**Введение**

В повседневной жизни, мы всё чаще слышим такие выражения: «Машинное обучение» «Искусственный интеллект» и другие. И хотя эти термины всё глубже входят в нашу жизнь, большинство людей лишь смутно представляют, что они значат и что вообще такое «Нейронная сеть»?

**Актуальность:** в последние несколько лет, мы всё чаще сталкиваемся с программами, основанными на принципах машинного обучения и нейронных сетей. Современный человек уже не удивляется, что поисковые системы могут распознать сказанный нами текст, что фотоаппарат способен распознать, что конкретно вы фотографируете. Нейронные сети – одно из самых перспективных направлений программирования, многие компании внедряют их в свой бизнес, аналитики предсказывают этой технологии большое будущее, поэтому можно смело сказать, что машинное обучение не просто новая модная технология, и ей предстоит занять своё место в нашей жизни.

**Цель:** изучить строение и работу нейронных сетей, узнать возможности их применение в повседневной жизни. На основании полученных данных создать простую нейронную сеть для распознавания рукописных цифр.

**Задачи:**

1. Найти основные направления применения нейронных сетей в современном обществе и возможности их использования в будущем.
2. Изучить теоретическую информацию, касающуюся структуры, обучения и использования нейронных сетей.
3. Написать программу на языке программирования Python, основанную на нейронной сети, которая сможет распознавать рукописные цифры.
4. На основе изученных данных попробовать спрогнозировать будущее развитие нейронных сетей.

**Глава 1. Теоретическая часть**

**Глава 1.1. История возникновения**

Датой зарождения искусственного интеллекта можно считать 1943 год, именно тогда американский математик Норберт Винер заложил основы нового научного направления «кибернетики». Основной его идеей стало представление сложных биологических процессов при помощи математических моделей.

Сам термин «искусственный интеллект» впервые был использован американским учёным Джоном Маккарти в 1956 году.

В эти годы начался активный подъём нового направления, многие учёные и инженеры занялись работой в этой сфере.

Новой вехой кибернетики стала представленная Френком Розенблаттом в 1957 году искусственная модель мозга, названная «перцептрон» (допустимо «персептрон»).

Однако несмотря на огромный популярность, энтузиазм, с которым первые исследователи начинали работу, потихоньку спал. И к началу 70-ых годов кибернетику практически забыли. Учёный слишком переоценили возможности, которыми обладает искусственный интеллект. Они считали, что за несколько лет получится разработать искусственную нейронную сеть, сравнимую с человеком, а получилось сделать только простые, узконаправленные и несовершенные программы, на пример искусственный интеллект, который может обыгрывать человека в шашки.

Но в 1974 году советский ученый Галушкин А. И. предложил первый математический алгоритм обучения многослойных нейронных сетей. В 1986 году он был усовершенствован группой американских учёных и назван «back propagation» (алгоритм обратного распространения ошибки). До сих пор он остаётся незаменимым инструментом в разработке нейронных сетей.

Это событие, а также активное развитие вычислительной техники и её повсеместное внедрение возродили интерес к искусственному интеллекту. Однако компьютерам тех лет всё ещё не хватало производительности, а поэтому после нескольких крупных открытий в 1987 году вновь наступает застой.

Но развитие компьютерной техники не собиралось останавливаться. Спустя десятилетие вычислительная мощность новых процессоров стала настолько велика, что смогла поддерживать гораздо более сложные модели нейронных сетей. И в 1997 году произошло ключевое событие в истории развития нейронных сетей. Алгоритм «Deep Blue» от компании «IBM» смог одержать победу в шахматах у действующего чемпиона мира Гарри Каспарова.

Последняя победа компьютера над человеком произошла в марте 2016 года, когда нейронная сеть «AlfaGo» обыграла чемпиона мира по игре «Го». Особенность этой игры в том, что в ней невозможно просчитать все ходы наперёд, как это происходил в шахматах. Считалось, что опытные игроки делают лучшие ходы на основе собственной интуиции, но и тут компьютерная программа смогла обойти нас. Анализ ходов этой нейросети показал, что её ходы ничуть не уступают интуитивным ходам лучшего игрока в мире.

Примерно в 2010 году, компьютеры необходимых мощностей стали появляться в каждом доме, и нейронные сети стали потихоньку входить в нашу повседневную жизнь.

Важную роль в этом сыграло развитие графических процессоров видеокарт, их архитектура параллельных вычислений сделала прорыв в реализации и обучении больших нейронных сетей.

**Глава 1.2. Сферы применения нейронных сетей**

Самыми крупными сферами использования нейронных сетей являются:

1. Распознавание образов и классификация

Нейронные сети очень хорошо могут распознавать и классифицировать объекты на фото или видео. Они распознают рукописный текст, подбирают наиболее подходящий фильтр в зависимости от объекта на фотографии. Уже сейчас во многих городах мира активно используются системы видеослежения, которые могут распознать человека на улице, а дорожные камеры на ходу определяют номер автомобиля.

1. Кластеризация

Разбиение множества входных сигналов на классы, при том, что ни количество, ни признаки классов заранее не известны. Таким способом на крупных сайтах разбивают клиентов на категории и в зависимости от результатов показывают интересующий их товар или рекламу.

1. Прогнозирование

Способности нейронной сети к прогнозированию напрямую следуют из её способности к обобщению и выделению скрытых зависимостей между входными и выходными данными. Таким способом можно с некоторой вероятностью предсказывать вероятные котировки акций на бирже.

Кроме этого нейронные сети активно применяются в задачах анализа данных, оптимизации, принятия решений и управления.

Из необычных сфер использования можно назвать управление транспортными средствами. Уже сейчас на улицах крупных городов можно увидеть машины без водителя. Компьютер такой машины самостоятельно определяет дорожные знаки, препятствия на пути и выбирает кротчайший и безопасный путь.

**Глава 1.3. Что такое нейронная сеть. Персептрон**

Искусственная нейронная сеть (ИНС) — упрощенная модель биологической нейронной сети, представляющая собой совокупность искусственных нейронов, взаимодействующих между собой.

Как было описано выше, первой предложенной архитектурой для нейронной сети стал перцептрон.

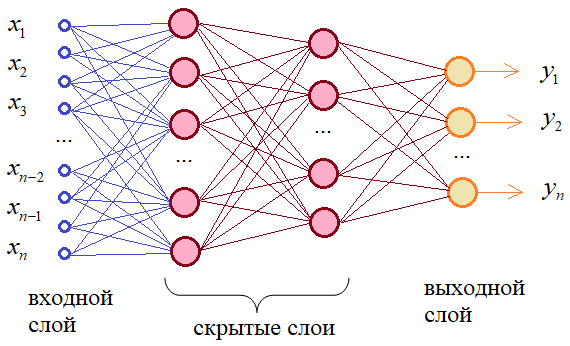


Рисунок 1. Строение персептрона

На рисунке выше видно, что перцептрон имеет несколько основных частей: один входной слой, скрытые слои, их число может быть различным в зависимости от задачи, и один выходной слой. В скрытом и выходных слоях содержатся нейроны. Каждый нейрон связан со всеми нейронами предыдущего и следующего слоёв. Нейрон имеет следующее строение:



Рисунок 2. Строение нейрона

Каждый нейрон суммирует входящие в него сигналы из нейронов предыдущего слоя, после чего пропускает полученную сумму через функцию активации. Результат отправляется в следующий слой нейронной сети.

Главной частью нейронной сети являются связи между нейронами. Каждый из них имеет свой вес, и, соответственно, каждый по имеет разное влияние на нейрон.

Целью программиста, при создании нейронной сети является настройка весов этих связей. Для этого существует алгоритм back propagation или алгоритм обратного распространения ошибки.

**Глава 1.4. Обучение нейронной сети. Алгоритм back propagation**

**Обучение нейронной сети** — поиск такого набора весовых коэффициентов, при котором входной сигнал после прохода по сети преобразуется в нужный нам выходной.

Для обучения программа должна получить тестовое задание и ответ на него. Если этот ответ не совпадает с её результатом, нейронная сеть корректирует свои веса, в соответствии с алгоритмом back propagation или алгоритмом обратного распространения ошибки.

Алгоритм обратного распространения ошибки – алгоритм, для вычисления необходимой коррекции весов связей нейронов. Этот алгоритм, в свою очередь, на методе градиентного спуска.

**Градиентный спуск** — метод нахождения локального минимума или максимума функции с помощью движения вдоль градиента.

**Градиент** — вектор, своим направлением указывающий направление наибольшего возрастания некоторой скалярной величины.

Как модифицировать веса? Мы будем подправлять веса после каждого обучающего примера и, таким образом, «двигаться» в пространстве весов. Чтобы «добраться» до минимума ошибки, нам нужно «двигаться» в сторону, противоположную градиенту. Локальный градиент нейронов последнего слоя считается по формуле:

https://proproprogs.ru/htm/neural_network/files/back-propagation-algoritm-obucheniya-po-metodu-obratnogo-rasprostraneniya.files/image009.png

Здесь:

е – ошибка работы нейросети

f '(vout) – производная функции активации в точке, равной выходу нейросети.

И наконец, нам нужно скорректировать сами веса связей. Для этого умножаем локальный градиент на выходной сигнал нейрона, от которого идёт эта связь и на шаг сходимости.

**Шаг сходимости** – коэффициент, которой уменьшает полученный результат, чтобы не перескочить необходимое значение. Шаг сходимости подбирается самостоятельно, вручную самим разработчиком. Обычно применяются значения: 0,1; 0,01 или 0,001.

Итак, мы с вами скорректировали связи последнего слоя. Переходим к нейрону следующего с конца слоя и для его входящих связей повторим ту же саму процедуру. Но для этого, нужно знать значение его локального градиента. Определяется он просто. Локальный градиент предпоследнего нейрона взвешивается весами входящих в него связей. Полученные значения на каждом нейроне умножаются на производную функции активации, взятую в точках входной суммы.



А дальше действуем по такой же самой схеме, корректируем входные связи по той же формуле.

В результате, мы выполнили одну итерацию алгоритма обучения нейросети. На следующей итерации мы должны взять другой входной вектор из нашего обучающего множества. Повторяя много раз этот процесс, весовые связи будут все точнее описывать обучающую выборку.

**Обучающая выборка –** тестовые задания и ответы на них, которые используются для обучения нейронной сети. Обучающее множество ключевой элемент обучения нейронной сети. Чем больше и разнообразнее будут тестовые задания, тем лучше будет работать нейронная сеть. Зачастую, формирование обучающей выборки крайне сложная задача, так как необходимо набрать не меньше пары тысяч разнообразных тестовых заданий, которые будут различаться между собой. Например, для создания качественной нейросети, которая будет различать мужчин и девушек, необходимо более 1 миллиона фотографий.

Однако не для всех задач нужна такая большая выборка. Для моей цели необходимо несколько тысяч изображений. Я использовал открытую базу данных образцов рукописного написания цифр MNIST от Национального института стандартов и технологий США.



Рисунок 3. Примеры изображений из MNIST

**Глава 1.5. Свёрточные нейронные сети**

Современные нейронные сети способны делать множество интересных вещей - классификация графических образов, стилизация изображений, их раскраску, генерация новых графических образов. Когда речь заходит об обработке изображений, то используется особая архитектура нейросети – сверточные нейронные сети.

Изначально они были предложены Яном Лекуном и созданы для задач классификации графических образов. В 2012-м году команда Алекса Крижевски выиграла в ежегодном соревновании ImageNet по распознаванию графических образов. Их алгоритм показал точность в 83,6% правильной классификации – рекорд того времени. И этот рекорд был достигнут сверточной нейросетью. Для справки, современные нейронные сети показывают точность более 97%, а обычный человек на том же тестовом множестве показывает точность 95%.

Общая идея архитектуры таких сетей была подсмотрена у биологической зрительной системы. Ученые выяснили, что дендриды каждого нейрона соединяются не со всеми рецепторами сетчатки глаза, а лишь с некоторой локальной областью. И уже дендриды всей группы зрительных нейронов покрывают сетчатку глаза целиком:

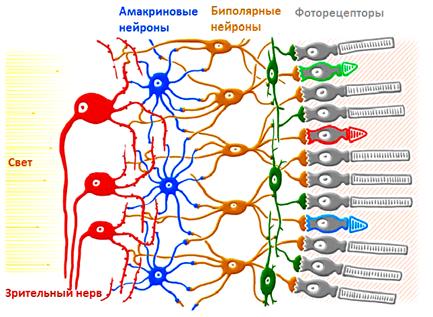
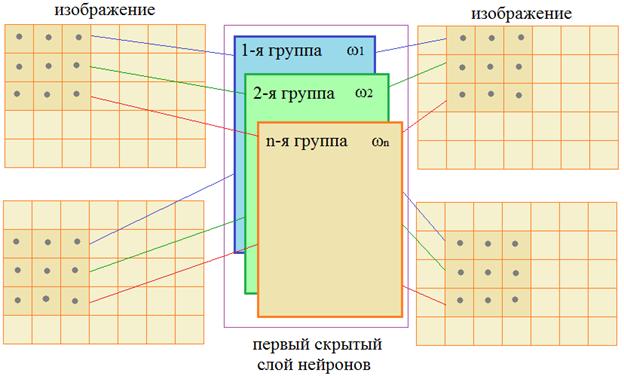


Рисунок 4. Устройство зрительной системы

Математики обобщили эту структуру и предложили следующее решение. Входной сигнал изображения подается на вход нейрона только в пределах ограниченной области, как правило, квадратной, например, 3х3 пикселей. Затем, эта область смещается вправо и входы подаются уже на второй нейрон. Так происходит сканирование всего изображения. Причем, весовые коэффициенты для всех нейронов этой группы – одинаковые.

После этого сканирование изображения повторяется, но с другим набором весовых коэффициентов. Получаем вторую группу нейронов. Затем, третью, четвертую и в общем случае имеем n различных групп. Так формируется первый скрытый слой нейронов сверточной нейросети.

То есть каждый фильтр ищет на изображении свои особые элементы (например, вертикальные или горизонтальные линии, круги) и обобщает эти элементы, передавая результат следующему слою нейронов, который ищет уже более сложные элементы (эллипсы, ломанные линии).



Свёрточные нейронные сети показывают гораздо лучшие результаты при работе с графическими изображениями чем классический перцептрон, так как позволяют определять конкретные элементы, а не просто цвет пикселей.

После изучения необходимой теории, я приступил к написанию самой программы.

**Глава 2. Практическая часть**

**Глава 2.1. Создание графического интерфейса и структуры программы**

**Список литературы**

1. Бринк Х., Ричардс Д., Феверолф М. Машинное обучение. – СПб.: Питер, 2017. –336с.
2. Шарден Б., Массарон Л., Боскетти А. Крупномасштабное машинное обучение вместе с Python / пер. с анг. А. А. Логунова. – М.: ДМК Пресс, 2018. –358с., гл. 4-5.
3. Google. Официальная документация Python-библиотеки «TensorFlow» [PyPl].

<https://www.tensorflow.org/>

1. Введение в нейронные сети, свёрточные нейронные сети [proproprogs].

<https://proproprogs.ru/data>

1. Шарма А. Understanding Activation Functions in Neural Networks [medium] / пер. Милютин И. Функции активации нейросети [Neurohive].

<https://medium.com/the-theory-of-everything/understanding-activation-functions-in-neural-networks-9491262884e0>,

<https://neurohive.io/ru/osnovy-data-science/activation-functions/>

1. Балов Б. Сверточные нейронные сети с нуля [medium]

[https://medium.com/@balovbohdan](https://medium.com/@balovbohdan/%D1%81%D0%B2%D0%B5%D1%80%D1%82%D0%BE%D1%87%D0%BD%D1%8B%D0%B5-%D0%BD%D0%B5%D0%B9%D1%80%D0%BE%D0%BD%D0%BD%D1%8B%D0%B5-%D1%81%D0%B5%D1%82%D0%B8-%D1%81-%D0%BD%D1%83%D0%BB%D1%8F-4d5a1f0f87ec#:~:text=%D0%A1%D0%B2%D0%B5%D1%80%D1%82%D0%BE%D1%87%D0%BD%D1%8B%D0%B5%20%D0%BD%D0%B5%D0%B9%D1%80%D0%BE%D0%BD%D0%BD%D1%8B%D0%B5%20%D1%81%D0%B5%D1%82%D0%B8%20%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%B0%D1%8E%D1%82%20%D0%BD%D0%B0,%D0%B2%20%D1%84%D0%B8%D0%BB%D1%8C%D1)