1. Тема моей курсовой работы "Использование нейронных сетей для классификации изображений"
2. Целью работы было создать нейронную сеть, классифицирующую изображения. В качестве изображений была выбрана рентгенография легких, так как это, как правило, довольно нечеткие изображения, сложные для классификации и от этого процесс реализации такой сети становится интереснее.
3. Набор данных был взят с сайта kaggle, на нем в открытом доступе имеется огромное количество разнообразных наборов данных. Выбранный набор данных содержит почти 6000 изображений. Если взглянуть на число примеров из каждого класса станет очевидно, что оно несбалансированно. Так часто бывает в медицине, да и не только в медицине, это наверно главная проблема при создании нейронных сетей – поиск хорошего набора данных. Но выход есть, его предоставляет библиотека Imgaug, она увеличивает число недостающих образцов, путем добавления некоторых фильтров к имеющимся изображениям, увеличивая тем самым их число. Конечно, это не полноценная замена, но метод тем не менее вполне действенный. По какой-то причине исходные изображения были разного размера, хоть они и были сделаны в одной больнице, и имели разный цветовой режим, поэтому были переведены в формат RGB и масштабированы к размеру 224x224 пикселей при помощи библиотеки OpenСV.
4. Имеется всего два класса: здоровые легкие и с пневмонией. На снимках с пневмонией грудная клетка полностью или почти белая, просвет между ребрами почти отсутствует, в то время как на снимках со здоровыми легкими четко виден просвет, на следующем слайде подробнее рассмотрим.
5. Вот на этих изображениях особенно хорошо видно различие, конечно, не всегда бывает так хорошо видно, но для того и создается нейронная сеть – выявлять, возможно, не самые очевидные закономерности.
6. При применении нейронных сетей к изображениям, как правило, применяется свертка. Предполагается, что пиксели, находящиеся близко друг к другу, теснее “взаимодействуют” при формировании интересующего нас признака, чем пиксели, расположенные в противоположных углах. По изображению проходит некий фильтр, который как-бы обобщает значения пикселей исходного изображения. Далее к такому изображению применяется операции макспул, которая делит его на несколько частей, из которой выбирает максимальное значение, уменьшая тем самым изображения в несколько раз, для ускорения обучения.
7. Вся нейронная сеть, представленная в данной работе, реализована с помощью трех основных инструментов: язык Python, библиотеки TensorFlow и Keras. Язык пайтон выбран по причине простоты изучения и того факта, что именно для него существует множество отличных библиотек, используемых в этой работе.

Если кратко сказать про TensorFlow, то он организует все вычисления и вообще всю логику работы нейросети. TensorFlow создает сессии, вычисления в которых организуются в виде графов. Все эти графы необходимо задавать вручную, это интуитивно не очень понятно, отнимает много времени и высок риск ошибиться, поэтому я использую Керас, он как-бы управляет TensorFlow, увеличивает уровень абстакции, работая поверх него, предоставляя понятные функции и методы, в которых мы определяем параметры сети, и при вызове этих функции запускаются все необходимые вычисления в TensorFlow.

1. Важную роль играет хорошая настройка гиперпараметров, основные из них представлены на слайде. Они управляют корректировкой весов в процессе обучения, могут вызывать раннюю остановку обучения для предотвращения переобучения, это когда модель хорошо классифицирует только примеры из обучающей выборки, управляют размером партии, т.е. числом экземпляров на каждой итерации и самим числом итераций или эпох, как это называют.
2. (тут я поясняю картинку)
3. Это схема получившейся в ходе работы нейронной сети, каждый квадрат обозначает слой, под квадратами размер изображения. Как видно, сначала несколько раза происходит свертка, затем изображения уменьшается вдвое и так несколько раз, пока оно не уменьшится до 14 на 14 пикселей. Далее слой флаттерн переводит полученную матрицу пикселей в вектор, с которым и работают два слоя нейронов по 1024 и 512 нейронов соответственно. Последний слой является выходным и имеет только два нейрона – по нейрону на класс. Стоит сказать, что модель в общем подбирается экспериментальным путем, есть лишь некоторые общие рекомендации, так что нельзя сказать, что какая-то модель является оптимальной, она может быть только лучше уже испытанных.
4. Для оценки эффективности работы сети обычно используют три метрики, которые можно рассчитать построим так называемую матрицу ошибок. Данная матрица построена по результатам работы уже обученной сети по тестовому набору из примерно шестисот изображений. Как можно заметить Модель редко принимает больного за здорового, но часто принимает здорового за больного, что вроде как не очень страшно. Как видно из формул метрика аккераси рассчитывается по всем классам, что не очень объективно, когда они несбалансированны, поэтому обратим внимание на пресижен, именно она говорит о том, что модель предсказывает верный результат в среднем в 76% случаев. А метрика рекол говорит о том, что модель в 98% случаев будет выявлять пневмонию, если она есть, что уже может говорить о том, что использование этой сети на практике может меть смысл.
5. Так-же для визуализации работы нейронной сети была написана небольшая программа, классифицирующая выбранное изображение.
6. В заключение хочется отметить, что в ходе работы была изучена теория нейронных сетей, спроектирована и реализована нейронная сеть, ставящая диагноз по рентгену верно в 76% общих случаев и в 98% случаев, когда имеется заболевание. Все это наводит на мысль, что использование нейронных сетей в медицине вполне оправдано и может аккуратно применяться уже сейчас, главное это иметь качественный набор данных, удачно подобрать модель и настроить гиперпараметры.