

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ

Заведующий кафедрой

_____ Г. М. Цибульский

подпись

« _____ » _____ 2020 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.02 — Информационные системы и технологии

Разработка информационной системы для анализа рентгеновских снимков

Руководитель _____ доцент, канд. техн. наук А. В. Пятаева
подпись, дата

Выпускник _____ А. А. Русскина
подпись, дата

Красноярск 2020

Продолжение титульного листа бакалаврской работы по теме «Разработка информационной системы для анализа рентгеновских снимков»

Нормоконтролер

подпись, дата

А. В. Пятаева

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ

Заведующий кафедрой

_____ Г.М. Цибульский

подпись

« ____ » _____ 2020 г.

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
в форме бакалаврской работы

Студенту Русскиной Алене Алексеевне

Группа КИ16-12Б, направление 09.03.02 «Информационные системы и технологии».

Тема выпускной квалификационной работы «Разработка информационной системы для анализа рентгеновских снимков».

Утверждена приказом по университету № 6506/с от 22.05.2020

Руководитель ВКР А.В. Пятаева кандидат технических наук, доцент кафедры систем искусственного интеллекта ИКИТ СФУ.

Исходные данные для ВКР: возможности применения сверточных нейронных сетей для распознавания пневмонии на рентгеновском снимке.

Перечень разделов ВКР:

- введение;
- анализ предметной области
- выводы к главе 1;
- выбор и модификация нейронной сети
- выводы по главе 2;
- экспериментальные исследования
- заключение;
- список использованных источников;
- приложение А (отчет «Антиплагиат»).
- приложение Б (плакаты презентации).

Перечень графического материала: презентация «Разработка информационной системы для анализа рентгеновских снимков».

Руководитель ВКР

подпись

А. В. Пятаева

Задание принял к исполнению

подпись

А. А. Русскина

« ___ » _____ 2020 г

График

График выполнения выпускной квалификационной работы студентом направления 09.03.02 «Информационные системы и технологии».

График выполнения выпускной квалификационной работы приведен в таблице 1.

Таблица 1 — График выполнения этапов ВКР

Наименование этапа	Срок выполнения этапа	Результат выполнения этапов	Примечание Руководителя (отметка о выполнении этапа)
Ознакомление с целью и задачами работы	30.01-10.02	Краткое эссе по теме ВКР	Выполнено
Сбор литературных источников	11.02-18.02	Список использованных источников	Выполнено
Анализ собранных источников литературы	19.02-3.03	Реферат о проблемно предметной области	Выполнено
Уточнение и обоснование актуальности цели и задач ВКР	4.03-13.03	Окончательная формулировка цели и задач ВКР	Выполнено
Решение первой задачи ВКР	14.03-02.04	Доклад и презентация по решению первой задачи	Выполнено
Решение второй задачи ВКР	03.04-27.04	Доклад и презентация по решению второй задачи	Выполнено
Решение третьей задачи ВКР	28.04-22.05	Доклад и презентация по третьей задаче ВКР	Выполнено
Подготовка доклада и презентации по теме ВКР	22.05-05.06	Доклад с презентацией по теме ВКР	Выполнено
Компоновка отчета по результатам решения задач ВКР	06.06-11.06	Отчет по результатам решения задач ВКР	Выполнено
Первичный нормоконтроль (Н/К)	11.06-26.06	Пояснительная записка, презентация ВКР	Выполнено

Наименование этапа	Срок выполнения этапа	Результат выполнения этапов	Примечание Руководителя (отметка о выполнении этапа)
Предварительная защита результатов ВКР	11.06-26.06	Доклад с презентацией по теме ВКР	Выполнено
Вторичный нормоконтроль (Н/К)	27.06	Пояснительная записка, презентация ВКР	Выполнено
Итоговый нормоконтроль (Н/К)	29.06	Пояснительная записка, презентация ВКР	Выполнено
Защита ВКР	02.07-03.07	Пояснительная записка, доклад и презентация по результатам бакалаврской работы	

Руководитель ВКР

подпись

А. В. Пятаева

Студент гр. КИ16-12Б

подпись

А. А. Русскина

« ___ » _____ 2020г

СОДЕРЖАНИЕ

Введение.....	9
Глава 1 Анализ предметной области.....	10
1.1 Диагностика пневмонии.....	10
1.2 Распознавание пневмонии с помощью сверточных нейросетей.....	12
1.3 Основы теории сверточных нейронных сетей.....	14
Выводы по главе 1.....	17
Глава 2 Разработка нейронной сети.....	18
2.1 Препроцессинг изображений.....	18
2.1.1 Масштабирование.....	18
2.1.2 Устранение шума на изображениях.....	19
2.1.3 Аугментация данных.....	21
2.2 Архитектура нейронной сети.....	22
2.2.1 Готовые нейронные сети.....	22
2.2.2 Составная нейронная сеть.....	26
2.2.2.1 Global Average Pooling.....	27
2.2.2.2 Дополнительный полносвязный слой.....	29
2.2.2.3 Пакетная нормализация.....	29
2.2.2.4 Dropout.....	31
2.3 Обучение нейронной сети.....	33
2.3.1 Функция потерь.....	33
2.3.2 Оптимизатор.....	35
2.3.3 Метрики.....	37
2.4 Результаты работы составной нейронной сети.....	38
Выводы по главе 2.....	39
Глава 3 Экспериментальные исследования.....	40
3.1 Набор данных.....	40
3.2 Результаты экспериментальных исследований.....	41
Заключение.....	43
Список сокращений.....	44

Список использованных источников	45
Приложение А	48
Приложение Б.....	49

ВВЕДЕНИЕ

В данной работе рассматривается применение сверточных нейронных сетей для распознавания пневмонии на рентгеновском снимке. За последние годы уже открыто немало новых возможностей нейронных сетей, а работы в данной области вносят существенный вклад в промышленность, науку и технологии. Широкий круг задач, решаемый нейронными сетями, не позволяет в настоящее время создавать универсальные, мощные сети, вынуждая разрабатывать специализированные сети, функционирующие по различным алгоритмам. Тем не менее, тенденции развития нейросетей растут с каждым годом.

Предметом исследования являются снимки рентгенограммы, которые являются первичным способом выявления пневмонии. Распознавание данной болезни методами машинного обучения является актуальной задачей в современном мире, так как ежегодно по всему миру болеют миллионы людей, а также существует нехватка специалистов, которые обрабатывают снимки рентгенограммы, особенно это касается развивающихся стран.

Целью работы является распознавание пневмонии на рентгеновском снимке с помощью сверточных нейросетей.

Поставлены следующие задачи:

1. Выполнить анализ современных способов распознавания пневмонии по рентгеновским снимкам;
2. Предварительно обработать данные;
3. Выбрать и модифицировать архитектуру;
4. Провести экспериментальные исследования;

Глава 1 Анализ предметной области

1.1 Диагностика пневмонии

Пневмония является 2-м наиболее часто выявляемым инфекционным заболеванием во всем мире. В 2016 году по всему миру было зарегистрировано 197 млн случаев пневмонии по всему миру, из них 1,2 млн человек умерло. Особенно данное заболевание опасно для маленьких детей и пожилых людей. По данным Всемирной организации здравоохранения (ВОЗ) [9], в 2017 году пневмония стала причиной 15% смертей в возрасте до пяти лет. Большинство детей умерли в бедных развивающихся странах, где им не обеспечили нужный уход и лечение.

В статье «Нехватка медицинских работников в развивающихся странах – Африка», авторами которой являются Сараладеви Найкер, Джейкоб Плангерулеи, Джон Б Иствуд, говорится о том, что миграции работников здравоохранения, серьезно влияет на устойчивость системы здравоохранения во многих развивающихся странах [3]. Конференция ООН 1998 года исследования и развития ВОЗ подсчитали, что 56% всех мигрирующих врачей перетекают из развивающихся в развитые страны в то время, как только 11% мигрируют в противоположном направлении. Врачи и медсестры являются опорой любой системы медицинской помощи. В странах, уже сильно лишенных профессионалов, потеря каждого имеет серьезные последствия для здоровья граждан.

Пневмония происходит в альвеолярных образованиях легких. Это состояние вызвано различными инфекционными заболеваниями, включая вирусы и бактерии. Одним из первичных способов выявления пневмонии является рентген, поэтому не стоит пренебрегать данной процедурой – своевременное исследование поможет предотвратить угрозу не только для здоровья, но и для жизни больного.

Рентгенография грудной клетки на рисунке 1 является одной из наиболее информативных методов диагностики заболеваний дыхательной системы, включая пневмонию. Она позволяет не только выявить патологический процесс, но также оценить эффективность проводимого лечения. Как же выглядят рентгенологические признаки пневмонии на снимках, и как проводится рентгенография при данной патологии?



Рисунок 1 - Пример рентгенограммы

Рентгеновское исследование включает сканирование внутренних органов с помощью специального аппарата, который излучает электромагнитные волны определенной длины. Проходя сквозь ткани человеческого тела, они оставляют на поверхности пленки следы, из которых формируется изображение соответствующего органа. По снимку можно определить патологические изменения и получить полное представление о разных заболеваниях, в том числе воспалительных процессов в респираторных органах.

Далее после получения снимка, его должен изучить специалист, и на основе полученных данных и результатов других исследований сделать соответствующие выводы и поставить диагноз.

Первый признак пневмонии на рентгеновском снимке – появление очагов затемнения с неровными контурами в разных частях легкого, которые могут иметь разный размер, от 3-4-х до 12 мм.

Тени различают по внешнему виду (круглые, овальные кольцевидные) и интенсивности окраски – чем темнее пятно, тем сильнее выражен патологический процесс.

При поражении лимфатических узлов и нарушении кровоснабжения органа могут наблюдаться изменения корней легких, а если болезнь затронула плевру – нарушение в рисунке куполов диафрагмы [11]. В остальных проявлениях пневмонии зависят от стадии, формы и клинических особенностей заболевания:

1. Очаговая форма. На рентгене отображаются небольшие (1-1,5 см) тени со слабой или умеренной интенсивностью окраски, неоднородной структурой и нечеткими границами. Очаги поражения могут быть единичными или множественными, а в некоторых случаях они сливаются в одно большое пятно. Корни легких расширены, причем нарушения нормального рисунка органа могут сохраняться в течение нескольких дней после выздоровления.

2. Крупозная пневмония. Наблюдаются изменения нормального легочного рисунка, жидкость в полости плевры, признаки инфильтрации одной из долей легкого, расширение корней. По мере развития воспалительного процесса выраженность изменений и интенсивность окраски затемнений усиливается.

3. Интерстициальная форма. На снимке заметно уплотнение корней легких и другие изменения, которые формируют выраженный рисунок, напоминающий ветви дерева.

4. Абсцедирующая пневмония. Проявляется обширным затемнением пораженной области, признаками утолщения плевры и наличием полостей разного размера, наполненных жидкостью.

5. Аспирационная форма. Рентген характеризуется треугольными пятнами с однородной структурой, светлыми очагами и приподнятой диафрагмой.

1.2 Распознавание пневмонии с помощью сверточных нейросетей

В статье [13] авторы предлагают новую систему глубокого обучения для обнаружения пневмонии с использованием концепции трансферного обучения. При таком подходе функции из изображений извлекаются с использованием

различных моделей нейронных сетей, прошедших предварительную подготовку в ImageNet, которые затем подаются в классификатор для прогноза. Они подготовили пять разных моделей и проанализировали их производительность. Их скомбинированная модель достигла точности 96,4% с отзывом 99,62%. Авторы заметили, что производительность может быть улучшена за счет увеличения размера набора данных с использованием первоначальных экземпляров. Авторы говорят о том, что их результаты подтверждают мнение о том, что методы глубокого обучения могут быть использованы для упрощения диагностики пневмонии.

В другом исследовании [14] предлагается сверточная модель нейронной сети, обученная с нуля, чтобы классифицировать и выявлять наличие пневмонии из коллекции образцов рентгеновских снимков грудной клетки. Предлагаемая архитектура состоит из свертки, макс-пула и слоев классификации, объединенных вместе. Карты признаков содержат свертки 3×3 , 32, 3×3 , 64, 3×3 , 128, 3×3 , 128, максимальный уровень пула размером 2×2 и функцией активации ReLu между ними. Выходные данные операций свертки и максимального объединения объединяются в 2D-плоскости, называемые картами объектов, и получили размеры карт объектов $198 \times 198 \times 32$, $97 \times 97 \times 62$, $46 \times 64 \times 128$ и $21 \times 21 \times 128$ соответственно, для операций свертки и размеров карт объектов с размерами $99 \times 99 \times 32$, $48 \times 48 \times 64$, $23 \times 23 \times 128$ и $10 \times 10 \times 128$ из операций объединения, соответственно, с вводом изображения размера $200 \times 200 \times 3$. Стоит отметить, что каждая плоскость слоя в сети была получена путем объединения одной или нескольких плоскостей предыдущих слоев. Конечными полученными результатами являются потери при обучении = 0,1288, точность обучения = 0,9531, потеря при проверке: 0,1835 и точность при проверке 0,9373.

В работе [15], авторы оценивали функциональность предварительно обученных моделей CNN, используемых в качестве выделяемых признаков, за которыми следуют различные классификаторы для классификации аномальных и нормальных рентгеновских снимков грудной клетки. Авторы аналитически

определяют оптимальную модель CNN для этой цели. Полученные статистические результаты показывают, что предварительно обученные модели CNN, используемые вместе с алгоритмами контролируемого классификатора, могут быть очень полезны при анализе рентгеновских изображений грудной клетки, особенно для выявления пневмонии. Наилучшие результаты, достигнуты с архитектурой DenseNet169 в качестве выделения функций, можно объяснить благодаря ее возможности доступа к картам функций со всех предыдущих уровней. и SVM в качестве процесса улучшения характеристик модели. Литературные исследования показывают вклад методов переносного обучения, в том числе извлечения признаков, в задачи визуального распознавания. По этой причине авторы извлекли функции из различных вариантов предварительно обученных моделей CNN, таких как VGGNets, Xception, ResNet-50 и DenseNets [9]. Исследования из литературы также показывают использование классификаторов в сочетании с извлечением признаков на основе CNN, главным образом, в анализе медицинских изображений для улучшения производительности моделей. Следуя упомянутым ранее подходам, были оценены каждая из предварительно обученных моделей с различными классификаторами, чтобы определить идеальную модель для этой цели. В связи с этим авторы предложили модельную архитектуру для обнаружения пневмонии на рентгеновских изображениях фронтальной грудной клетки с использованием Densenet в качестве выделения признаков.

1.3 Основы теории сверточных нейронных сетей

Сверточные нейросети (Convolution Neural Networks, CNN) являются одной из разновидностей нейронных сетей и в общем виде представлены на рисунке 2. Слои CNN состоят из нейронов, которые располагаются в трех измерениях: ширине, высоте и глубине. Основная идея свёрточных нейронных сетей заключается в чередовании свёрточных слоёв и субдискретизирующих слоёв [1].

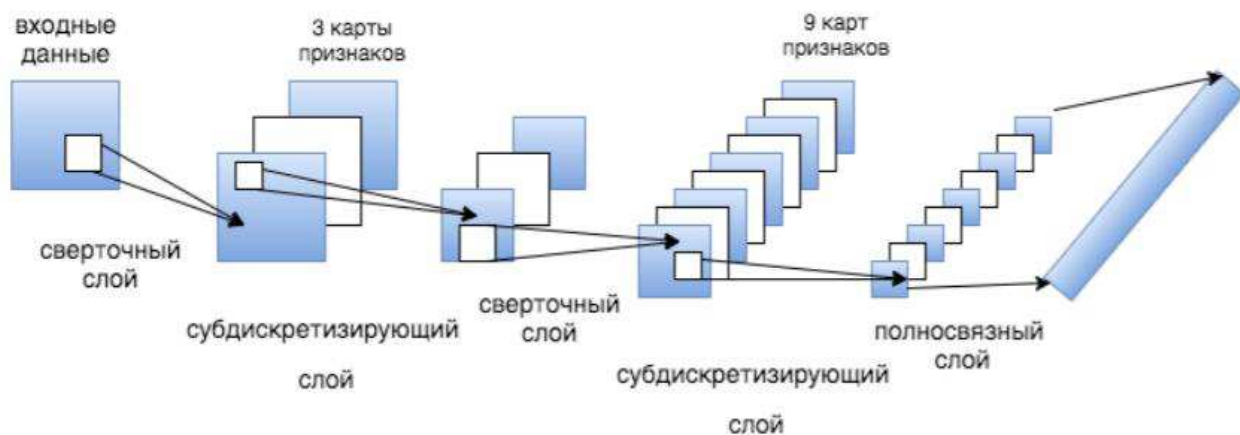


Рисунок 2 – Сверточная нейросеть

Сверточный слой накладывает ядро свертки, поэлементно умножая значения фильтра и изображения, и записывает сумму произведений элементов исходного изображения и ядра. Ядро представляет собой фильтр или окно, которое перемещается по всей области предыдущей карты, находя определенные признаки объектов. Результаты перемножений ячеек суммируются в одном выходном пикселе. В случае сверточных нейросетей ядро определяется в ходе обучения сети. Начальные веса, аналогично случаю перцептрона, могут иметь случайные значения, и корректируются в процессе обучения. Как правило, применяется сразу несколько ядер свертки. Результатом работы данного слоя становятся карты признаков.

Слой активации.

На функцию активации, которая представляет собой нелинейную функцию, попадает скалярный результат каждой свёртки. Слой активации часто логически объединяют со слоем свёртки, считают, что функция активации встроена в слой свёртки. Чаще всего используется функция активации типа гиперболического тангенса или сигмоиды. Функция сигмоиды имеет следующий вид:

$$A = \frac{1}{1+e^{-x}} \quad (1)$$

Сигмоида хорошо подходит для задач классификации так, как в диапазоне значений $[-2;2]$ значения Y меняются очень быстро. Это означает, что Y имеет тенденцию прижиматься к одному из краев, что позволяет при предсказании находить четкие границы. Функция гиперболического тангенса создана на основе сигмоидной функции. Однако градиент такой функции больше.

Также в 2000-х годах была предложена и исследована новая функция активации — ReLu (сокращение от англ. rectified linear unit), имеющая следующий вид:

$$A = \max(0, x) \quad (2)$$

Данная функция позволила существенно ускорить процесс обучения и одновременно упростить вычисления, за счет простоты самой функции. Функция является хорошим аппроксиматором, так как любая функция может быть аппроксимирована комбинацией ReLu. Также ReLu позволяет оставлять некоторые нейроны не активированными, что позволит сделать процесс активации более эффективным.

Слой субдискретизации.

Слой пулинга (иначе субдискретизации) представляет собой нелинейное уплотнение карты признаков, при этом группа пикселей чаще всего размера 2×2 , сжимается до одного пикселя, проходя нелинейное преобразование. Наиболее часто используемая при этом функция максимума, изображённая на рисунке 3. Преобразования происходят в непересекающихся квадратах, каждый из которых сокращается до одного пикселя, при этом выбирается пиксель, имеющий максимальное значение. Операция пулинга позволяет существенно уменьшить пространственный объём изображения. Операция пулинга интерпретируется так: если на предыдущей операции свёртки уже были выявлены некоторые признаки, то для дальнейшей обработки настолько подробное изображение уже не нужно, и оно уплотняется до менее подробного. Данный вид фильтрации позволяет модели не переобучаться. Слой пулинга обычно вставляется между слоями свертки.

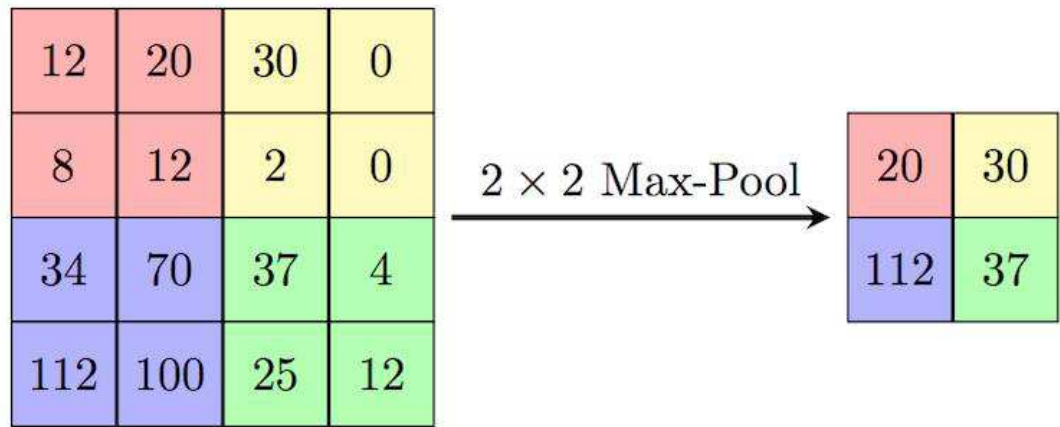


Рисунок 3 – Пример работы пулинга по максимальному значению

Полносвязный слой.

Полносвязный слой берёт входные данные и выводит N -пространственный вектор, где N — число классов, из которых нейронная сеть классифицирует объекты. Работа полносвязного слоя осуществляется через обращение к выходу предыдущего слоя и определение свойств, связанных с определенным классом [6].

После нескольких проходов свёртки изображения и уплотнения с помощью пулинга система перестраивается от конкретной сетки пикселей с высоким разрешением к более абстрактным картам признаков, как правило на каждом следующем слое увеличивается число каналов и уменьшается размерность изображения в каждом канале. В конце концов остаётся большой набор каналов, хранящих небольшое число данных (даже один параметр), которые объясняются, как самые абстрактные понятия, выявленные из исходного изображения.

Выводы по главе 1.

Таким образом, рассмотрены основные виды пневмонии на рентгеновском снимке. Проблема заключается в том, что при большом количестве пациентов или нехватке врачей рентгенологов может возникать снижение качества обработки информации, содержащейся в рентгенограмме. Например, в

густонаселенных районах, а также в районах, где отсутствуют достаточно специалистов в области медицины, врач-рентгенолог вынужден изучать большое количество снимков ежедневно. Использование методов компьютерного зрения для автоматизированного распознавания заболеваний по рентгеновским снимкам позволит решить указанную выше проблему.

Рассмотрены основные аспекты работы сверточных нейронных сетей.

Глава 2 Разработка нейронной сети

2.1 Препроцессинг изображений

Препроцессинг изображений – это предварительная обработка изображений. Для задачи распознавания изображений процесс формирования и настройки обучающей выборки может включать в себя такие этапы, как: способ создания выборки, сжатие изображений до требуемого размера, нормализацию, применение фильтров и т.д. Выполнение всех этапов является необязательным, напротив некоторые из них могут вообще не использоваться, а другие повторяться многократно.

2.1.1 Масштабирование

Масштабирование – это изменение вертикального и горизонтального размера изображения. Существует два типа масштабирования:

1. Пропорциональное – в этом случае соотношение между высотой и шириной рисунка не изменяется, а меняется общий размер.
2. Непропорциональное – в этом случае оба измерения изменяются по-разному.

В нашем эксперименте масштабирование заключалось в изменении вертикального и горизонтального размера изображения до 299x299 пикселей.

Это было сделано с помощью программы XnView. Таким образом, все снимки были приведены к единому размеру.

2.1.2 Устранение шума на изображениях

Подавление шума на рентгенограммах улучшает визуальное восприятие и увеличивает возможность различения низко-контрастных деталей, что делает анализ снимков более удобным для рентгенолога и повышает качество обработки снимка.

Большинство эффективных методов подавления шума использует информацию об уровне шума на изображении. Обычно под шумом понимают случайное отклонение измеренного значения от ее точного значения некоторой физической величины. В рентгенографии детектор фиксирует интенсивность рентгеновского излучения, при этом каждая ячейка работает, как сумматор фотонов, накапливая за время экспонирования в среднем N электронов за счет поглощения фотонов.

Поэтому до применения какого-либо метода фильтрации необходимо определить уровень дисперсии шума. Уровень шума на снимках проекционной цифровой рентгенографии зависит от интенсивности сигнала.

Суммарный эффект данных источников шума моделируется по закону Гаусса случайной величины [11]. Принятой моделью, согласно которой при линейных электронных схемах дисперсия шума пикселя выходного изображения линейно растет со значением накопленного в данном пикселе сигнала:

$$\sigma^2 I(p) = aI(p) + b \quad (3)$$

Шум исходного цифрового изображения моделируется как аддитивная по отношению к сигналу случайная величина, дисперсия которой зависит от сигнала согласно закону (3).

Существует два метода удаления шума:

1. методы фильтрации в пространстве преобразования
2. методы фильтрации в пространстве исходного изображения

В нашей задаче мы используем вторую группу методов, а именно билатеральный фильтр. Чаще всего, эти методы основываются на поиске подобия между пикселями или группами пикселей в исходном изображении и применению той или иной модели усреднения яркости этих пикселей.

Билатеральный фильтр, устраняя шумы, способствует сглаживанию изображения. Но в отличие, например, от медианного фильтра билатеральный фильтр, сохраняет объекты достаточно четкими. Недостатком данного типа фильтра является то, что для фильтрации входных изображений требуется больше времени, чем у других фильтров. Билатеральный фильтр хорошо устраняет импульсный шум, который связан с потерями при передаче изображений по каналам связи.

Основная идея билатерального фильтра заключается в том, чтобы рассмотреть близость между пикселями различных цветов и выполнить переход цветов более последовательно.

Выходное значение яркости отфильтрованного изображения в точке (i,j) можно вычислить по следующей формуле:

$$Y(i, j) = \sum_{l=-N}^N \sum_{k=-m}^M H(i, j; l, k) X(l, k)$$

где $H(i, j; l, k)$ – ядро билатерального фильтра; $X(i, j)$ – значение интенсивности в точке (i, j) исходного изображения.

В программе билатеральная фильтрация реализована с помощью библиотеки OpenCV. Метод `bilateralFilter()` имеет следующие параметры:

1. `dst` – объект `Mat`, представляющий собой выходное изображение.
2. `d` – переменная типа `integer`, представляющая диаметр окрестности пикселя и имеющая значение 15.

3. `sigmaColor` – переменная типа `integer`, представляющая сигма фильтр в цветовом пространстве, имеет значение 80.
4. `sigmaSpace` – переменная типа `integer`, представляющая сигма фильтр в координатном пространстве, имеет значение 80.
5. `borderType` – Целочисленный объект, представляющий тип используемой границы, имеет значение по умолчанию.

2.1.3 Аугментация данных

Когда в наборе данных мало обучающих примеров, как правило происходит переобучение. Один из способов устранения нехватки данных – это расширение, которое заключается в увеличении экземпляров до нужного количества и нужной изменчивости. Расширение представляет собой генерацию данных из существующих экземпляров с помощью использования различных трансформаций к начальному набору обучающей выборки. Данный метод позволяет увеличить количество уникальных входных экземпляров, которые модель больше не увидит и тем самым позволит продемонстрировать большую точность на проверочном наборе данных.

С помощью `tf.keras` мы можем реализовать данные преобразования и генерацию новых изображений через класс `ImageDataGenerator`. Нам будет необходимо передать различные трансформации в виде параметров, которые мы хотели бы применить к изображениям.

Начать можно с простого преобразования, путем горизонтального переворачивания изображения. Чтобы достигнуть подобной трансформации необходимо передать параметрам: `rotation_range`, `width_shift_range`, `height_shift_range`, `zoom_range` значения сдвига конструктору класса `ImageDataGenerator`. Преобразование в виде различных поворотов произвольным образом повернёт изображение на определённое количество градусов. В нашем примере значение угла поворота равно 10 градусов.

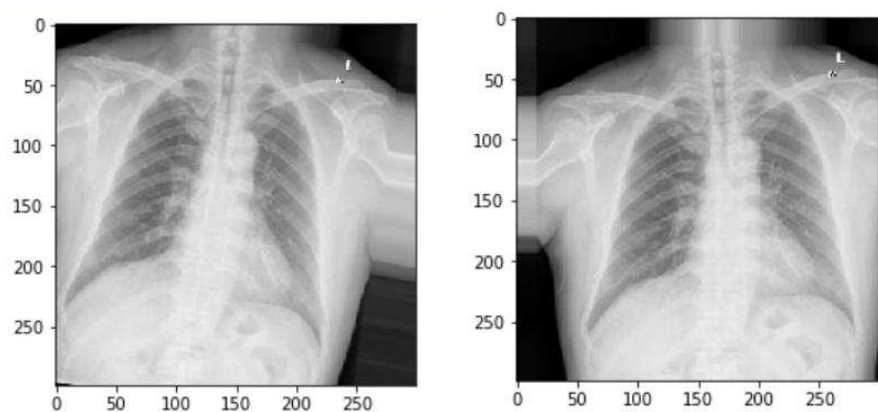


Рисунок 4 – Пример полученных изображений

2.2 Архитектура нейронной сети

В главе рассмотрено сравнение использования нейронной сети Inception V3, VGG 16, DenseNet, а также модифицированной (составной) сети, состоящей из слоев сети Inception v3 и добавочных слоев, операций и методов. Составная нейронная сеть включает в себя слои заранее обученной нейронной сети Inception V3, операции Global Average Pooling, дополнительный полносвязный слой, метод пакетной нормализации, метод Dropout. В качестве функции активации использовалась ReLu после пакетной нормализации, после последнего полносвязного слоя использовалась сигмоидная функция активации.

2.2.1 Готовые нейронные сети

2.2.1.1 VGG 16

VGG16 – это сверточная модель нейронной сети, предложенная К. Симоньяном и А. Циссерманом из Оксфордского университета в статье «Очень глубокие сверточные сети для распознавания крупномасштабных изображений». Модель достигает 92,7% точности из 5 тестов в ImageNet, который представляет собой набор данных из более чем 14 миллионов изображений, принадлежащих 1000 классов. Это была одна из знаменитых моделей, представленных на ILSVRC-2014.

Сети VGG на рисунке 5 используют несколько сверточных слоев 3×3 для представления сложных объектов. Обратите внимание, что блоки 3, 4, 5 фильтров VGG-E: 256×256 и 512×512 3×3 используются несколько раз подряд для извлечения более сложных функций и комбинации таких функций. Это, очевидно, сводится к огромному количеству параметров, а также мощности обучения. Но обучение этих сетей довольно сложное. Все это из-за отсутствия способов регуляризации модели или каким-то образом ограничить массивное пространство поиска, продвигаемое большим количеством параметров.

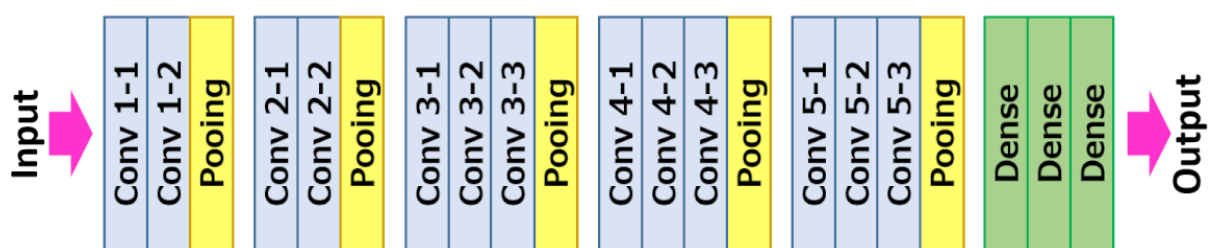


Рисунок 5 - Архитектура сети VGG16

Однако данная сеть имеет два основных недостатка:

1. Низкая скорость обучения.
2. Процесс развертывания требует значительные вычислительные мощности.

Так как обучение сети требует много времени, а также точность распознавания составила 67,2%, сделан вывод, что данная сеть не подходит для распознавания пневмонии на рентгенограмме.

2.2.1.2 DenseNets

Сверточная сеть (DenseNet) соединяет каждый слой с каждым другим слоем в режиме прямой связи. В то время как традиционные сверточные сети с L уровнями имеют L соединений - по одному между каждым уровнем и его последующим уровнем - данная сеть имеет $L(L + 1) / 2$ прямых соединений. Для каждого слоя карты объектов всех предыдущих слоев используются в качестве

входных данных, а его собственные карты объектов используются в качестве входных данных для всех последующих слоев. DenseNets имеет несколько неоспоримых преимуществ: они облегчают проблему исчезающего градиента, улучшают распространение признаков, стимулируют повторное использование признаков и существенно сокращают количество параметров.

Каждый слой добавляет к этой карте свои собственные карты объектов. Скорость роста определяет, сколько новой информации каждый слой вносит в глобальное состояние. После написания глобального состояния к нему можно обращаться из любой точки сети, и, в отличие от традиционных сетевых архитектур, нет необходимости копировать его из уровня в уровень.

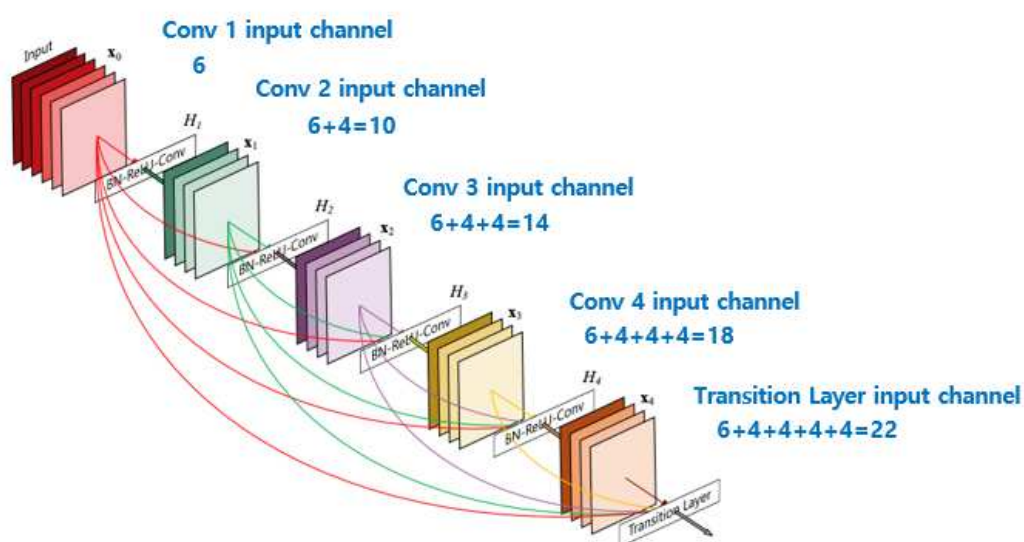


Рисунок 6 - Архитектура сети DenseNet

Архитектура DenseNet разделена на несколько плотно связанных блоков, между которыми находятся переходные слои. Они состоят из слоя нормализации активаций нейронов, сверточного слоя 1×1 , за которым следует средний пул размера 2×2 .

Однако, обучение данной сети происходило довольно много времени, так как для обучения необходимо было 30 и более эпох, а результат (таблица 1) сеть показывала достаточно низкий.

2.2.1.3 Inception V3

Сверточная архитектура нейронной сети Inception V3 на рисунке 7 – это 3-я версия в серии сверточных архитектур глубокого обучения. Inception V3 была обучена с использованием набора данных, состоящего из 1000 классов.

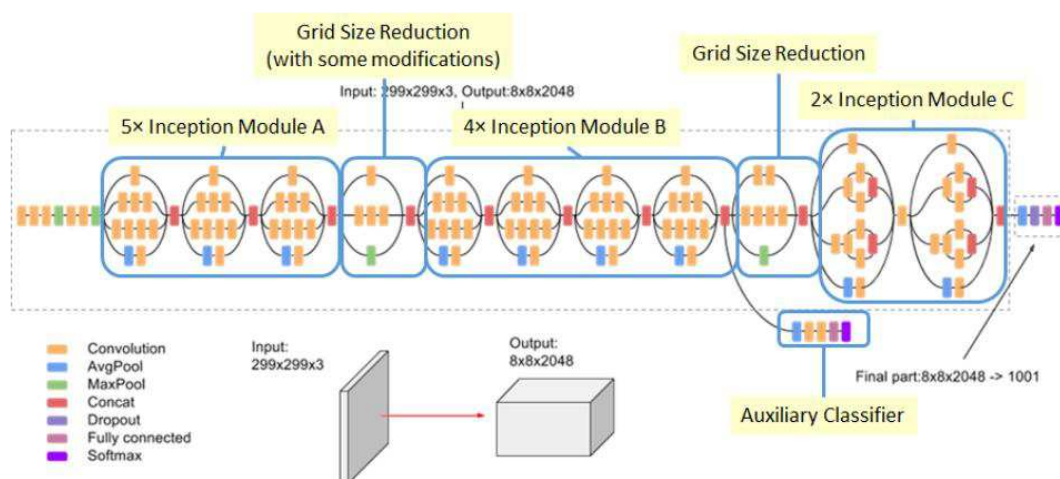


Рисунок 7 – Архитектура сверточной нейронной сети

Сеть Inception v3 объединяет 11 начальных модулей. В данной версии содержатся факторизованные свертки 7x7 и сглаживание меток (тип регуляризующего компонента, добавляемого к формуле потерь, который не позволяет сети быть слишком уверенной в классе) [5]. Также возникли свертки размером 1x1, которые позволили уменьшить количество признаков. В Inception-v3 последовательно соединены 3 модуля Inception A, 5 модулей Inception B и 2 модуля Inception C, модули показаны на рисунке 8. Каждый начальный модуль состоит из нескольких сверточных слоев и объединенных слоев. Пул обозначает объединяющий слой, а $n \times m$ обозначает сверточный слой, где n и m определяют размеры сверток.

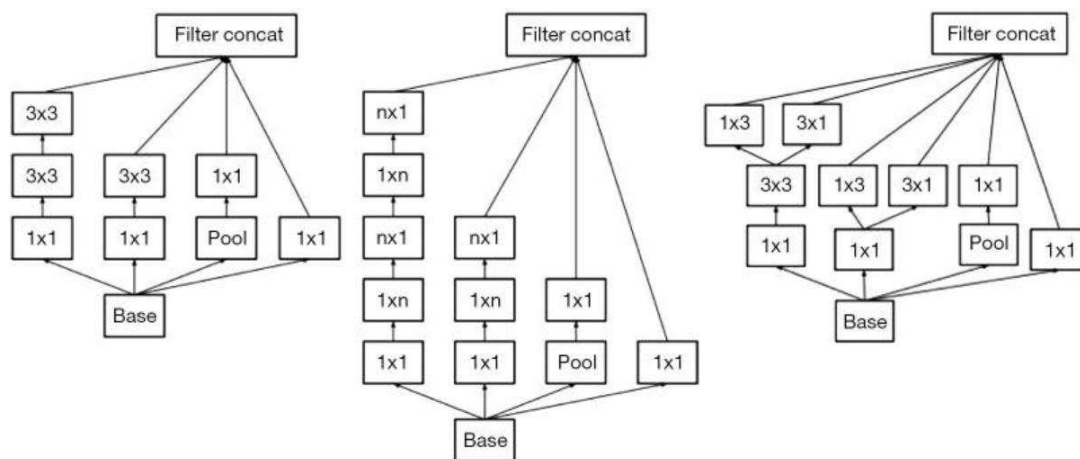


Рисунок 8 - Начальные модули А, В и С (слева направо)

После рассмотрения трех архитектур нейронных сетей получены следующие результаты, представленные в таблице 1.

Таблица 1 – Результат работы нейросетей

	VGG16	DenseNet	Inception V3
Точность обучения	67.2%	72.3%	75.70%
Количество эпох	20	30	20

По результатам тестирования было принято решение о создании составной нейронной сети на основе нейронной сети Inception V3.

2.2.2 Составная нейронная сеть

В модифицированной составной нейронной сети к слоям сети Inception V3 были добавлены: операция Global Average Pooling, метод пакетной нормализации, метод Dropout и дополнительные полносвязные слои с сигмоидной функцией активации.

Архитектура модифицированной составной нейронной сети имеет вид:

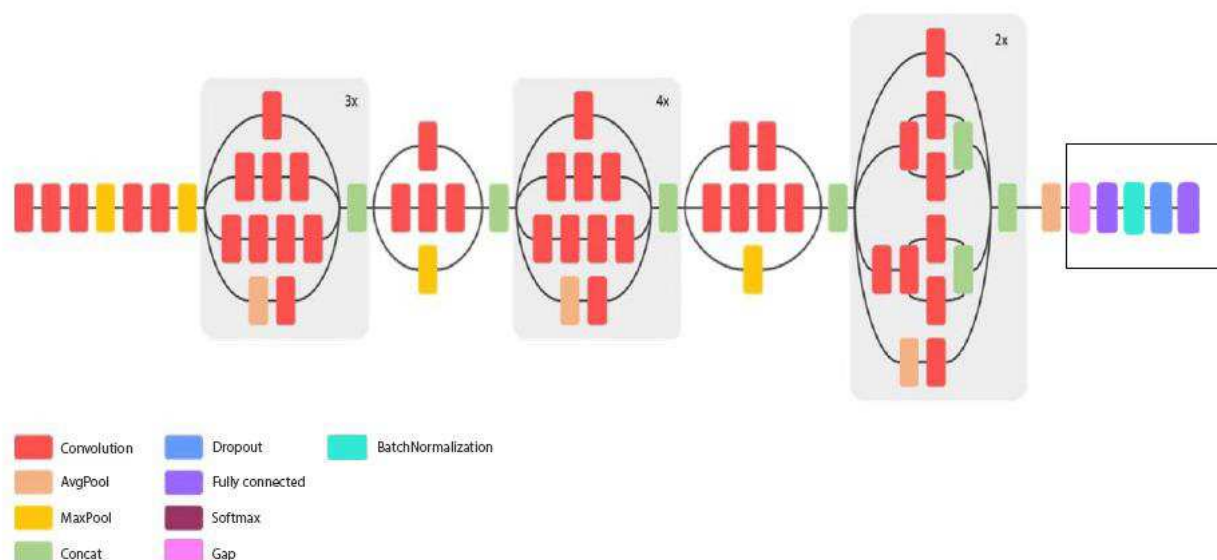


Рисунок 9 – Архитектура модифицированной нейронной сети

Далее подробно будут рассмотрены методы и операции, примененные в данной архитектуре.

2.2.2.1 Global Average Pooling

Глобальный средний пул (GAP) — это операция, которая вычисляет средний вывод каждой карты объектов в предыдущем слое. Эта довольно простая операция, которая значительно сокращает объем данных и подготавливает модель к окончательному классификационному слою. Он также не имеет обучаемых параметров - так же, как Max Pooling (подвыборка с определением максимального значения). На рисунке ниже показано, как используется данная операция в сверточной нейронной сети:

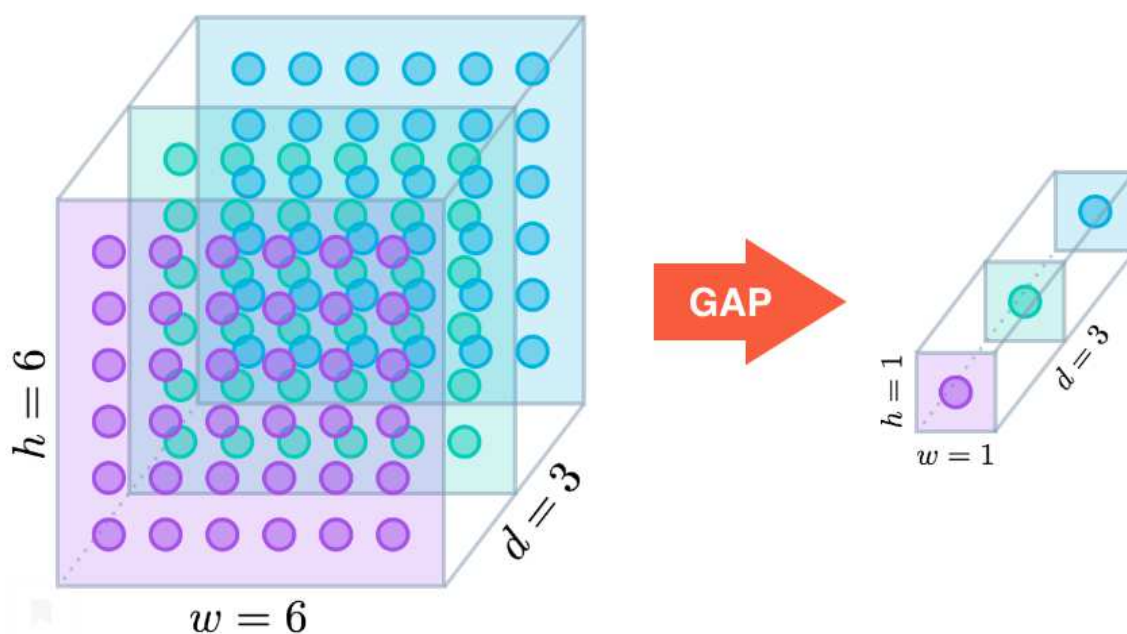


Рисунок 10 – Работа Global Average Pooling

Однако слои GAP выполняют снижение размерности признакового пространства, где тензор с размерами $h \times w \times d$ уменьшается в размере, чтобы иметь размеры $1 \times 1 \times d$. Слои GAP уменьшают каждую карту объектов $h \times w$ до единого числа, рассчитывая среднее значение всех значений $h \times w$. [10]

Глобальный средний пул обладает следующими преимуществами по сравнению с полносвязными слоями:

Устранение всех этих обучаемых параметров уменьшает тенденцию к чрезмерной «подгонке», которая должна управляться в полносвязных слоях с помощью отсева.

Удаление полностью связанных слоев классификации заставляет карты объектов быть более тесно связанными с категориями классификации – так что каждая карта объектов становится своего рода «картой доверия категории».

Благодаря операции усреднения по картам объектов модель более устойчива к пространственным переводам в данных. Другими словами, до тех пор, пока требуемый объект включен / активирован где-то на карте объектов, он все еще будет «подхвачен» операцией усреднения.

2.2.2.2 Дополнительный полносвязный слой

В главе 1 описан принцип работы полносвязного слоя. В программе добавлен дополнительный полносвязный слой, содержащий 512 нейронов с функцией активации ReLu, что позволило избежать исчезновения градиента, через более прямые связи между слоями, как показано на рисунке 11. Плотные связи позволяют сигналам ошибок получить дальнейшее обратное распространение с меньшим градиентом, исчезающим между слоями в глубокой нейронной сети.

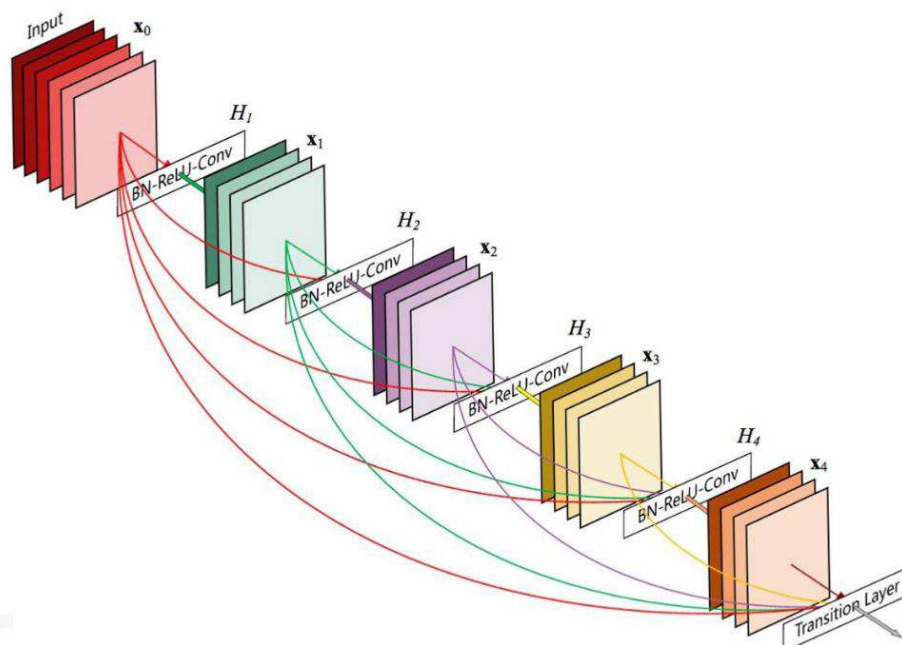


Рисунок 11 - Работа полносвязного слоя

Также задачей данного слоя является классификация для интерпретации финального предсказания результата классификации.

2.2.2.3 Пакетная нормализация

Пакетная нормализация (BatchNormalization) — это метод для повышения скорости, производительности и стабильности работы искусственных нейронных сетей. Он используется для нормализации входного слоя путем повторного центрирования и масштабирования. Каждый слой нейронной сети

имеет входы с соответствующим распределением, на которое в процессе обучения влияет случайность в инициализации параметров и во входных данных. Влияние этих источников случайности на распределение входов во внутренние слои при обучении определяется как внутренний ковариантный сдвиг.

Пакетная нормализация первоначально была предложена для смягчения внутреннего ковариантного сдвига. На этапе обучения сетей, по мере изменения параметров предыдущих слоев, соответственно изменяется распределение входных данных для текущего слоя, так что текущий слой должен постоянно перестраиваться на новые распределения. Эта проблема особенно серьезна для глубоких сетей, поскольку небольшие изменения в более мелких скрытых слоях будут усиливаться по мере их распространения внутри сети, что приведет к значительному сдвигу в более глубоких скрытых слоях. Поэтому предложен метод пакетной нормализации, позволяющий уменьшить эти нежелательные сдвиги для ускорения обучения и получения более надежной модели.

В нейронной сети пакетная нормализация достигается с помощью шага нормализации, который фиксирует средние значения и дисперсии входных данных каждого слоя. В идеале нормализация проводилась бы по всему обучающему набору, но для использования этого шага совместно со стохастическими методами оптимизации нецелесообразно использовать глобальную информацию. Таким образом, нормализация сдерживается к каждой мини-партии в тренировочном процессе.

Ниже представлен алгоритм пакетной нормализации.

1. На вход: значения x из пакета $B = \{x_1, \dots, x_m\}$, настраиваемые параметры - β, γ , константа для вычислительной устойчивости.
2. Выход: $\{y_i = BN_{\beta, \gamma}(x_i)\}$
3. $\mu_B = \frac{1}{m} \sum_{i=1}^m x_i$ – математическое ожидание пакета.
4. $\sigma_B^2 = \frac{1}{m} \sum_{i=1}^m (x_i - \mu_B)^2$ – дисперсия пакета.

$$5. x'_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} - \text{нормализация.}$$

$$6. y_i = x' + \beta \equiv BN_{\beta, \gamma}(x_i) - \text{сжатие и сдвиг.}$$

2.2.2.4 Dropout

Глубокие нейронные сети образуют сложные взаимосвязи между входами и выходами, а также содержат несколько нелинейных скрытых слоев. Однако при ограниченных данных обучения многие из этих сложных взаимосвязей будут результатом выборки шума, так как они будут существовать в тренировочном наборе, но не в реальном тестовых данных, даже если они взяты из того же обучающего набора. Это приводит к переобучению сети.

«Dropout» — это метод, который предотвращает переобучение и обеспечивает способ приближительного комбинирования экспоненциально много различных нейронных сетей [16]. Термин «Dropout(выпадение)» относится к выбывшим единицам в нейронной сети. Отбрасывая единицу, мы имеем в виду ее временное удаление, сеть вместе со всеми входящими и исходящими соединениями, как показано на рисунке ниже.

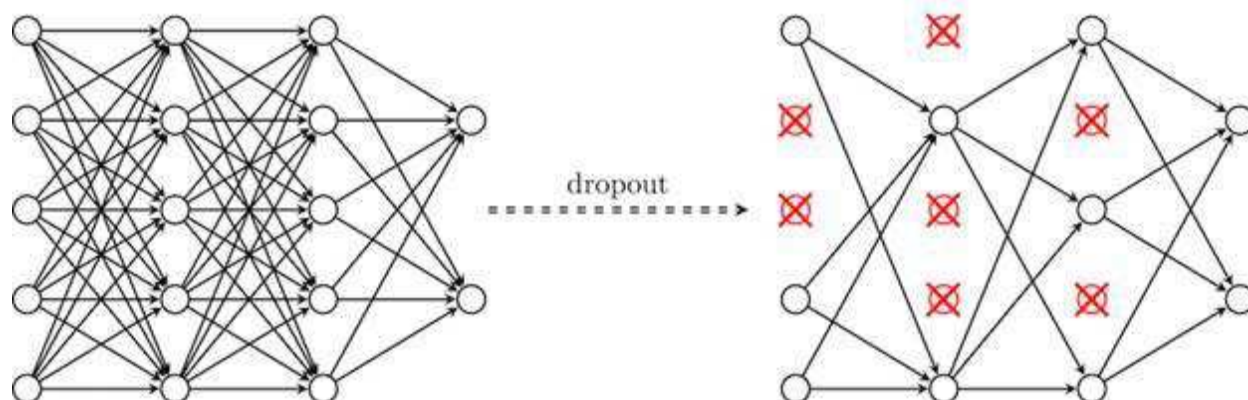


Рисунок 12 - Графическое представление метода Dropout

Случайным образом выбираются единицы для сброса. Каждая единица сохраняется с фиксированной вероятностью p , независимо от других единиц, где p можно выбрать или можно просто установить на 0,5, что близко к оптимальному для широкого диапазона сети и задачи.

Как говорилось ранее, «Dropout» отключает нейроны с вероятностью p и, тем самым, оставляет включенными с вероятностью $q=1-p$. Вероятность отключения каждого нейрона одинакова. Это говорит о том, что при следующих условиях:

1. $h(x) = xW + b$ – линейная проекция входного d_i -мерного вектора x на d_h -мерное пространство выходных значений.
2. $a(h)$ – функция активации.

Применяя «Dropout» к данной проекции на этапе обучения можно увидеть следующую функцию активации:

$$f(h) = D \cdot a(h),$$

Где $D = (X_1, \dots, X_{d_h})$ - d_h - мерный вектор случайных величин X_i , распределенных по закону Бернулли.

X_i имеет следующее распределение вероятностей:

$$f(k, p) = f(x) = \begin{cases} p, & \text{if } k = 1 \\ 1 - p, & \text{if } k = 0 \end{cases},$$

где k – все возможные выходные значения.

Видно, что эта случайная величина соответствует операции «Dropout», примененной к одному нейрону [17]. Действительно, нейрон отключают с вероятностью $p=P(k=1)$, в противном случае — оставляют включенным.

Применение Dropout к i -му нейрону:

$$O_i = X_i a\left(\sum_{k=1}^{d_i} w_k x_k + b\right) = \begin{cases} a\left(\sum_{k=1}^{d_i} w_k x_k + b\right), & \text{if } X_i = 1 \\ 0, & \text{if } X_i = 0 \end{cases},$$

где $P(X_i = 0) = p$.

Так как на этапе обучения нейрон остается в сети с вероятностью q , на этапе тестирования нам необходимо имитировать поведение комплекса нейронных сетей, использованного при обучении. Поэтому на этапе тестирования необходимо умножить функцию активации на коэффициент q . Таким образом, на этапе обучения:

$$O_i = X_i a \left(\sum_{k=1}^{d_i} w_k x_k + b \right)$$

На этапе тестирования:

$$O_i = qa \left(\sum_{k=1}^{d_i} w_k x_k + b \right)$$

Таким образом, прием «dropout» с параметром p за одну итерацию обучения проходит по всем нейронам определенного слоя и с вероятностью p полностью исключает их из сети на время итерации. Это заставит сеть обрабатывать ошибки и не полагаться на существование определенного нейрона (или группы нейронов), а полагаться на “единое мнение” (consensus) нейронов внутри одного слоя. Это довольно простой метод, который эффективно борется с проблемой переобучения, без необходимости вводить другие регуляризаторы.

2.3 Обучение нейронной сети

Процесс обучения включает в себя поиск набора весов в сети, который оказывается хорошим или достаточно хорошим для решения конкретной проблемы. Этот учебный процесс является итеративным, то есть он шаг за шагом продвигается с небольшими обновлениями модели, взвешивающими каждую итерацию, и, в свою очередь, изменением производительности модели на каждой итерации. Итеративный процесс обучения нейронных сетей решает задачу оптимизации, которая находит параметры (веса модели), которые приводят к минимальной ошибке или потере при оценке примеров в наборе обучающих данных. Оптимизация — это процедура направленного поиска, и задача оптимизации, которую мы хотим решить при обучении модели нейронной сети, является очень сложной.

2.3.1 Функция потерь

Функция потерь позволяет определить количественную оценку «пригодности» любого входного веса. В качестве функции потерь будет использоваться бинарная кросс-энтропия.

Бинарная кросс-энтропия – это функции потерь, которая используется при принятии решений «да / нет», например, при классификации по нескольким меткам. Потеря говорит вам, насколько неверны предсказания вашей модели. Например, в задачах с несколькими метками, где пример может принадлежать нескольким классам одновременно, модель пытается решить для каждого класса, принадлежит ли пример этому классу или нет.

Двоичная кросс-энтропия измеряет, насколько далеко от истинного значения (которое равно 0 или 1) прогнозируется для каждого из классов, а затем усредняет эти ошибки по классам, чтобы получить окончательную потерю. Формула бинарной кросс-энтропии имеет вид:

$$L(y, y') = -\frac{1}{N} \sum_{i=0}^N (y * \log(y'_i) + (1 - y) * \log(1 - y'_i))$$

Формула говорит о том, что для истинного значения точки ($y=1$) она добавляет $\log(p(y))$ к потере, то есть логарифмическую вероятность того, что она истинная. И наоборот, он добавляет $\log(1-p(y))$, то есть логарифмическую вероятность того, что она будет не истинной, для каждой точки ($y=0$).

Почему необходимо брать отрицательный логарифм? Поскольку мы пытаемся вычислить потерю, нам нужно «штрафовать» за неправильные прогнозы, верно? Если вероятность, связанная с истинным классом, равна 1,0, нам нужно, чтобы его потеря была равна нулю. И наоборот, если эта вероятность мала, скажем, 0,01, нам нужно, чтобы ее потеря была огромной. Получается, принимая (отрицательный) логарифм вероятности (поскольку логарифм значений от 0,0 до 1,0 является отрицательным, мы берем отрицательный логарифм, чтобы получить положительное значение для потери). После этого вычисляется среднее значение из всех потерь. А далее с помощью оптимизаторов

осуществляется поиск правильных весов, при которых функция достигает минимума.

2.3.2 Оптимизатор

Оптимизаторы служат для повышения производительности нейронной сети. В данной программе использовался оптимизатор RMSprop (Root Mean Square Propagation) или среднеквадратичное распределение [19].

Оптимизатор RMSprop аналогичен алгоритму градиентного спуска с импульсом. Стохастический градиентный спуск с импульсом () запоминает обновление Δw на каждой итерации и определяет следующее обновление как линейную комбинацию градиента и предыдущего обновления:

$$\Delta w = \alpha \Delta w - \eta \nabla Q_i(w)$$

$$\Delta w = w + \Delta w$$

$$w = w - \eta \nabla Q_i(w) + \alpha \Delta w$$

где параметр w , который сводит к минимуму $Q(w)$ должен быть оценен, η размер шага (иногда называемый скоростью обучения) и α – коэффициент экспоненциального затухания между 0 и 1, который определяет относительный вклад текущего градиента и более ранних градиентов в изменение веса. Название импульса происходит от аналогии с импульсом в физике: вектор веса w , рассматриваемый как частица, путешествующая через пространство параметров [15], испытывает ускорение от градиента потерь («сила»). В отличие от классического стохастического градиентного спуска, он имеет тенденцию продолжать движение в одном направлении, предотвращая колебания. Momentum успешно использовался компьютерными учеными в обучении искусственных нейронных сетей в течение нескольких десятилетий.

Однако, оптимизатор RMSprop ограничивает колебания в вертикальном направлении. Таким образом, мы можем увеличить скорость нашего обучения, и наш алгоритм может сделать большие шаги в горизонтальном направлении, сходясь быстрее. Разница между RMSprop и градиентным спуском заключается

в том, как вычисляются градиенты. Следующие уравнения показывают, как вычисляются градиенты для оптимизатора RMSprop. Значение импульса обозначается бета и обычно устанавливается равным 0,9.

Также RMSprop сам регулирует скорость обучения и делает это автоматически. Более того, RMSprop выбирает разную скорость обучения для каждого параметра. В RMSprop каждое обновление выполняется в соответствии с уравнениями, описанными ниже. Это обновление выполняется отдельно для каждого параметра. Формула для расчета обновления:

$$v_t = \rho v_{t-1} + (1 - \rho) * g_t^2$$
$$\Delta w_t = -\frac{\eta}{\sqrt{v_t + \epsilon}} * g_t$$
$$w_{t+1} = w_t + \Delta w_t$$

Где η – шаг обучения, v_t - экспоненциальное среднее квадратов и градиентов, g_t – градиент в момент времени t вдоль w^j .

В первом уравнении вычисляется экспоненциальное среднее квадрата градиента. Поскольку вычисление производится отдельно для каждого параметра, градиент соответствует проекции или компоненте градиента вдоль направления, представленного параметром, который мы обновляем. Для этого мы умножаем экспоненциальное среднее, вычисленное до последнего обновления с гиперпараметром, представленным греческим символом ρ . Затем мы умножаем квадрат текущего градиента на $(1 - \rho)$. Затем мы складываем их вместе, чтобы получить экспоненциальное среднее значение до текущего временного шага. На самом деле, название «экспоненциальный» происходит от того, что вес предыдущих терминов падает экспоненциально (самый последний термин взвешивается как ρ , следующий-как квадрат ρ , затем куб ρ и так далее.)

Затем во втором уравнении мы определили размер нашего шага. Мы движемся в направлении градиента, но на размер нашего шага влияет экспоненциальное среднее. Мы выбрали начальную скорость обучения η , а затем

делим ее на среднюю. В нашем случае, поскольку среднее значение w_1 намного больше, чем w_2 , шаг обучения для w_1 намного меньше, чем для w_2 .

Третье уравнение — это просто шаг обновления. Гиперпараметр ρ обычно выбирается равным 0,9. Эпсилон — это уравнение 2, чтобы гарантировать, что мы не закончим делением на ноль, и обычно выбирается $1e-10$.

Также следует отметить, что RMSProp неявно выполняет уменьшение шага. Предположим, что мы направляемся к минимумам, и мы хотим замедлиться, чтобы не превысить минимумы. RMSProp автоматически уменьшит размер шагов градиента в сторону минимумов, когда шаги слишком велики (большие шаги делают нас склонными к промахам).

2.3.3 Метрики

В задачах машинного обучения для оценки качества моделей и сравнения различных алгоритмов используются метрики. В нашей системе в качестве метрик используются: precision (точность) и recall (полнота) [23].

Таблица 2 – Ошибки классификации

	$y = 1$	$y = 0$
$\hat{y} = 1$	True Positive (TP)	False Positive (FP)
$\hat{y} = 0$	False Negative (FN)	True Negative (TN)

Здесь \hat{y} — это ответ алгоритма на объекте, а y — истинная метка класса на этом объекте. Таким образом, ошибки классификации бывают двух видов: False Negative (FN) и False Positive (FP).

Формулы для расчета вероятности того, что класс будет предсказан правильно:

$$precision = TP / (TP + FP)$$

$$recall = TP / (TP + FN)$$

Precision можно интерпретировать как долю объектов, названных классификатором положительными и при этом действительно являющимися положительными, а recall показывает, какую долю объектов положительного класса из всех объектов положительного класса нашел алгоритм.

Recall демонстрирует способность алгоритма обнаруживать данный класс вообще, а precision — способность отличать этот класс от других классов.

Ошибки классификации бывают двух видов: False Positive и False Negative. В статистике первый вид ошибок называют ошибкой I-го рода, а второй — ошибкой II-го рода. В нашей задаче по определению пневмонии, ошибкой первого рода будет принятие лояльного абонента за уходящего, так как наша нулевая гипотеза состоит в том, что никто из абонентов не уходит, а мы эту гипотезу отвергаем. Соответственно, ошибкой второго рода будет являться «пропуск» уходящего абонента и ошибочное принятие нулевой гипотезы.

2.4 Результаты работы составной нейронной сети

Созданная архитектура для распознавания пневмонии на рентгеновском снимке, основанная на архитектуре Inception V3 сходится при следующих параметрах:

1. Оптимизатор RMSprop со скоростью обучения $lr=1e-4$.
2. Количество эпох 20
3. Использование всех тренировочных изображений

Таблица 3 – Результаты работы сети

Эпоха	Значение функции потерь	Точность распознавания
1	0.8687	50.83%
2	0.6039	51.67%
3	0.3836	58.33%

4	0.3636	58.33%
5	0.3011	55.83%
6	0.2545	56.67%
7	0.2444	58.33%
8	0.1395	65.00%
9	0.1617	70.83%
10	0.1136	70.83%
11	0.0881	65.00%
12	0.0757	60.83%
13	0.0886	64.17%
14	0.0527	75.00%
15	0.0319	76.67%
16	0.0250	75.00%
17	0.0222	85.00%
18	0.0531	85.83%
19	0.0160	88.33%
20	0.0317	88.33%

Исходя из результатов таблицы 3 видно, что при увеличении количества эпох, точность распознавания увеличивается, а значение функции потерь уменьшается. Точность распознавания составляет 88.33%.

Выводы по главе 2

В главе 2 описан препроцессинг изображений, представлено сравнение готовых архитектур таких, как VGG16, DenseNet, Inception V3, а также представлен процесс модификации составной нейронной сети на основе Inception V3, а также ее обучение.

По результатам работы созданная составная нейронная сеть показала хороший результат 88,33% точности распознавания, по сравнению с готовой архитектурой Inception V3, точность которой составила 75,7%.

Глава 3 Экспериментальные исследования

3.1 Набор данных

Использован открытый набор данных Chest X-Ray Images (Pneumonia) на Kaggle. Исходные изображения были разного размера, поэтому снимки были преобразованы к размеру 299x299 и распределены в каталоги по классам, как требует Keras.

В работе всего использовано 5857 снимков флюорографии, снимки разделены на каталоги для обучения, тренировки, проверки и каждый каталог разделен еще на два каталога, содержащих снимки без пневмонии и с пневмонией.

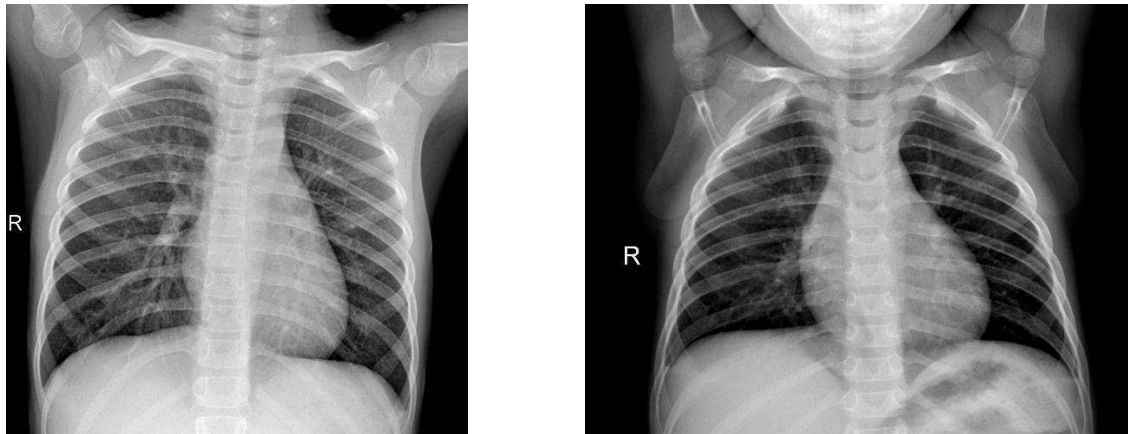


Рисунок 13 – Снимок здоровых легких, не пораженных пневмонией

Из них 1583 снимка, содержащих снимки флюорографии без признаков пневмонии, а 4274 снимка, с диагнозом пневмония. Снимки представлены в оттенках серого цвета.

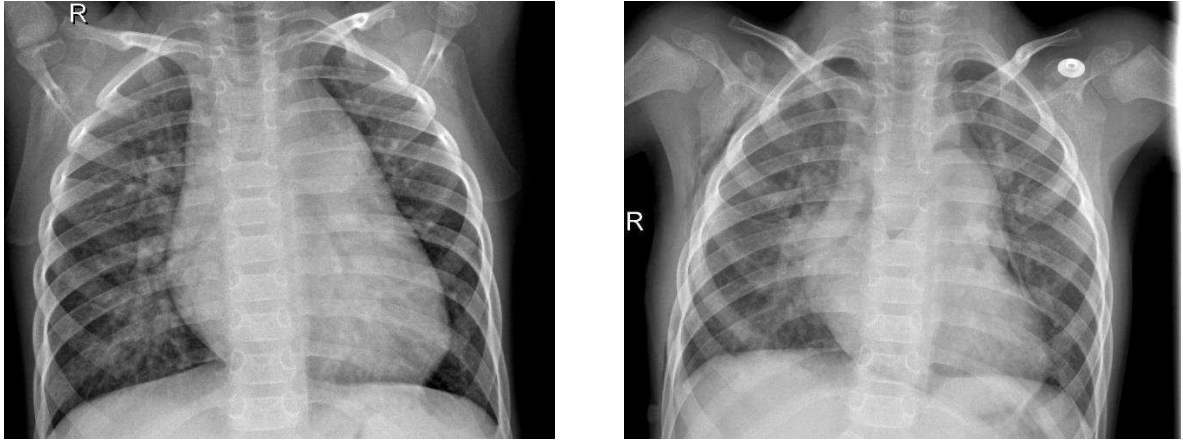


Рисунок 14 – Снимки легких, пораженный пневмонией

3.2 Результаты экспериментальных исследований.

Для выполнения исследований использовались:

1. Язык программирования: Python 3.7.
2. Библиотеки: keras, numpy, matplotlib.
3. Среда разработки: Jupyter Notebook.

Для оценки качества работы сети были взяты 50 снимков рентгенограммы с диагнозом пневмония и столько же снимков без диагноза пневмонии. Нейросеть показала среднюю точность 80,0% при распознавании пневмонии. Нейросеть правильно распознала 40 снимков с диагнозом пневмония. Результаты подтверждают мнение о том, что методы глубокого обучения могут быть использованы для упрощения диагностики и обработки снимков.

На графиках ниже показаны значения функции потерь и точности распознавания на тестовом наборе данных.



Рисунок 15 – Точность на проверочном наборе данных



Рисунок 16 – Функция потерь на проверочном наборе данных

ЗАКЛЮЧЕНИЕ

В ходе выпускной квалификационной работы выполнен анализ современных способов распознавания пневмонии на рентгеновском снимке с помощью нейросетей. Предварительно обработаны данные, а именно выполнено масштабирование экземпляров до единого размера, устранен шум с помощью билатерального фильтра и проведена аугментация данных.

Проведено тестирование готовых архитектур нейронных сетей таких, как DenseNet, VGG16 и Inception V3. Проведена модификация нейронной сети Inception V3, полученная сеть показала точность распознавания 88,3%.

По результатам экспериментальных исследований показано, что методы глубокого обучения могут быть использованы для упрощения диагностики обработки снимков. Полученные результаты также показывают, что методы глубокого обучения могут быть использованы для улучшения диагностики относительно традиционных методов, которые помогут улучшить качество лечения и служить медицинским работникам двусторонней системой подтверждения диагноза.

СПИСОК СОКРАЩЕНИЙ

НС – Нейронная сеть

ВОЗ – Всемирная организация здравоохранения

GAP (Global Average Pooling) – глобальный средний пул

CNN (convolutional neural network) – свёрточная нейронная сеть

RMSprop (Root Mean Square Propagation) – среднеквадратичное отклонение

ReLU (Rectified Linear unit) – линейная функция активации

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. LeCun Y., Kavukcuoglu K., Farabet K. Convolutional networks and applications in vision // In International Symposium on Circuits and Systems, Paris, France, 2010. – pp. 253-256.
2. Rajkomar A., Lingam S., G. Taylor A., Blum M., Mongan J. High-Throughput Classification of Radiographs Using Deep Convolutional Neural Networks. // Journal of Digital Imaging. -№30. - 2016. – pp. 95-101.
3. Naicker S., Plange-Rhule J., Tutt R. C., Eastwood J. The shortage of health workers in developing countries of Africa. // Ethnicity & Disease. - №19. -2009. – pp. 1892-1900.
4. Szegedy C., Liu W., Jia Y., Sermanet P., Reed S., Anguelov D., Dumitru Erhan, Rabinovich A. Going deeper with convolutions //The IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 2015 – pp. 1-9.
5. Szegedy C., Vanhoucke V., Ioffe S., Shlens J., Wojna Z. Rethinking the Inception Architecture for Computer Vision. // The IEEE Conference on Computer Vision and Pattern Recognition, 2016 – pp. 2818 – 2826.
7. Сикорский О.С. Обзор свёрточных нейронных сетей для задачи классификации изображений // Новые информационные технологии в автоматизированных системах, 2017, № 20, 2017. - с. 37-42.
8. Официальный сайт Cloud.Google [Электронный ресурс]: режим доступа: <https://cloud.google.com/tpu/docs/inception-v3-advanced>.
9. Всемирная организация здравоохранения [Электронный ресурс]. Режим доступа: <https://www.who.int/ru/news-room/fact-sheets/detail/the-top-10-causes-of-death> .
10. Cook A. Global Average Pooling Layers for Object Localization [Электронный ресурс]. Режим доступа: <https://alexisbcook.github.io/2017/global-average-pooling-layers-for-object-localization/>
11. Меркурьев С.В. Методы подавления шума цифровых рентгенограмм // НИПК «Электрон» - с. 7

12. Min Lin, Qiang Chen, Shuicheng Yan. Network In Network. [Электронный ресурс]. Режим доступа: <https://arxiv.org/pdf/1312.4400.pdf>
13. Vikash Chouhan , Sanjay Kumar Singh, Aditya Khamparia, Deepak Gupta, Prayag Tiwari, Catarina Moreira, Robertas Damaševičius and Victor Hugo C. de Albuquerque A Novel Transfer Learning Based Approach for Pneumonia Detection in Chest X-ray Images // MDPI, 17 December 2019; Accepted: 9 January 2020; Published: 12 January 2020 – pp.17.
14. Okeke Stephen, Mangal Sain , Uchenna Joseph Maduh, Do-Un Jeong. An Efficient Deep Learning Approach to Pneumonia Classification in Healthcare // 2019 – pp. 7.
15. Dimpy Varshni, Kartik Thakral, Lucky Agarwal, Rahul Nijhawan, Ankush Mittal. Pneumonia Detection Using CNN based Feature Extraction // 17 October 2019.
16. Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. // Journal of Machine Learning Research 15 (2014) – 1929-1958 pp.
17. Paolo Galeone. Analysis of Dropout. [Электронный ресурс]. Режим доступа: <https://pgaleone.eu/deep-learning/regularization/2017/01/10/analysis-of-dropout/#fnref:1>
18. Daniel Godoy. Understanding binary cross-entropy / log loss: a visual explanation. [Электронный ресурс]. Режим доступа: <https://towardsdatascience.com/understanding-binary-cross-entropy-log-loss-a-visual-explanation-a3ac6025181a>
19. V. Bushaev. Understanding RMSprop — faster neural network learning. [Электронный ресурс]. Режим доступа: <https://towardsdatascience.com/understanding-rmsprop-faster-neural-network-learning-62e116fcf29a>
20. Sebastian Ruder. An overview of gradient descent optimization algorithms. // Insight Centre for Data Analytics, NUI Galway Aylien Ltd., Dublin – 2017.

21. СТО 4.2-07-2014: Стандарт организации. – Общие требования к построению, изложению и оформлению документов учебной деятельности. – Система управления СФУ

22. John Duchi, Elad Hazan, and Yoram Singer. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, 2011 - 12:2121–2159.

23. Chris Nicholson. Evaluation Metrics for Machine Learning - Accuracy, Precision, Recall, and F1 Defined. [Электронный ресурс]. Режим доступа: <https://pathmind.com/wiki/accuracy-precision-recall-f1>

24. Kaggle datasets. [Электронный ресурс]. Режим доступа: <https://www.kaggle.com/data>

25. Xiangyu Zhang, Jianhua Zou, Kaiming He, Jian Sun. Accelerating Very Deep Convolutional Networks for Classification and Detection. // *Computer Vision and Pattern Recognition*, 2015.

Приложение А

Отчет «Антиплагиат»



Отчет о проверке на заимствования №1



Автор: Русскина Алена Алексеевна
Проверяющий: Захаров Павел Алексеевич (bik@sfu-kras.ru / ID: 256)
Организация: Сибирский федеральный университет
Отчет предоставлен сервисом «Антиплагиат» - <http://sfukras.antiplagiat.ru>

ИНФОРМАЦИЯ О ДОКУМЕНТЕ

№ документа: 91286
Начало загрузки: 30.06.2020 04:49:55
Длительность загрузки: 00:00:15
Имя исходного файла: Неизвестно
Название документа: Диплом
Размер текста: 1 кб
Тип документа: Автореферат докторской диссертации
Символов в тексте: 57912
Слов в тексте: 6745
Число предложений: 434

ИНФОРМАЦИЯ ОБ ОТЧЕТЕ

Последний готовый отчет (ред.)
Начало проверки: 30.06.2020 04:50:11
Длительность проверки: 00:03:28
Комментарии: не указано
Модули поиска: Модуль поиска ИПС "Адилет", Модуль выделения библиографических записей, Сводная коллекция ЭБС, Модуль поиска "Интернет Плюс", Коллекция РГБ, Цитирование, Модуль поиска переводных заимствований, Модуль поиска переводных заимствований по eLibrary (EnRu), Модуль поиска переводных заимствований по интернет (EnRu), Коллекция eLIBRARY.RU, Коллекция ГАРАНТ, Коллекция Медицина, Диссертации и авторефераты НББ, Модуль поиска перефразирований eLIBRARY.RU, Модуль поиска перефразирований Интернет, Коллекция Патенты, Модуль поиска "СФУ", Модуль поиска общеупотребительных выражений, Кольцо вузов



ЗАИМСТВОВАНИЯ
15,53%


САМОЦИТИРОВАНИЯ
0%

ЦИТИРОВАНИЯ
9,41%

ОРИГИНАЛЬНОСТЬ
75,06%

Приложение Б

Плакаты презентации




СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
SIBIRIAN FEDERAL UNIVERSITY

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра систем искусственного интеллекта

Разработка информационной системы для анализа рентгеновских снимков

Руководитель: к.т.н. доцент
Пятаева А.В
Выпускник: студент КИ16-12Б
Рускина А.А

Рисунок Б. 1 – Плакат презентации №1



СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
SIBIRIAN FEDERAL UNIVERSITY

Цель и задачи

Целью работы является распознавание пневмонии на рентгеновском снимке с помощью сверточной нейросети.

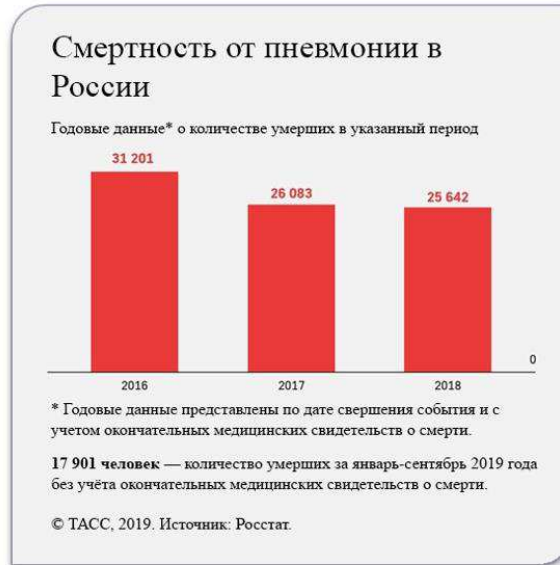
Задачи:

- Выполнить анализ современных способов распознавания пневмонии по рентгеновским снимкам;
- Предварительно обработать данные;
- Выбрать и модифицировать архитектуру;
- Провести экспериментальные исследования;

2/13

Рисунок Б. 2 – Плакат презентации №2

Актуальность



3/13

Рисунок Б. 3 – Плакат презентации №3

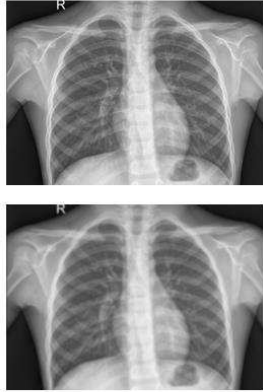


4/13

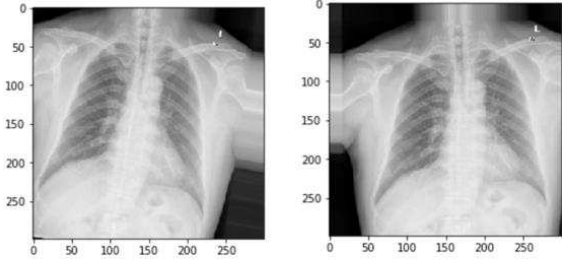
Рисунок Б. 4 – Плакат презентации №4

Предварительная обработка данных

Применение билатерального фильтра



Аугментация данных



5/13

Рисунок Б. 5 – Плакат презентации №5

Архитектуры нейронных сетей

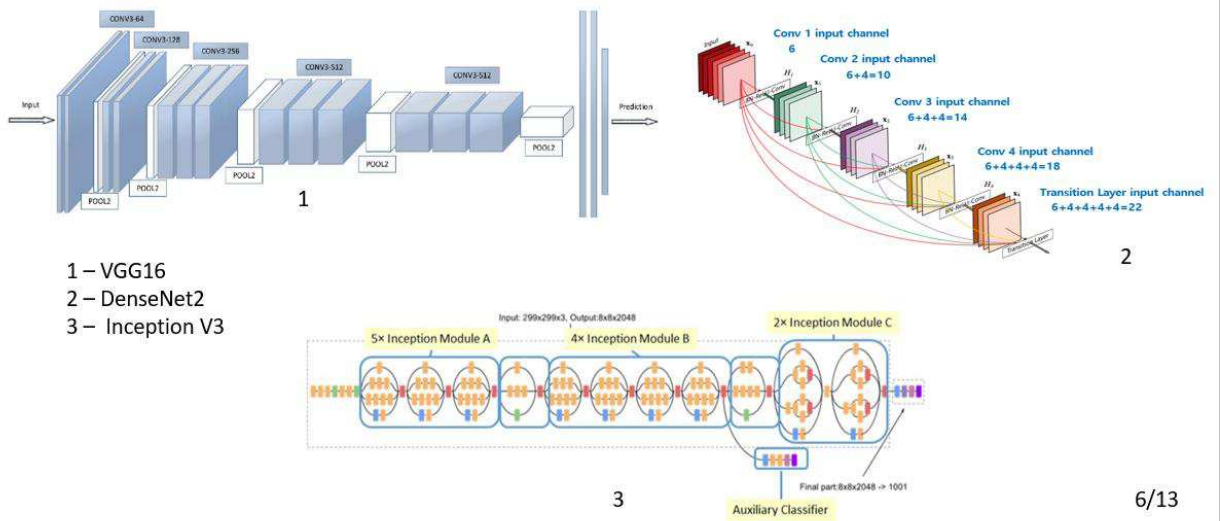


Рисунок Б. 6 – Плакат презентации №6

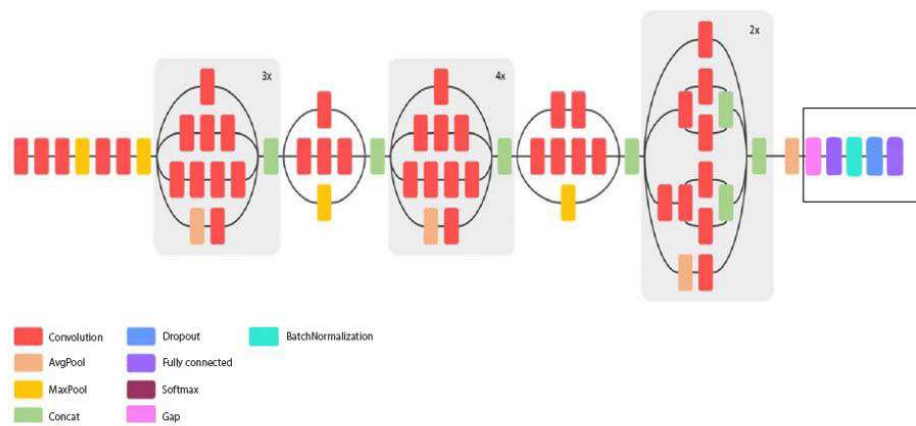
Результат работы готовых архитектур

	VGG16	DenseNet	Inception V3
Точность обучения	67.2%	72.3%	75.70%
Количество эпох	20	30	20

7/13

Рисунок Б. 7 – Плакат презентации №7

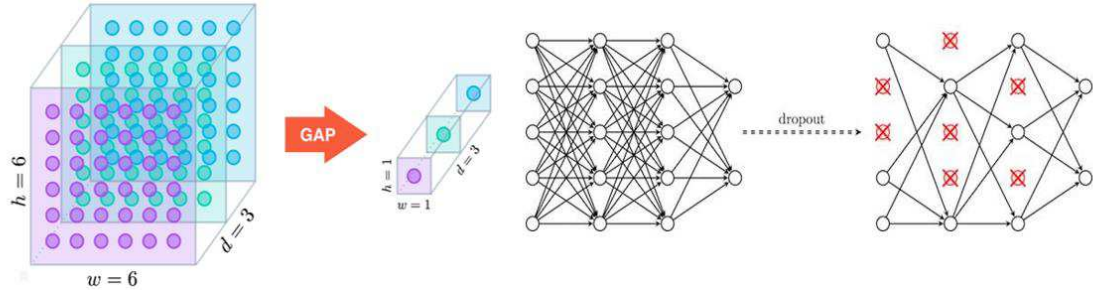
Архитектура модифицированной нейронной сети



8/13

Рисунок Б. 8 – Плакат презентации №8

Модификация архитектуры: составная нейронная сеть



9/13

Рисунок Б. 9 – Плакат презентации №9

Результат работы обученной составной нейронной сети

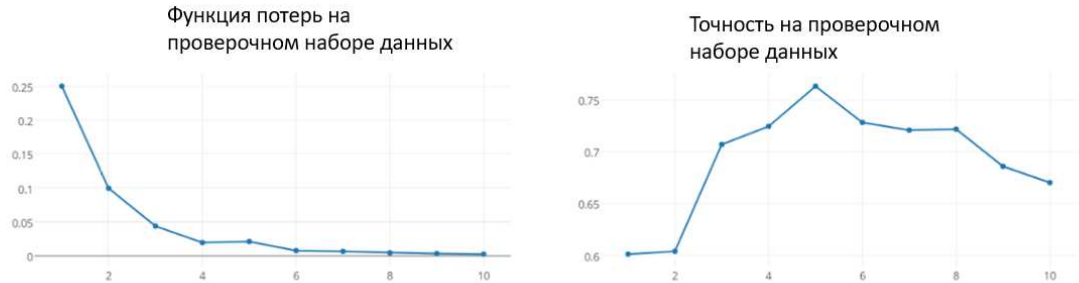
- Оптимизатор RMSprop со скоростью обучения $\text{lr}=1e-4$.
- Количество эпох 20
- Использование всех тренировочных изображений

Эпоха	Значение функции потерь	Точность распознавания
1	0.8687	50.83%
2	0.6039	51.67%
3	0.3836	58.33%
4	0.3636	58.33%
5	0.3011	55.83%
6	0.2545	56.67%
7	0.2444	58.33%
8	0.1395	65.00%
9	0.1617	70.83%
10	0.1136	70.83%
11	0.0881	65.00%
12	0.0757	60.83%
13	0.0886	64.17%
14	0.0527	75.00%
15	0.0319	76.67%
16	0.0250	75.00%
17	0.0222	85.00%
18	0.0531	85.83%
19	0.0160	88.33%
20	0.0317	88.33%

10/13

Рисунок Б. 10 – Плакат презентации №10

Результаты



11/13

Рисунок Б. 11 – Плакат презентации №11

Заключение

В ходе выпускной квалификационной работы выполнен анализ современных способов распознавания пневмонии на рентгеновском снимке с помощью нейросетей. Предварительно обработаны данные, а именно выполнено масштабирование экземпляров до единого размера, устранен шум с помощью билатерального фильтра и проведена аугментация данных.

Проведено тестирование готовых архитектур нейронных сетей таких, как DenseNet, VGG16 и Inception V3. Проведена модификация нейронной сети Inception V3, полученная сеть показала точность распознавания 88, 3%.

- Основные положения и результаты работы обсуждались на конференции:

Конференция «Робототехника и искусственный интеллект» - 14 декабря 2019 (г. Железногорск).

12/13

Рисунок Б. 12 – Плакат презентации №12

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ

Заведующий кафедрой

_____ Г. М. Цибульский

подпись

« _____ » _____ 2020 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.02 — Информационные системы и технологии

Разработка информационной системы для анализа рентгеновских снимков

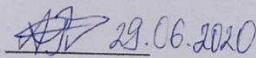
Руководитель


_____ 29.06.20
подпись, дата

доцент, канд. техн. наук

А. В. Пятаева

Выпускник


_____ 29.06.2020
подпись, дата

А. А. Русскина

Красноярск 2020