

КРАЕВОЙ МОЛОДЁЖНЫЙ ФОРУМ
«НАУЧНО-ТЕХНИЧЕСКИЙ ПОТЕНЦИАЛ СИБИРИ»
НОМИНАЦИЯ «НАУЧНЫЙ КОНВЕНТ»

муниципальное бюджетное общеобразовательное учреждение

«Средняя школа № 31»

Муниципальный этап

МАТЕМАТИКА, ИНФОРМАТИКА

Нейронные сети

Выполнил: Матвеев Матвей Александрович

Дата рождения 15.14.2008

МБОУ СШ № 31

Класс 9а

Телефон +7 (923)-311-23-83

E-mail matvclichko@grmail.com

Руководитель: Кокорина Елена Аркадьевна

МБОУ СШ № 31

Должность: учитель информатики

Телефон +7(391)266-97-38

E-mail sch31@mailkrsk.ru

г. Красноярск

2024 г.

ОГЛАВЛЕНИЕ

	Стр.
1 ВВЕДЕНИЕ *	3
2 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ.....	4
2.1 введение.....	4
2.2 нейрон.....	5
2.3 нейронные веса.....	6
2.4 архитектуры нейросетей.....	6
2.5 функции активации и ошибки	К
2.6 обучение нейронных сетей.....	8
3 ПРАКТИЧЕСКАЯ ЧАСТЬ.....	9
3.1 Ход работы.....	Ю
3.2 Как испытывать модель?.....	13
4 ВЫВОДЫ.....	13
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	14

I. ВВЕДЕНИЕ

Актуальность. Данная тема является особенно актуальной, поскольку в современном мире мы отовсюду слышим про возможности нейросетей и несомненно удивляемся. Они являются актуальным и быстро развивающимся направлением в области искусственного интеллекта. Нейронные сети применяются для решения широкого спектра задач, таких как распознавание образов, автоматический перевод, голосовое управление, анализ данных и многое другое.

Постановка проблемы. Часто, принимая та должное факт су шест вования нейросетей мы не задумываемся о том. на что они способны. Значи і не знаем і де их использовать и где они применяются.

Цель моей работы. Изучить принципы работы нейросетей, их архитектуры и доступно объяснить их устройство.

Я поставил и решил следующие задачи:

1. Познакомиться с принципами работы нейросетей.
2. Изучить архитектуры нейросетей и их назначение.
3. Понять, как работают функции потерь и оптимизации, алгоритмы обучения.
4. Сделать демонстрационную работу по изученному материалу.

Методами решения задач стали: сбор информации, обработка данных, сопоставление различных источников (используемая литература и интернет), анализ материалов, экспериментирование и наблюдение, систематизация накопленных знаний, оформление их в проект и предоставление результатов.

Гипотеза: //Нейронные сети дают нам большие возможности в различных сферах.

Моя исследовательская работа включает теоретическую часть: академические знания о нейросетях, и практическую часть: Демонстрационную модель.

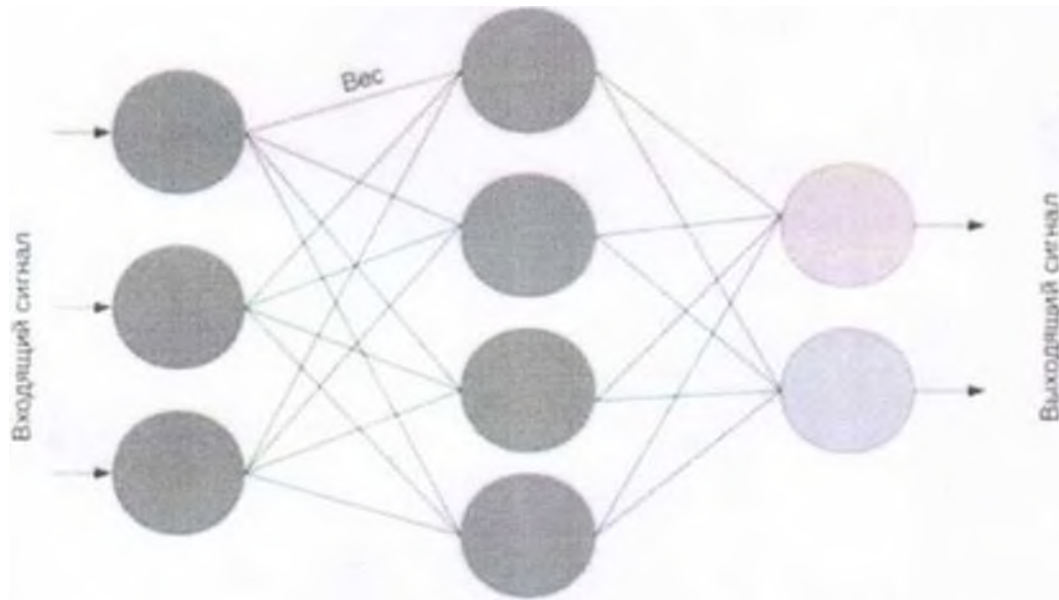
2. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

2.1 Введение

Нейронные сети — это математические модели, которые послужили основой для разработки искусственного интеллекта. Они были вдохновлены биологической нервной системой человека, в которой миллионы нейронов работают параллельно для обработки информации. Нейронные сет используются для решения сложных задач в области:

Распознавания образов	В задачах распознавания образов они обрабатывают входные данные, которые являются изображениями, и на основе обученной модели определяют, что находится на изображении.
Кластеризация данных	В задачах кластеризации нейросеть распределяет объекты на «кластеры» по самостоятельно выявленным признакам.
Обработки естественного языка	В задачах обработки естественного языка нейросеть будет пытаться понять смысл слов и составить осмысленный ответ. Ярчайший пример такой модели это голосовой помощник «Алиса» от компании Яндекс
Предсказания значения	В задачах регрессии они пытаются предсказать значение определенного выходного параметра, например, цены на недвижимость на основе различных характеристик.
Обработка изображений •	В задачах обработки изображений нейросеть способна распознавать объекты, выделять их границы и определять текстуры для анализа, а далее возможного изменения картинки.

Нейронные сети состоят из множества искусственных нейронов, которые объединены в слои, то есть они могут иметь несколько слоев нейронов, и каждый слой будет обрабатывать данные на более высоком уровне. Благодаря этой возможности удастся увеличить точность



работы и сложность задач.

Перцептрон это модель искусственной нейрона, базовый элемент нейронной сети.

Основные компоненты нейронной сети включают в себя следующие типы слоев:

1. Входной слой: Этот слой принимает входные данные, которые затем передаются для дальнейшей обработки.
2. Скрытые или промежуточные слои: Нейронные сети могут иметь один или несколько скрытых слоев, в которых выполняются вычисления и извлекается информация из входных данных. (Глубокие нейронные сети имеют несколько слоев скрытых нейронов, что позволяет

**Входящий
слой**

**Промежуточный
слой(и)**

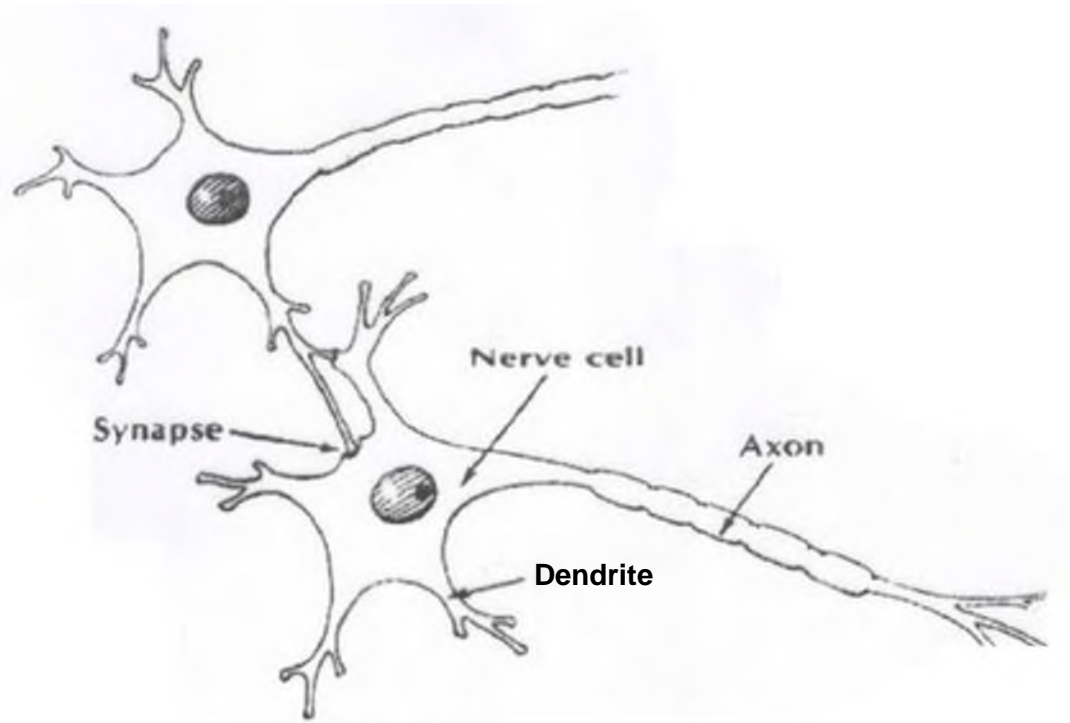
**Выходящий
слой**

им автоматически извлекать особенности из данных и строить сложные модели)

3. Выходной слой: Этот слой генерирует результаты или прогнозы на основе обработанных данных.

2.3 Нейронные веса

Нейроны связаны друг с другом синапсами. Синапсы места соединения аксона одного нейрона с телом или отростком (дендритом или аксоном) другого нейрона. И у каждого синапса есть свой «вес», то есть как бы своя «важность», придаваемая значению сигнала, проходящего через данный синапс. В процессе обучения нейронной сети эти веса изменяются для достижения оптимального результата.



2.4 Архитектура нейросетей

Слои не могут быть устроены как вздумается, и для их структуризации существуют различные архитектуры. Её выбор и настройка нейронной сети зависит от конкретной задачи, для которой она предназначена. Архитектура нейронной сети играет решающую роль в ее способности обучаться и вносить разумные решения в контексте конкретной задачи.

Архитектура нейронной сети включает в себя несколько ключевых компонентов:

1. Тип слоев: Нейронные сети обычно состоят из различных типов слоев (входные слои, скрытые слои и выходные слои). Каждый тип слоя выполняет определенные функции и помогает в обучении сети для конкретных задач.
2. Размер и количество слоев: Архитектура определяет количество нейронов в каждом слое и количество слоев в нейронной сети. Это влияет на сложность и емкость сети, а также скорость и качество ее обучения.
3. Функции активации: Функции активации определяют поведение нейронов и их способность передавать активацию через сеть. Различные функции активации могут быть применены для различных слоев и влияют на способность нейронной сети обучаться.
4. Схема подключения: Архитектура также определяет структуру связей между нейронами в различных слоях. Это является критическим аспектом, который влияет на способность сети извлекать признаки из данных.

Существует множество различных архитектур, которые разработаны для решения разнообразных задач и работают в различных предметных областях. Несколько известных и широко используемых архитектур включают в себя:

1. Прямое распространение (Feedforward) нейронные сети: Это самый простой тип нейронной сети, в котором данные передаются от входного слоя к выходному без циклических связей.
5. Сверточные нейронные сети (CNN): Эта архитектура часто используется для задач обработки изображений и видео. Они способны идентифицировать шаблоны и особенности в изображениях, делая их идеальным выбором для распознавания образов и обработки изображений.
6. Рекуррентные нейронные сети (RNN): Эти сети имеют возможность работать с последовательными данными, что делает их идеальным выбором для задач обработки естественного языка, временных рядов и других задач, связанных с временной зависимостью.
7. Сети с короткой памятью (LSTM) и сети внимания (Attention): Эти архитектуры

расширяют возможности рекуррентных сетей для работы с последовательными данными, обеспечивая более эффективное управление информацией и зависимостями в данных.

8. Глубокие нейронные сети (DNN): Это класс нейронных сетей, имеющий несколько скрытых слоев, что делает их способными извлекать сложные признаки и закономерности из данных.

9. Автокодировщики: Эти нейронные сети используются для сжатия и извлечения ключевой информации из данных, что делает их полезными для задач сжатия данных, уменьшения размерности и извлечения признаков.

2.5 Функции

Функции потерь (иногда также называют функцией ошибки) и функции активации являются важными составляющими нейросетей.

Функция потерь (loss function) измеряет расхождение между предсказанными значениями модели и фактическими значениями. Она показывает, насколько точно нейросеть предсказывает правильные ответы. Задача функции потерь - минимизировать эту разницу.

Функции активации (activation function) определяют выходное значение и активацию искусственного нейрона. Они добавляют нелинейность в нейросеть, позволяя моделировать сложные зависимости в данных. Функции активации применяются к выходному значению каждого нейрона и определяют, будет ли нейрон активирован и с какой интенсивностью.

2.6 Обучение

Тренировка нейронной сети происходит путем подачи большого количества входных данных с соответствующими выходными значениями. На основе различий между предсказанными и фактическими результатами нейронная сеть корректирует свои веса, чтобы повысить точность предсказаний.

Процесс обучения нейронной сети может быть осуществлен с использованием различных алгоритмов:

1. Обучение с учителем: Этот тип обучения включает в себя подачу нейронной сети обучающих данных с известными выходными значениями. На основе различий между предсказанными и фактическими значениями нейронная сеть корректирует свои веса. Примерами такого обучения являются задачи классификации (например, определение, является ли изображение кошкой или собакой) и регрессии (предсказание числовых значений).

2. Обучение без учителя: В этом случае нейронная сеть обучается на размеченных данных,

то есть без пары входных-выходных данных. Целью такого обучения является извлечение структуры и закономерностей из набора данных. Примерами задач обучения без учителя являются кластеризация (группировка данных на основе их сходства) и понижение размерности (представление данных в меньшем количестве измерений, сохраняя важные характеристики).

3. Обучение с подкреплением: В этом случае нейронная сеть обучается взаимодействием с окружающей средой и получением обратной связи в зависимости от своих действий. Цель состоит в том, чтобы нейронная сеть могла сама определять, какие действия следует предпринять для максимизации вознаграждения. Примерами задач обучения с подкреплением являются обучение игры в видеоигры и управление роботами.

Все эти методы используются для тренировки нейронных сетей и создания моделей, позволяющих нейронным сетям учиться, адаптироваться и принимать решения на основе входных данных.

3. ПРАКТИЧЕСКАЯ ЧАСТЬ ПРОЕКТА «НЕЙРОННЫЕ СЕТИ»

Смысл практической части заключается в том, чтобы показать, как сказать, изнутри принцип работы нейронной сети.

Чтобы продемонстрировать вам возможность моей нейросети, мне пришлось проделать невероятно большое количество работы.

Моя демонстрационная модель будет распознавать объекты на картинках. Для достижения такого результата я должен взять модель, загрузить в неё изображение и отобразить результат в понятном для всех виде.

Во-первых: необходимо откуда-то взять модель, и я решил взять обученную с сайта в открытом доступе «TensorFlow». Она уже обучена и в процессе демонстрации ждать придётся только результат определения, не обучения.

Во-вторых: необходимо где-то отобразить результат распознавания модели. Я решил открыть сервер на своём ноутбуке и вывести все данные на него.

Писать я решил в программе «Visual Studio» на трёх языках программирования максимально удобных для этих целей (и меня):

JavaScript функции нейросети, препроцессинга, обработки изображения и работа с HTML

HTML функции сайта

CSS - оформление

3.1 Ход работ ы

Функция загрузки

запуска модели выглядит так:

```
twMira <<смм) l ttrat «mM ■ «rat-<
carat lrag.rrrae» • t' Wwxr.lmt.ultira.)

ra4«UXMiHan/aatUMI_aj.*raLI_IM^IU;i.*ialfLEtUM^Ui>ctraU.11

tn* l)

П

I) » I
«rat
< ratal
carat • a 1,
carat txt" a rail te tcpHU.ra4"«>(Ufifi>, k) carat 'iratMilrra * tad. IW'Н Uiax,ar()i carat arraarauliac • tea* valwra aa'atyaO, aa

C'ra oaaaja.
«и «ravra'a

raaaalc lrq/cialatalMUr. rearatiUUac)

carat m«lt a fl
]>.Cl HTML.t.(!luit:-> )(ciraUaeM«al»JJJ *ara«>lltiac(«J lwama's «Ultit'UNtcm )(• laaalaatl.rctIHl »rv»a"lt It U«(l l. jw4c' actararat «aciacaea )(• la«IM>«ra|||] »r»»»ll It laaffli
l;

mull affrarat)
ПИ. racu.'l 1Ю. rraux" crarat t
```

Здесь я импортирую модель, запускаю её и возвращаю значение предсказания нейросети.

Важно понимать, что все ошибаются, а особенно алгоритмы, поэтому я сохранил 3 результата, а не один.

В этой нейросети используются функция активации для запуска модели и функция ошибки для вычисления точности ответов. Использовался метод обучения с учителем - я дождался «предсказания» и давал нейросети сравнить с входными данными. А тип архитектуры: сверточная нейросеть.

Следом стоило загрузить изображение в модель. Для этого я воспользуюсь «IT.ML». Этот язык программирования объединит все части программы в одно целое, покажет предсказания и поможет получить входные данные (фотографию).

```

9      <h1 class="headText">Загрузка изображений</h1>
10     <form action="/upload" method="POST" enctype="multipart/form-data">
11     <input class="upload-btn" type="file" name="image" accept="image/*" !>
12     <br />
13     <button id="upload-button" type="submit">Нажмите здесь</button>
14     </form>
15     <br />
16     <div id="loader" />
17     <img id="img" />
18     <div class="container">
19     <h5 class="text">Первое предсказание</h5>
20     <div class="bar" />
21     <h5 class="text">Второе предсказание</h5>
22     <div class="bar" />
23     <h5 class="text">Третье предсказание</h5>
24     <div class="bar" />
25     </div>
26     <script src="public/script.js" />
27 </body>
28 </html>
29

```

Обработка фотографий должна в определённом разрешении и следующая часть кода преобразует разрешение фотографии (нормирует) в необходимое для анализа моделью (224x224 px). Функция преобразования:

```

1  ссылка
function preprocessImageTensor() {
  const widthToHeight = imageTensor.shape[1] / imageTensor.shape[0];
  let squareCrop;
15  if (widthToHeight > 1) {
16  const heightToWidth = imageTensor.shape[0] / imageTensor.shape[1];
17  const cropTop = (1 - heightToWidth) / 2;
18  const cropBottom = 1 - cropTop;
19  squareCrop = [[cropTop, 0, cropBottom, 1]];
20  } else {
21  const cropLeft = (1 - widthToHeight) / 2;
22  const cropRight = 1 - cropLeft;
23  squareCrop = [[0, cropLeft, 1, cropRight]];
24  }
25  const crop = tf.image.cropAndResize(
26  tf.expandDims(imageTensor), squareCrop, [0], [224, 224]);
27  return crop.div(255);
28 }

```

Так выглядит код для создания сервера на который я загружу все данные:

```

39 app.get('/', (req, res) => {
31  fs.readFile('page.html', (err, data) => {
3    if (err) {
      res.writeHead(500, { 'Content-Type': 'text/plain' }); res.end("Internal server error")
    } else {
      res.writeHead(200, { 'Content-Type': 'text/html' }); res.end(data)
    }
  });
}

```

```

app.listen(3600, () => { console.log("RUM SERVER!");
});

```

(В ходе работы я сменил стиль оформления, так как от белого экрана уставали глаза)

После написания функциональной части кода настало время уделить внимание внешнему виду проекта.

Сейчас всё выглядит! так:

Загрузка изображения «пм:и|м»СибИ».

Очень недоработано и непонятно. Чтобы ио корректировать я поДірузил CSS файл. Он добавит красок, отступов, анимаций, да и в целом удовольствия от пользования.

кпл<. а>фи«41.ч:

Загрузка изображения

ВwOесиго ф*№ **Ал *

I leffоov пре КШ ЫПМе

Вк>|м>с иргККЛХШП'

Для демонстрации работы моей модели я подгрузил в HTML файл остальные данные и запустил программу.

Как испытывать модель?

В область для загрузки файла '***' * загружаем картинку, образ на которой хотим распознать, подтверждаем действие кнопкой классифицировать и ждём результат.

Результаты получаются на английском, так как обучал модель не я и менять список на котором училась модель нерационально.

4.ВЫВОДЫ

Были выполнены цели, поставленные в ходе работы.

Нейронные сети нашли применение в различных областях. Их способность обрабатывать сложные данные и выявлять неочевидные зависимости делает их мощным инструментом для анализа информации.

Они действительно открывают нам большие возможности. С помощью нейросетей можно в разы ускорить процессы, ранее занимавшие значительную часть времени. Нейросети это не какое-то не осязаемое будущее, это уже давно наступившая реальность.

Важно научиться работать с новыми инструментами и использовать их с умом, ведь с большой силой приходит и большая ответственность.

СПИСОК ИСПОЛЬЗОВАННЫХ источников

>1 пользовался сайтами:

[https://liabr.com-riv/articles.](https://liabr.com-riv/articles)

[https://ru.wikipedia.org,](https://ru.wikipedia.org)

<https://vandcx.ru/>