проветривание после каждого часа работы [20].

Для защиты операторов ПЭВМ от негативного воздействия электромагнитных полей в первую очередь необходимо, чтобы используемая техника удовлетворяла нормам и правилам сертификации. При работе с ПЭВМ установлены регламентированные перерывы, а также иногда предусмотрено использование экранов и фильтров в целях защиты оператора [22].

Для создания и поддержания благоприятных условий освещения для операторов ПЭВМ, их рабочие места должны соответствовать санитарно-

эпидемиологическим правилам СанПиН 2.2.2/2.4.1340-03. Рабочее помещение должно иметь естественное и искусственное освещение, соответствующее показателям, представленным в таблице 34. Для рассеивания естественного освещения следует использовать жалюзи на окнах рабочих помещений. В качестве источников искусственного освещения должны быть использованы люминесцентные лампы, лампы накаливания – для местного освещения [23].

Для предупреждения преждевременной утомляемости пользователей ПЭВМ рекомендуется организовывать рабочую смену путем чередования работ с использованием ПЭВМ и без него. В случаях, когда характер работы требует постоянного взаимодействия с компьютером (работа программиста-разработчика) с напряжением внимания и сосредоточенности, при исключении возможности периодического переключения на другие виды трудовой деятельности, не связанные с ПЭВМ, рекомендуется организация перерывов на 10–15 мин. через каждые 45–60 мин. работы. При высоком уровне напряженности работы рекомендуется психологическая разгрузка в специально оборудованных помещениях [22].

К мероприятиям по предотвращению возможности поражения электрическим током относятся:

- При производстве монтажных работ необходимо использовать только исправный инструмент, аттестованный службой КИПиА;
- С целью защиты от поражения электрическим током, возникающим между корпусом приборов и инструментом при пробое сетевого напряжения на корпус, корпуса приборов и инструментов должны быть заземлены;
- При включенном сетевом напряжении работы на задней панели должны быть запрещены;
- Все работы по устранению неисправностей должен производить квалифицированный персонал;

- Необходимо постоянно следить за исправностью электропроводки [22, 24].

Для профилактики организации действий при пожаре должен проводиться следующий комплекс организационных мер: должны обеспечиваться регулярные проверки пожарной сигнализации, первичных средств пожаротушения; должен проводиться инструктаж и тренировки по действиям в случае пожара; не должны загромождаться или блокироваться пожарные выходы; должны выполняться правила техники безопасности и технической эксплуатации электроустановок; во всех служебных помещениях должны быть установлены «Планы эвакуации людей при пожаре и других ЧС», регламентирующие действия персонала при возникновении пожара.

Для предотвращения пожара помещение с ПЭВМ должно быть оборудовано первичными средствами пожаротушения: углекислотными огнетушителями типа ОУ-2 или ОУ-5; пожарной сигнализацией, а также, в некоторых случаях, автоматической установкой объемного газового пожаротушения [25].

5.3 Экологическая безопасность

В данном разделе рассматривается воздействие на окружающую среду деятельности по разработке проекта, а также самого продукта в результате его реализации на производстве.

Разработка программного обеспечения и работа за ПЭВМ не являются экологически опасными работами, потому объект, на котором производилась разработка продукта, а также объекты, на которых будет производиться его использование операторами ПЭВМ относятся к предприятиям пятого класса, размер селитебной зоны для которых равен 50 м [26].

Непосредственно программный продукт, разработанный в ходе выполнения магистерской диссертации, не наносит вреда окружающей среде ни на стадиях его разработки, ни на стадиях эксплуатации. Однако, средства,

необходимые для его разработки и эксплуатации могут наносить вред окружающей среде.

Современные ПЭВМ производят практически без использования вредных веществ, опасных для человека и окружающей среды. Исключением являются аккумуляторные батареи компьютеров и мобильных устройств. В аккумуляторах содержатся тяжелые металлы, кислоты и щелочи, которые могут наносить ущерб окружающей среде, попадая в гидросферу и литосферу, если они были неправильно утилизированы. Для утилизации аккумуляторов необходимо обращаться в специальные организации, специализировано занимающиеся приемом, утилизацией и переработкой аккумуляторных батарей [27].

Люминесцентные лампы, применяющиеся для искусственного освещения рабочих мест, также требуют особой утилизации, т.к. в них присутствует от 10 до 70 мг ртути, которая относится к чрезвычайно-опасным химическим веществам и может стать причиной отравления живых существ, а также загрязнения атмосферы, гидросферы и литосферы. Сроки службы таких ламп составляют около 5-ти лет, после чего их необходимо сдавать на переработку в специальных пунктах приема. Юридические лица обязаны сдавать лампы на переработку и вести паспорт для данного вида отходов [27-29].

5.4 Безопасность в чрезвычайных ситуациях

В рабочей среде оператора ПЭВМ возможно возникновение следующих чрезвычайных ситуаций техногенного характера:

- Пожары и взрывы в зданиях и на коммуникациях;
- Внезапное обрушение зданий.

Среди возможных стихийных бедствий можно выделить метеорологические (ураганы, ливни, заморозки), гидрологические (наводнения, паводки, подтопления), природные пожары.

К чрезвычайным ситуациям биолого-социального характера можно отнести эпидемии, эпизоотии, эпифитотии.

Экологические чрезвычайные ситуации могут быть вызваны изменениями состояния, литосферы, гидросферы, атмосферы и биосферы в результате деятельности человека [19].

Наиболее характерной для объекта, где размещаются рабочие помещения, оборудованные ПЭВМ, чрезвычайной ситуацией является пожар.

Помещение для работы операторов ПЭВМ по системе классификации категорий помещений по взрывопожарной и пожарной опасности относится к категории Д (из 5-ти категорий А, Б, В1-В4, Г, Д), т.к. относится к помещениям с негорючими веществами и материалами в холодном состоянии [30].

Каждый сотрудник организации должен быть ознакомлен с инструкцией по пожарной безопасности, пройти инструктаж по технике безопасности и строго соблюдать его.

Запрещается использовать электроприборы в условиях, не соответствующих требованиям инструкций изготовителей, или имеющие неисправности, которые в соответствии с инструкцией по эксплуатации могут привести к пожару, а также эксплуатировать электропровода и кабели с поврежденной или потерявшей защитные свойства изоляцией. Электроустановки и бытовые электроприборы в помещениях по окончании рабочего времени должны быть обесточены (вилки должны быть вынуты из розеток). Под напряжением должны оставаться дежурное освещение и пожарная сигнализация. Недопустимо хранение легковоспламеняющихся, горючих и взрывчатых веществ, использование открытого огня в помещениях офиса.

Перед уходом из служебного помещения работник обязан провести его осмотр, закрыть окна, и убедиться в том, что в помещении отсутствуют источники возможного возгорания, все электроприборы отключены и выключено освещение. С периодичностью не реже одного раза в три года необходимо проводить замеры сопротивления изоляции токоведущих частей силового и осветительного оборудования.

Повышение устойчивости достигается за счет проведения соответствующих организационно-технических мероприятий, подготовки персонала к работе в ЧС [19].

Работник при обнаружении пожара или признаков горения (задымление, запах гари, повышение температуры и т.п.) должен:

- Немедленно прекратить работу и вызвать пожарную охрану по телефону «01», сообщив при этом адрес, место возникновения пожара и свою фамилию;
- Принять по возможности меры по эвакуации людей и материальных ценностей;
- Отключить от сети закрепленное за ним электрооборудование;
- Приступить к тушению пожара имеющимися средствами пожаротушения;
- Сообщить непосредственному или вышестоящему начальнику и оповестить окружающих сотрудников;
- При общем сигнале опасности покинуть здание согласно «Плану эвакуации людей при пожаре и других ЧС».

Для тушения пожара применять ручные углекислотные огнетушители (типа ОУ-2, ОУ-5), находящиеся в помещениях офиса, и пожарный кран внутреннего противопожарного водопровода. Они предназначены для тушения начальных возгораний различных веществ и материалов, за исключением веществ, горение которых происходит без доступа воздуха. Огнетушители должны постоянно содержаться в исправном состоянии и быть готовыми к действию. Категорически запрещается тушить возгорания в помещениях офиса при помощи химических пенных огнетушителей (типа ОХП-10) [21].

5.5 Правовые и организационные вопросы обеспечения безопасности

5.5.1 Правовые нормы трудового законодательства для рабочей зоны оператора ПЭВМ

Регулирование отношений между работником и работодателем, касающихся оплаты труда, трудового распорядка, особенности регулирования труда женщин, детей, людей с ограниченными способностями и проч., осуществляется законодательством РФ, а именно трудовым кодексом РФ.

Продолжительность рабочего дня не должна быть меньше указанного времени в договоре, но не больше 40 часов в неделю. Для работников до 16 лет – не более 24 часов в неделю, от 16 до 18 лет и инвалидов I и II группы – не более 35 часов.

Возможно установление неполного рабочего дня для беременной женщины; одного из родителей (опекуна, попечителя), имеющего ребенка в возрасте до четырнадцати лет (ребенка-инвалида в возрасте до восемнадцати лет). Оплата труда при этом производится пропорционально отработанному времени, без ограничений оплачиваемого отпуска, исчисления трудового стажа и других прав.

При работе в ночное время продолжительность рабочей смены сокращается на один час. К работе в ночную смену не допускаются беременные женщины; работники, не достигшие возраста 18 лет; женщины, имеющие детей в возрасте до трех лет, инвалиды, работники, имеющие детей-инвалидов, а также работники, осуществляющие уход за больными членами их семей в соответствии с медицинским заключением, матери и отцы-одиночки детей до пяти лет.

Организация обязана предоставлять ежегодный отпуск продолжительностью 28 календарных дней. Дополнительные отпуска предоставляются работникам, занятым на работах с вредными или опасными условиями труда, работникам имеющими особый характер работы, работникам

с ненормированным рабочим днем и работающим в условиях Крайнего Севера и приравненных к нему местностях.

В течение рабочего дня работнику должен быть предоставлен перерыв для отдыха и питания продолжительностью не более двух часов и не менее 30 минут, который в рабочее время не включается. Всем работникам предоставляются выходные дни, работа в выходные дни осуществляется только с письменного согласия работника.

Организация-работодатель выплачивает заработную плату работникам. Возможно удержание заработной платы только в случаях установленных ТК РФ ст. 137. В случае задержки заработной платы более чем на 15 дней, работник имеет право приостановить работу, письменно уведомив работодателя.

Законодательством РФ запрещена дискриминация по любым признакам и принудительный труд [32].

5.5.2 Влияние разработанного программного обеспечения на рабочий процесс

К мероприятиям, относящимся к компоновке рабочей зоны относятся работы по организации рабочего места пользователя, позволяющие наилучшим образом организовать деятельность работника, делая его работу максимально удобной и безопасной.

Требования к помещениям для работы с ПЭВМ регламентируются в соответствии с СанПиН 2.2.2/2.4.1340-03. В документе указаны нормы помещениям для работы с ПЭВМ, норма площади рабочего места с персональным компьютером составляет 4,5 м².

Основным направлением реализации разработанного продукта является применение его в качестве программного обеспечения для сегментации и распознавания машиночитаемых бланков, для специалистов занимающихся обработкой бланков ЕГЭ. Сам продукт не влияет на организацию рабочей зоны, но работа с ним позволит реорганизовать работу специалистов, что в свою

очередь повлияет на организацию рабочей зоны. Это может быть охарактеризовано с помощью следующих факторов:

- Сокращение времени обработки бланков, что уменьшает нагрузки специалиста, способствует увеличению частоты смены вида деятельности работника, как оператора ПЭВМ, и тем самым снижает влияние психофизических вредных факторов;
- Сокращение умственных и зрительных нагрузок на специалиста за счет автоматизации обработки бланков;
- Снижение общего времени нагрузок на специалиста, что уменьшает влияние вредных и опасных факторов производства.

Все перечисленные факторы, способствующие увеличению степени безопасности и удобства работы пользователей разработанной систем, что в свою очередь облегчает работу и положительно сказывается на производительности труда.

ЗАКЛЮЧЕНИЕ

Распознавание рукописного текста является важной задачей при обработке документов. Одним из самых распространенных методов машинного обучения при решении данной задачи являются нейронные сети.

В ходе работы был проведен сравнительный анализ использования сверточной и простой нейронной сети, типа персептрон для распознавания рукописных символов. Проведенные исследования показали, что использования сверточной сети для решения поставленной задачи, будет наиболее оптимальным, так как она имеет меньше количество настраиваемых параметров и выше точность распознавания.

Опытным путем была выявлена наиболее подходящая архитектура сверточной нейронной сети для решения поставленной задачи, исследованы алгоритмы оптимизации на основе метода градиентного спуска и виды функции активации нейронов. Все тесты проводились с использованием базы рукописных символов MNIST и базы данных рукописных символов русского алфавита.

Были исследованы алгоритмы сегментации текстовых изображений, на основе пороговых алгоритмов и методов связанных компонент.

Была разработана архитектура программы, включающая в себя основные классы и функции необходимые для реализации выбранных алгоритмов сегментации и распознавания машиночитаемых бланков. Была осуществлена программная реализация предложенного алгоритма. Максимально достигнутая точность распознавания составила 94,1% для изображений из тестовой выборки.

Так же был произведен анализ влияния разработанной системы на конечных пользователей и определен уровень научно-технического эффекта данной работы.

СПИСОК ПУБЛИКАЦИЙ СТУДЕНТА

1. Попова, Е.С. Программное обеспечение сегментации текстовых изображений / Е.С. Попова, науч. рук. Е.С. Болотова // Молодежь и современные информационные технологии Том I: сборник трудов XIII Международной научно-практической конференции студентов, аспирантов и молодых ученых. – Томск: Изд-во ТПУ, 2015. – С. 156-157
2. Попова, Е.С. Проектирование Сверточных нейронных сетей для распознавания рукописных символов / Е.С. Попова, науч. рук. Е.С. Болотова // Молодежь и современные информационные технологии Том II: сборник трудов XIV Международной научно-практической конференции студентов, аспирантов и молодых ученых. – Томск: Изд-во ТПУ, 2016. – С. 84-85

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Хайкин С. Нейронные сети: полный курс. М.: Вильямс, 2006. - 1104 с.
2. Маркелов А.А. Алгоритмы и программная система классификации полутоновых изображений на основе нейронных сетей: Диссертация на соискание ученой степени кандидата технических наук / А. А. Маркелов. – Томск, 2007.
3. Гонсалес, Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс. – М.: Техносфера, 2005. – 1072 с.
4. В.Г. Спицын, Интеллектуальные системы: учебное пособие / В.Г. Спицын, Ю.Р. Цой; Томский политехнический университет. – Томск: Изд-во Томского политехнического университета, 2012. – 176 с.
5. LeCun, Y. Efficient BackProp in Neural Networks: Tricks of the trade / Y.LeCun, L. Bottou, G. Orr, K. Muller – Springer, 1998.
6. LeCun, Y. Scaling learning algorithms towards AI / Y.LeCun, Y. Bengio – MIT Press, 2007.
7. Шапиро Л. Компьютерное зрение / Л. Шапиро, Дж. Стакан, 2006. – 53 с.
8. Систематические методы распознавания образов/ Ю. Лившиц, 2005. – 12 с.
9. Грузман И.С., Киричук В.С., Косых В.П., Перетягин Г.И., Спектор А.А. Цифровая обработка изображений в информационных системах: Учеб. пособие. – Новосибирск.: Изд-во НГТУ, 2003. – 352 с.
10. Солдатова О.П., Гаршин А.А. Применение сверточной нейронной сети для распознавания рукописных цифр. Компьютерная оптика. – 2010. – Том 34, №2. – с. 252-260 – ISSN 0134-2452.
11. LeCun Y. The MNIST database of handwritten digits / Y.LeCun. [Электронный ресурс] Точка доступа: <http://yann.lecun.com/exdb/mnist> (Дата обращения 11.02.2017)
12. Онлайн-курсы от ведущих университетов [Электронный ресурс] Точка доступа: <https://www.coursera.org> (дата обращения 15.05.2017)

13. Mike O'Neill. Neural Network for Recognition of Handwritten Digits. [Электронный ресурс] Точка доступа: <https://www.codeproject.com/Articles/16650/Neural-Network-for-Recognition-of-Handwritten-Digi> (Дата обращения 12.02.2017)
14. Поршнева С.В., Левашкина А. О. Универсальная классификация алгоритмов сегментации изображений // Журнал научных публикаций аспирантов и докторантов [Электронный ресурс] — Электронный научный журнал – 2006. – Режим доступа: <http://jurnal.org/articles/2008/inf23.html> (дата обращения: 05.03.2017).
15. Введение в цифровую обработку изображений: лекция 3. [Электронный ресурс] – 2011. – Режим доступа: <http://cvbeginner.blogspot.ru/2011/09/3.html> (дата обращения: 20.03.2017).
16. Охрана труда. Основы безопасности жизнедеятельности // www.Grandars.ru. 2016. [Электронный ресурс] Точка доступа: <http://www.grandars.ru/shkola/bezopasnost-zhiznedeyatelnosti/ohrana-truda.html> (дата обращения: 22.04.2017).
17. ГОСТ 12.0.003-74. Система стандартов безопасности труда. Опасные и вредные производственные факторы. Классификация // Библиотека ГОСТов. 2016. [Электронный ресурс] Точка доступа: <http://vsegost.com/Catalog/41/41131.shtml> (дата обращения: 22.04.2016).
18. Ефремова О. С. Требования охраны труда при работе на персональных электронно-вычислительных машинах. – 2-е изд., перераб. и доп. – М.: Издательство «Альфа-Пресс», 2008. – 176 с.
19. Назаренко О. Б. Безопасность жизнедеятельности: учебное пособие / О. Б. Назаренко, Ю. А. Амелькович; Томский политехнический университет. – 3-е изд., перераб. и доп. – Томск: Изд-во Томского политехнического университета, 2013. – 178 с.
20. СанПиН 2.2.4.548-96. Санитарные правила и нормы. Гигиенические требования к микроклимату производственных помещений // Библиотека гостей и нормативов. 2016. [Электронный ресурс] Точка доступа:

http://ohranatruda.ru/ot_biblio/normativ/data_normativ/5/5225/ (дата обращения: 23.03.2017).

21. Белов С. В. Безопасность жизнедеятельности и защита окружающей среды (техносферная безопасность): учебник / С. В. Белов. – 2-е изд., испр. и доп. – М.: Издательство Юрайт, 2011. – 680 с.
22. СанПиН 2.2.2/2.4.1340-03. Санитарно-эпидемиологические правила и нормы. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы // Библиотека гостей и нормативов. 2016. [Электронный ресурс] Точка доступа: http://www.ohranatruda.ru/ot_biblio/normativ/data_normativ/39/39082/#i72870 (дата обращения: 23.04.2017).
23. СП 52.13330.2011. Естественное и искусственное освещение. Актуализированная редакция СНиП 23-05-95 // Докипедия. 2016. [Электронный ресурс] Точка доступа: <http://dikipedia.ru/document/5147250> (дата обращения: 23.04.2016).
24. ГОСТ Р 12.1.019-2009 ССБТ. Электробезопасность. Общие требования и номенклатура видов защиты // Электронный фонд правовой и нормативно-технической документации. 2010. [Электронный ресурс] Точка доступа: <http://docs.cntd.ru/document/gost-r-12-1-019-2009-ssbt> (дата обращения: 24.04.2017).
25. СНиП 21-01-97. Пожарная безопасность зданий и сооружений // Библиотека гостей и нормативов. 2016. [Электронный ресурс] Точка доступа: http://www.ohranatruda.ru/ot_biblio/normativ/data_normativ/2/2107/ (дата обращения: 24.04.2017).
26. СанПиН 2.2.1/2.1.1.1200-03. Санитарно-эпидемиологические правила и нормативы. Санитарно-защитные зоны и санитарная классификация предприятий, сооружений и других объектов // Библиотека гостей и нормативов. 2016. [Электронный ресурс] Точка доступа: http://ohranatruda.ru/ot_biblio/normativ/data_normativ/11/11774/ (дата обращения: 24.04.2017).

27. СанПиН 2.1.7.1322-03. Санитарно-эпидемиологические правила и нормативы. Гигиенические требования к размещению и обезвреживанию отходов производства и потребления. 2.1.7. Почва, очистка населённых мест, бытовые и промышленные отходы, санитарная охрана почвы // Библиотека гостов и нормативов. 2016. [Электронный ресурс] Точка доступа: http://ohranatruda.ru/ot_biblio/normativ/data_normativ/11/11774/ (дата обращения: 02.05.2017).
28. Постановление Правительства РФ от 03.09.2010 N 681 (ред. от 01.10.2017) "Об утверждении Правил обращения с отходами производства и потребления в части осветительных устройств, электрических ламп, ненадлежащие сбор, накопление, использование, обезвреживание, транспортирование и размещение которых может повлечь причинение вреда жизни, здоровью граждан, вреда животным, растениям и окружающей среде // Консультант Плюс. 2015. [Электронный ресурс] Точка доступа: http://www.consultant.ru/document/cons_doc_LAW_104420/e1b31c36ed1083efeb6cd9c63ed12f99e2ca77ed/#dst100007 (дата обращения: 02.05.2017).
29. Энергосбережение в компьютерном мире // НВП. 2008. [Электронный ресурс] Точка доступа: http://www.hwp.ru/articles/Energoberezhnie_v_kompyuternom_mire_SNast_1_osnovnie_tendentsii/?SHOWALL_1=1 (дата обращения: 24.04.2017).
30. НПБ 105-03 Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности // Электронный фонд правовой и нормативно-технической документации. 2016. [Электронный ресурс] Точка доступа: <http://docs.cntd.ru/document/1200032102> (дата обращения: 24.04.2017).
31. ППБ 01–03. Правила пожарной безопасности в Российской Федерации. – М.: Министерство Российской Федерации по делам гражданской обороны, чрезвычайным ситуациям и ликвидации последствий стихийных бедствий, 2003.

32. Трудовой кодекс Российской Федерации" от 30.12.2001 N 197-ФЗ (ред. от 30.12.2017) // Консультант Плюс. 2015. [Электронный ресурс] Точка доступа:

http://www.consultant.ru/document/cons_doc_law_34683/?utm_campaign=law_doc&utm_source=google.adwords&utm_medium=cpc&utm_content=Labor%20Code&gclid=CjwKEAjwtgPe4BRcB66GG8PO69QkSJAC4EhHhU-5yAFZCJfmzkTLNGnrpgHHAYFPhhPzRo-sZGWmqnBoCPynw_wcB (дата обращения: 25.04.2017).

Приложение А

(обязательное)

Раздел 1

Review of recognition and segmentation algorithms

Студент:

Группа	ФИО	Подпись	Дата
8ИМ5Б	Попова Екатерина Сергеевна		

Консультант – лингвист кафедры ИЯИК:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Морозов Василий Сергеевич			

Консультант кафедры ИСТ:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Мирошниченко Е.А.	К.Т.Н.		

1. REVIEW OF RECOGNITION AND SEGMENTATION ALGORITHMS

1.1. Artificial neural networks

Attempts to reproduce the ability of the biological nervous system to learn and correct mistakes led to the creation of artificial neural networks. Artificial neural networks are a family of models built on the principle of the organization and functioning of biological neural networks - nerve cell networks of a living organism.

To date, neural networks have been successfully applied to solve problems of classification or clustering of multidimensional data.

Despite the existence of differences, certain types of neural networks have several common features. First, the basis of each neural network is relatively simple, in most cases - the same type, elements (formal neurons), simulating the work of neurons of the brain.

The second common feature of the inherent neural networks is the principle of parallel signal processing, which is achieved by combining a large number of neurons into layers. In this case, neurons of different layers of the connection in a certain way, and in some configurations of neural networks, neurons of one layer can also be connected together, and the processing of interaction of all neurons is conducted layer by layer. Figure 1 shows the bond structure of a single-layer, fully-connected neural network.

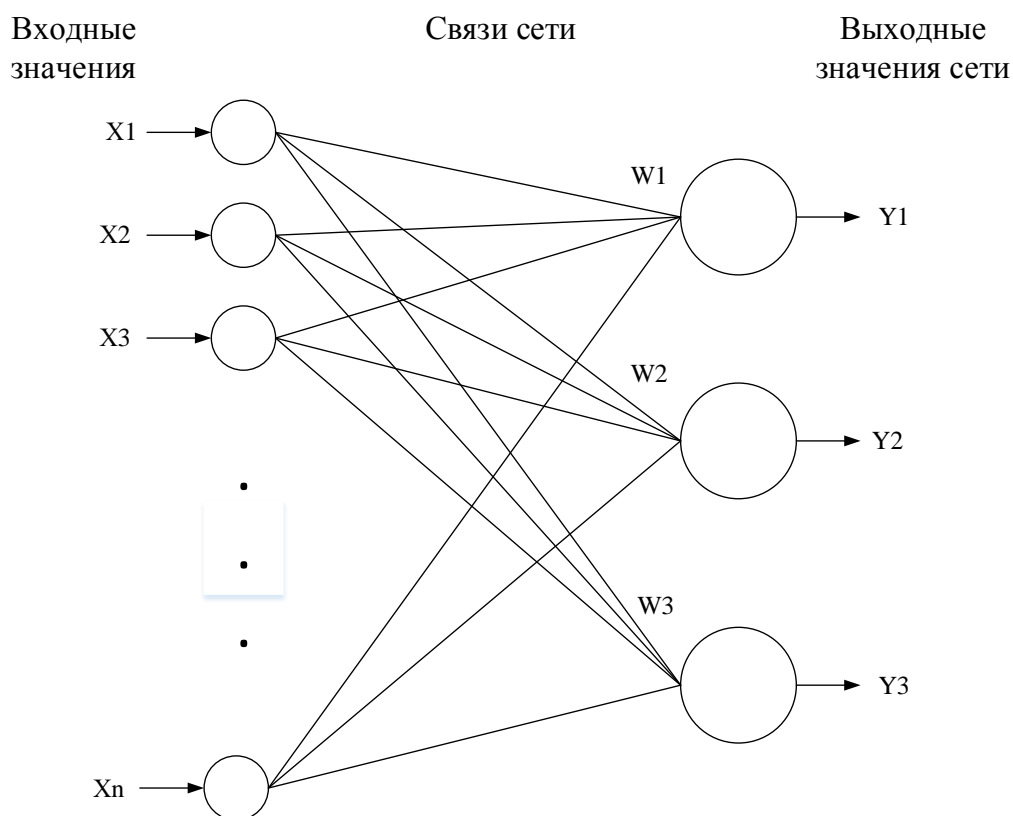


Figure 1 - Structure of connections of a single layer neural network

Theoretically, the number of layers and the number of neurons in each layer can be arbitrary, but in fact it is limited by the resources of the computer on which the calculations will be performed. The more complex a neural network, the more large-scale task it is able to solve.

The process of functioning of the neural network depends on the values of the synaptic connections (network weights), therefore, after determining the structure of the neural network, it is necessary to find the optimal values of all variables of the weight coefficients to solve a problem. This stage is called the training of a neural network, and on how well it will be performed, the network's ability to solve its goals during operation depends on it.

1.1.1. Model of an artificial neuron

At the heart of any neural network is the model of the formal neuron, represented in Figure 2, which is the basic functional unit of the neural network.

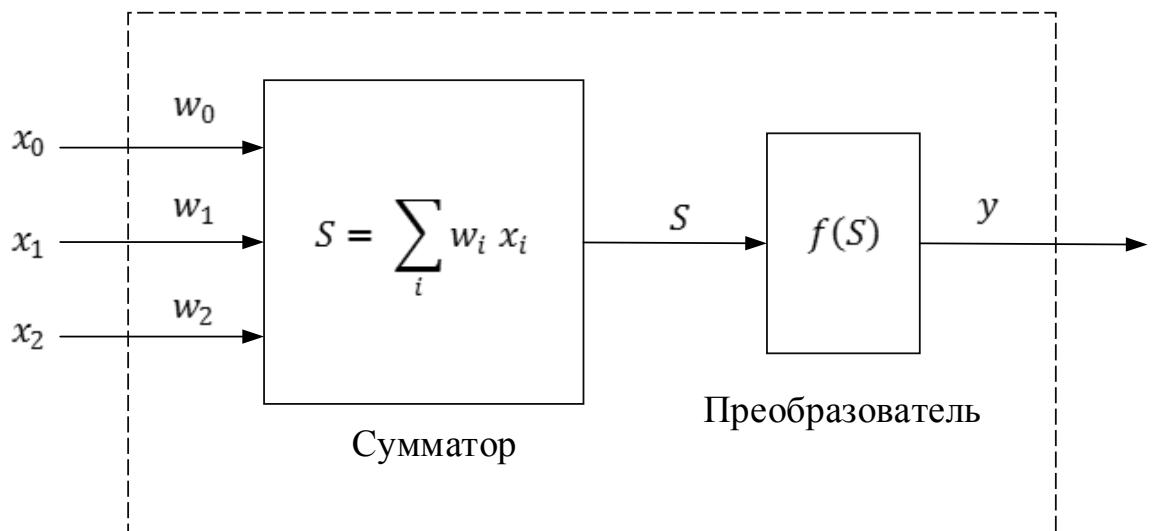


Figure 2 – Two-layer neural network

In this model, the neuron receives input signals ($x_0 \dots x_n$) that pass through synapse bonds, the weight of which ($w_0 \dots w_n$, respectively) emulates a different throughput of synapses in natural neurons. After receiving the signals, they are processed by an adder, whose task is to obtain a linear combination of all input signals, which in turn serves as an argument for the transfer function-converter. It is she who determines the value of the signal that will be received at the output of the neuron. The result is sent to the only output - the axon. Such artificial neurons are networked by connecting the outputs of one neuron to the inputs of others.

Thus, a formal neuron consists of 3 types of elements:

- multipliers;
- adder;
- converter.

The synapse characterizes the strength of the connection between the two neurons. The adder adds the input signals, previously multiplied by the corresponding weights. The converter implements the function of one argument - the output of the adder. This function is called the activation function or transfer function of a neuron. Based on this description, the mathematical model of a neuron can be represented as follows:

$$y = f(S), \quad (1)$$

$$S = \sum_{i=1}^n w_i x_i + b, \quad (2)$$

where x - components of the input signal vector, w - weights of the input signals of the neuron, b - displacement of the neuron, and y - output signal of the neuron.

1.1.2. Types of activation function

The activation functions represented in the formulas as $f(S)$ determine the output signal of the neuron depending on the induced local field S (weighted sum of inputs). There are three main types of activation function.

1. *The function of a unit jump*, or a threshold function. This function is described as follows:

$$f(S) = \begin{cases} 0, & S < \Theta \\ 1, & S \geq \Theta \end{cases}$$

2. *Piecewise-linear function*. The piecewise-linear function described by the following expression:

$$f(S) = \begin{cases} 1, & S \geq \frac{1}{2} \\ |S|, & +\frac{1}{2} > S > -\frac{1}{2} \\ 0, & S \leq -\frac{1}{2} \end{cases}$$

3. *Sigmoid function*. The sigmoid function is one of the most common functions used to create an artificial neural network. This is a rapidly increasing function that maintains a balance between linear and non-linear behavior. An example of a sigmoid function is the logosigmoid function given by the following expression:

$$f(S) = \frac{1}{1 + e^{(aS)}}$$

where a – the slope parameter of the sigmoid function, the change of which will allow us to construct a function with different slopes. The sigmoid function takes an infinite set of values in the range from 0 to 1.

However, sometimes an activation function is required that has a range of values from -1 to +1, in which case the activity function must be symmetric with respect to the origin. In particular, the threshold function, in this case, can be defined as follows:

$$f(S) = \begin{cases} 1, & S > 0 \\ 0, & S = 0 \\ -1, & S < 0 \end{cases}$$

This function is usually called a *signum*. In this case, the sigmoidal function will have the form of a *hyperbolic tangent*:

$$f(S) = \frac{e^{aS} - e^{-aS}}{e^{aS} + e^{-aS}},$$

$$f(S) = A \tanh(aS)$$

where $f(S)$ – the desired value of the element, a – weighted sum of the signals of the previous layer, A , S – are parameters of the activating function.

The hyperbolic tangent has some advantages:

- Symmetric activation functions, such as hyperbolic tangent, provide faster convergence than the standard logistic function;
- the function has a continuous first derivative;
- The function has a simple derivative that can be computed through its value, which saves calculations.

The activation functions of ReLU and Softmax used in the work are described below, at the moment they have the best results when learning neural networks.

4. *Straightened linear activation function (rectified linear unit, ReLU)*. It is known that neural networks are capable of approximating an arbitrarily complex function if there are enough layers in them, and the activation function is nonlinear. Activation functions such as sigmoid or tangential are non-linear, but lead to problems with fading or increasing gradients. However, we can use a much simpler version - a rectified linear activation function. The ReLU function is a rectified linear function and at the moment it is considered much simpler and more efficient from the point of view of computational complexity by the variant of the transfer function, it is one of the last successes in the field of methods for tuning deep neural networks:

$$f(S) = \max(0, S) = \begin{cases} 0, & \text{при } S < 0 \\ S, & \text{при } S \geq 0 \end{cases}$$

Figure 3 shows the graph of the function ReLU and its first derivative.

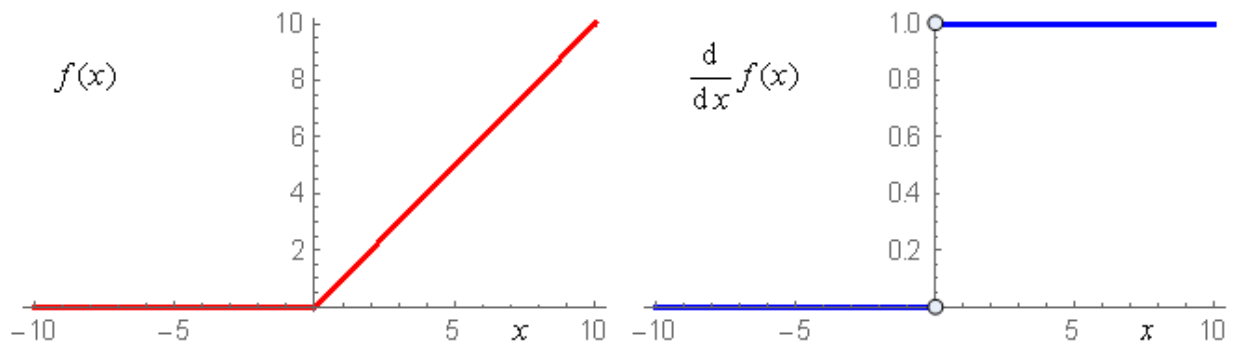


Figure 3 - Graph ReLU function

Its derivative is either one or zero, and therefore no growth or attenuation of the gradients can occur. Moreover, the use of this function leads to a thinning of the balance.

Today there is a family of different modifications of ReLU solving the reliability problems of this transfer function when passing through the neuron of large gradients: Leaky ReLU, Parametric ReLU, Randomized ReLU.

5. Softmax activation function. This activation function is designed to convert any vector with real values into a probability vector and is defined for the i -th neuron as follows:

$$z_i = \frac{\exp(y_i)}{\sum_{j=1}^n \exp(y_j)} ,$$

where z_i is the desired output value of the i -th neuron, y_i is the initial value of the output i -th neuron.

It is clear from the formula that the output of each neuron depends on the totalizers of all other neurons of the softmax group, and the sum of the output values of the whole group is equal to one. The advantage of this function is that the partial derivative of the i -th neuron in its adder is equal to:

$$\frac{\partial z_i}{\partial y_i} = z_i(1 - z_i) .$$

Figure 4 shows the graph of the Softmax function.

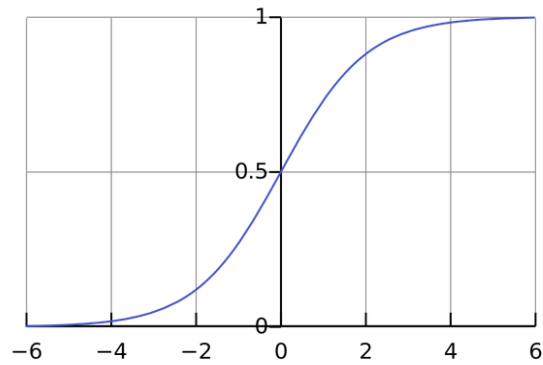


Figure 4 - Graph of the Softmax function

1.1.3. Loss function

In this section, one of the most common loss functions used in this paper, cross-entropy, will be considered.

Since the activation function in the output layer will use the softmax activation function, which converts any input vector into a probability vector, a correct measure is needed to compare two probability distributions. As a measure, cross-entropy will be used:

$$C = - \sum_{j=1}^n t_j \log(y_j) ,$$

where t_j is the required output for the current learning example, y_j is the real output of the neural network.

1.1.4. Algorithm for learning neural network

The most common algorithm for learning neural networks is the method of gradient descent or as it is also called the back propagation method of error, and its modifications. This algorithm refers to the methods of teaching with the teacher.

In the process of learning a multilayer neural network using the algorithm for back propagation of an error, it repeatedly presents a predetermined set of training examples. One full cycle of presenting a complete set of examples of training is called an era. The learning process is carried out from era to era, until the synoptic weights

and threshold levels (mixing) stabilize, and the network error on the entire training set does not converge to a certain minimum value.

It is advisable to randomly change the order of presentation of examples of training for different epochs. This principle of presentation of images makes the search in the space of weights stochastic, preventing the potential possibility of the appearance of closed cycles during the evolution of the synoptic weights.

The algorithm for back propagation of the error can be implemented in two ways:

Serial mode. The sequential training mode based on the error back propagation algorithm is also sometimes called the stochastic gradient descent. In this mode, the balance is adjusted after each example is given. This is the most commonly used mode. The first example of this epoch is presented to the network, after which the synoptic weights of the network are adjusted. After this network, the second pair from the era is presented, and the synoptic weights are corrected. This process is repeated until the network finishes processing the last training pair of this era

Batch mode. In the batch mode of learning, based on the algorithm for back propagation of the error, the adjustment of the weights is carried out after the submission to the network of all training examples of one epoch

Mini-batch. Between these two types of back propagation method of error (gradient descent) there is a compromise, sometimes called "mini-batch". In this case, the adjustment of the synoptic weights of the network occurs after a small number of training samples.

From the point of view of real-time processes, a sequential mode is more preferable than a batch mode, since it requires less internal storage for each synoptic connection. Moreover, by presenting training examples in a random order (in the process of successive correction of weights), the search in the space of weights can be made really stochastic. This, in turn, minimizes the possibility of stopping the algorithm at the point of any local minimum.

Algorithm of the method of back propagation of error for batch mode of learning

This method of learning a multilayer neural network is called a generalized delta rule or an error backpropagation rule. The method was proposed in 1986 by Rumelhart, McClelland and Williams.

This algorithm is used to minimize the deviation of the real values of the output signals of the neural network from the required ones. As a function of estimating the work of ANN, we will consider the method of least squares (quadratic error):

$$E(w) = \frac{1}{2} \sum_{i,k} (f_{i,k} - y_{i,k}^{(T)})^2$$

where $f_{i,k}$ - value of the output signal of the k -th output neuron of the network when the i -th set of training data is fed to its inputs, $y_{j,k}^{(T)}$ - required value of the output signal of the k -th output neuron for i -th data set for training. Summation is carried out over all neurons of the output layer and over all data sets from the training sample. Training INS is aimed at minimizing the function $E(w)$.

Minimization by the method of gradient descent provides the adjustment of the weight coefficients as follows:

$$\Delta w_{ij}^{(q)} = \eta \frac{\partial E}{\partial w_{ij}}$$

where $\Delta w_{ij}^{(q)}$ is the magnitude of the change in the bond weight connecting the i th neuron ($q-1$) of the layer with the j th neuron of the layer q ; η - training speed coefficient, $0 < \eta < 1$. Thus, the weight of the link varies in proportion to its contribution to the neuron error value, for which this connection is the input, since The partial derivative by weight $\partial E / \partial w_{ij}$ shows the dependence of the rate of change of the error function E on the change in this weight.

The change in connection weight is determined as follows:

$$\Delta w_{ij}^{(q)} = \eta \delta_j x_i$$

where δ_j is the error value of the j -th neuron in the layer q , x_i is the value of the i -th input signal for the j -th neuron of the layer q . This formula is also applicable for setting neuron displacements, but instead of x_i it is necessary to substitute «1».

Thus, the value of the neuron error is proportional to its «influence» on the error values of neurons of the next layer, as well as the rate of change of its output signal for the k -th set of training data.