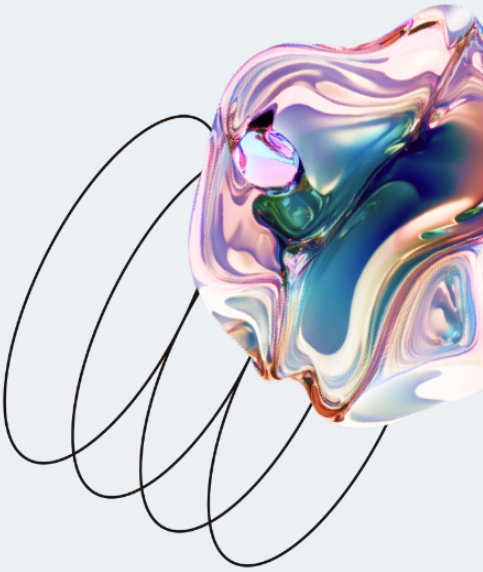
GeekBrains



Введение в нейронные сети

Архитектура нейронных сетей

Оглавление

Введение

Всем привет и добро пожаловать на курс Архитектура нейронных сетей. Меня зовут Антон. Я руковожу командой Дата Сайнтистов в компании SAP. За время работы в дата сайнс я проектировал и внедрял различные нейросетевые технологии в бизнес-процессы, занимался компьютерным зрением и сейчас плотно занимаюсь проектами связанными с NLP и большими языковыми моделями наподобие ChatGPT.
На этом курсе мы с вами познакомимся с нейронными сетями рассмотрим различные типы нейронных сетей. Попробуем разобраться как они организованы и чем отличаются их архитектуры. Построим как применять на практике различные нейросети. В последней части курса рассмотрим, что такое PyTorch и как с его помощью можно проектировать и тренировать различные нейронные сети.

А что же будет на нашем первом уроке? Мы с вами поговорим о том, что же такое нейросети и какие виды нейронных сетей бывают. Начнем мы с обсуждения искусственных нейронных сетей и того, как они вдохновлены реальными биологическими нейронными сетями в наших собственных телах. Далее мы рассмотрим классический перцептрон и роль, которую он сыграл в истории нейронных сетей. В продолжение урока рассмотрим историю развития и как менялись нейронные сети до сегодняшнего дня. Ну и в конце посмотрим в каких сферах сейчас применяются искусственные нейронные сети.

Словарь терминов

ИНС — это вычислительная система, которая пытается имитировать (или, по крайней мере, вдохновляется) нейронные связи в нашей нервной системе.
Нейронная сеть прямого распространения — это искусственная нейронная сеть, в которой связи между элементами не *образуют* цикл
CNN — это особый вид нейронных сетей, особенно хорошо подходящий для обработки данных изображений, таких как 2D-изображения или даже 3D-видеоданные
GAN — это особое семейство нейронных сетей, основной, но не единственной целью которых является создание синтетических данных, которые точно имитируют заданный набор данных реальных данных
Большие языковые модели (LLM) — это революционная категория многоцелевых и мультимодальных (принимающих входные изображения, аудио и текст) глубоких нейронных сетей

Введение в нейронные сети

Нейронные сети являются строительными блоками систем глубокого обучения. Для того, чтобы быть успешным в глубоком обучении, мы должны начать с изучения основ нейронных сетей, в том числе архитектура, типы узлов, и алгоритмы для обучения нашей сети.

Мы начнем с обзора нейронных сетей высокого уровня и мотивации, стоящей за ними, включая их отношение к биологии человеческого разума. Оттуда мы обсудим наиболее распространенный тип архитектуры, нейронные сети прямого распространения. Мы также кратко обсудим концепцию нейронное обучение и как это позже будет соотноситься с алгоритмами, которые мы используем для обучения нейронных сетей.

Что такое нейронные сети?

Многие задачи, связанные с интеллектом, распознаванием образов и обнаружением объектов, чрезвычайно сложно автоматизировать, но, похоже, животные и маленькие дети выполняют их легко и естественно.
Например, как ваша домашняя собака узнает вас, владельца, по сравнению с совершенно незнакомым человеком? Как маленький ребенок учится различать школьный автобус и обычный маршрутный автобус ? И как наш собственный мозг каждый день подсознательно выполняет сложные задачи по распознаванию образов, даже не замечая этого?
Ответ кроется в нашем собственном теле. Каждый из нас содержит реальную биологическую нейронную сеть, связанную с нашей нервной системой — эта сеть состоит из большого количества взаимосвязанных нейронов (нервных клеток). Слово «нейронный» является прилагательной формой слова «нейрон», а «сеть» обозначает графоподобную структуру; следовательно, «искусственная нейронная сеть» — это вычислительная система, которая пытается имитировать (или, по крайней мере, вдохновляется) нейронными связями в нашей нервной системе. Искусственные нейронные сети также называют «нейронными сетями» или «искусственными нейронными системами». Искусственные нейронные сети принято сокращать и называть их «ИНС».
Чтобы система считалась ИНС, она должна содержать помеченную структуру ориентированного графа, в которой каждый узел графа выполняет некоторые простые вычисления. Из теории графов мы знаем, что ориентированный граф состоит из набора узлов (т. е. вершин) и набора связей (т. е. ребер), которые связывают вместе пары узлов. На рисунке 1 мы можем увидеть пример такого графа ИНС.

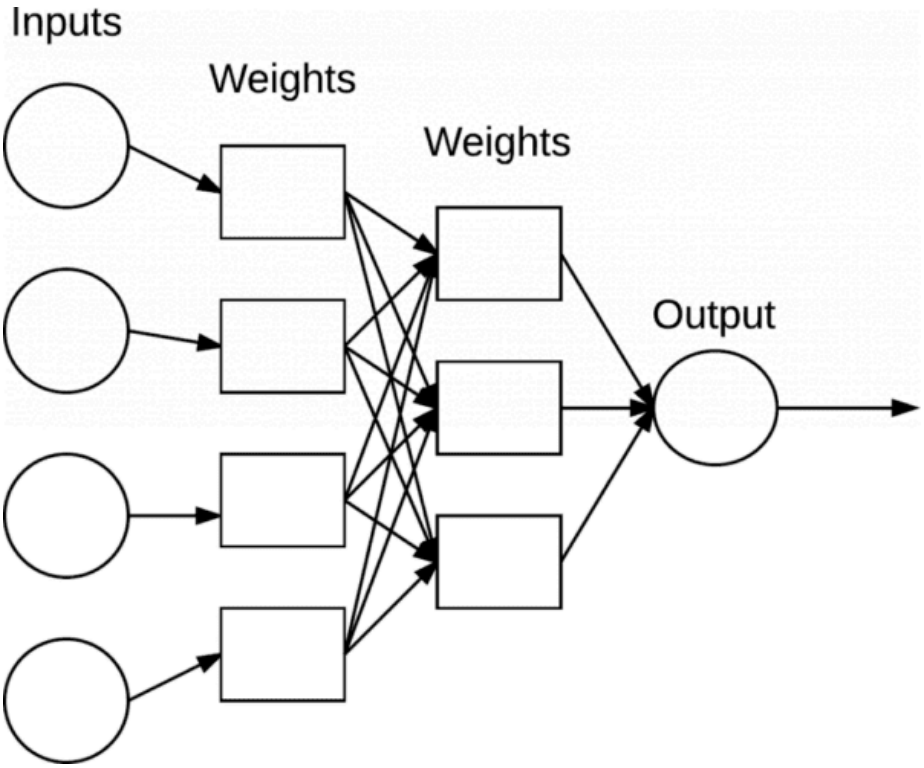


Рисунок 1: Простая архитектура нейронной сети. Входные данные представлены в сети. Каждое соединение передает сигнал через два скрытых уровня сети. Последняя функция вычисляет метку выходного класса.

Каждый узел выполняет простое вычисление. Затем каждое соединение передает сигнал (т. е. результат вычислений) от одного узла к другому, помеченный весом, указывающим степень усиления или ослабления сигнала. Некоторые соединения имеют большие положительные веса , которые усиливают сигнал, что указывает на то, что сигнал очень важен при классификации. Другие имеют отрицательные веса, уменьшая силу сигнала и тем самым указывая, что выход узла менее важен для окончательной классификации. Мы называем такую систему искусственной нейронной сетью, если она состоит из графовой структуры (как на рисунке 1) с весами соединений, которые можно изменить с помощью алгоритма обучения.

Связь с биологией

Наш мозг состоит примерно из 10 миллиардов нейронов, каждый из которых связан примерно с 10 000 других нейронов. Тело клетки нейрона называется сомой, где

входы (дендриты) и выходы (аксоны) соединяют одну сому с другой сомой (рис. 2).

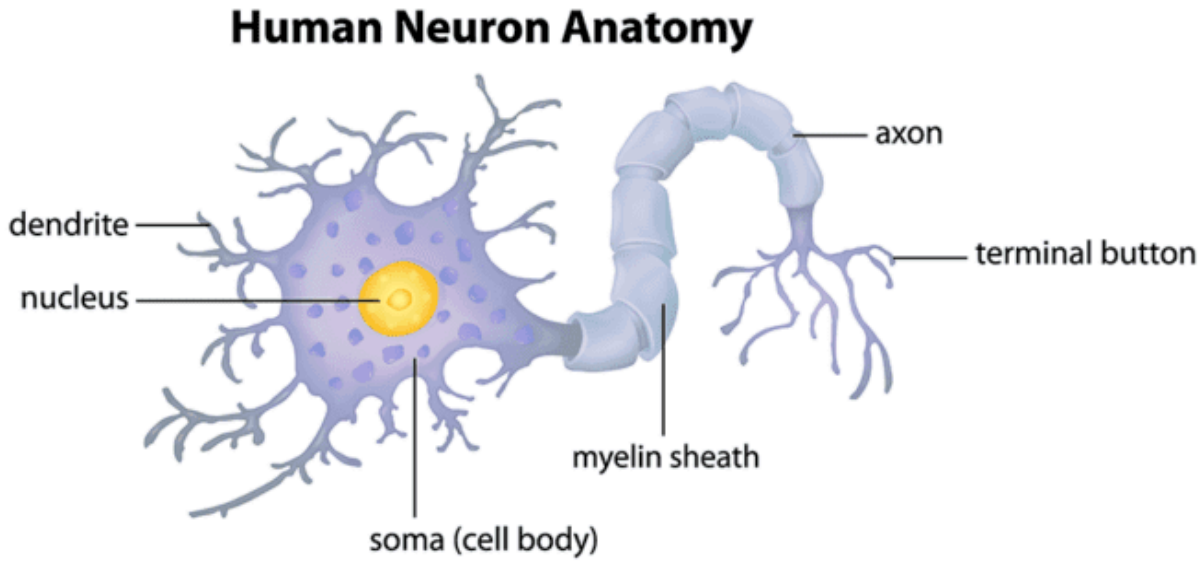


Рисунок 2: Структура биологического нейрона. Нейроны связаны с другими нейронами через свои дендриты и нейроны.

Каждый нейрон получает электрохимические сигналы от других нейронов в своих дендритах. Если эти электрические входы достаточно мощны, чтобы активировать нейрон, то активированный нейрон передает сигнал по своему аксону, передавая его дендритам других нейронов. Эти прикрепленные нейроны также могут активироваться, продолжая таким образом процесс передачи сообщения.

Ключевой вывод здесь заключается в том, что срабатывание нейрона — это бинарная операция : нейрон либо срабатывает, либо нет. Разных «сортов» активации нет. Проще говоря, нейрон сработает только в том случае, если общий сигнал, полученный в соме, превысит заданный порог.

Однако имейте в виду, что ИНС просто созданы на основе того, что мы знаем о мозге и о том, как он работает. Цель глубокого обучения — не имитировать работу нашего мозга, а скорее взять те фрагменты, которые мы понимаем , и позволить нам провести аналогичные параллели в нашей собственной работе. В конце концов, мы недостаточно знаем о нейробиологии и более глубоких функциях мозга, чтобы правильно моделировать работу мозга — вместо этого мы черпаем вдохновение и двигаемся дальше.

Основной единицей вычислений в нейронной сети является нейрон, часто называемый узлом или единицей. Он получает входные данные от некоторых других узлов или из внешнего источника и вычисляет выходные данные. Каждому входному сигналу присвоен вес (w), который присваивается на основе его относительной важности по отношению к другим входным данным. Узел применяет функцию к взвешенной сумме своих входов.

Идея состоит в том, что синаптические силы (веса w) являются обучаемыми и контролируют силу воздействия и его направление: возбуждающее (положительный вес) или тормозное (отрицательный вес) одного нейрона на другой. В базовой модели дендриты переносят сигнал в тело клетки, где все они суммируются. Если итоговая сумма превышает определенный порог, нейрон может сработать , отправив импульс по своему аксону. В вычислительной модели мы предполагаем, что точное время появления всплесков не имеет значения и что информацию передает только частота срабатываний. мы моделируем скорость срабатывания нейрона с помощью функции активации (ex сигмовидная функция) , которая представляет частоту спайков вдоль аксона.

Архитектура нейронных сетей

Из приведенного выше объяснения мы можем заключить, что нейронная сеть состоит из нейронов. Биологически нейроны связаны через синапсы, по которым течет информация (веса для нашей вычислительной модели). Когда мы тренируем нейронную сеть, мы хотим, чтобы нейроны сработали всякий раз, когда они изучают определенные шаблоны из данных, и мы моделируем скорость стрельбы с помощью функции активации.

Но это еще не все...

- **Входные узлы (входной уровень):** в этом слое никакие вычисления не выполняются, они просто передают информацию на следующий уровень (большую часть времени скрытый уровень). Блок узлов также называется **слоем**.

- **Скрытые узлы (скрытый уровень).** В скрытых слоях выполняется промежуточная обработка или вычисления, они выполняют вычисления, а затем передают веса (сигналы или информацию) из входного слоя на следующий уровень (другой скрытый уровень или на выходной слой). Нейронную сеть можно создать и без скрытого слоя, и я объясню это позже.

- **Выходные узлы (выходной уровень):** здесь мы, наконец, используем функцию активации, которая соответствует желаемому выходному формату (например, softmax для классификации).

- *Связи* **и веса:** Сеть состоит из соединений, каждое соединение передает выход нейрона i на вход нейрона *j* . В этом смысле *i* является предшественником *j* , а *j* является преемником *i* . Каждому соединению присваивается вес *Wij.*

- **Функция активации:** функция **активации** узла определяет выходные данные этого узла с учетом входных данных или набора входных

данных. Стандартную схему компьютерного чипа можно рассматривать как цифровую сеть функций активации, которые могут быть «ВКЛ» (1) или «ВЫКЛ» (0), в зависимости от входа. Это похоже на поведение линейного перцептрона в нейронных сетях. Однако именно *нелинейная* функция активации позволяет таким сетям решать нетривиальные задачи, используя лишь небольшое количество узлов. В искусственных нейронных сетях эту функцию еще называют передаточной функцией.

- **Правило обучения. Правило** *обучения* — это правило или алгоритм, который изменяет параметры нейронной сети, чтобы данный вход в сеть давал предпочтительный результат. Этот процесс *обучения* обычно сводится к изменению весов и порогов.

Искусственные нейронные сети

Начнем с взгляда на базовые нейронные сети, которые выполняют простое взвешенное суммирование входных данных на Рисунке 3. Значения x1, x2, и, x3 являются ли входы к нашей ИНС и обычно соответствуют а один ряд (то есть точка данных) из нашей матрицы дизайна. Постоянное значение 1 - это наше смещение, которое, как предполагается, встроено в матрицу проектирования. Мы можем думать об этих входах как о векторах входных признаков в ИНС.

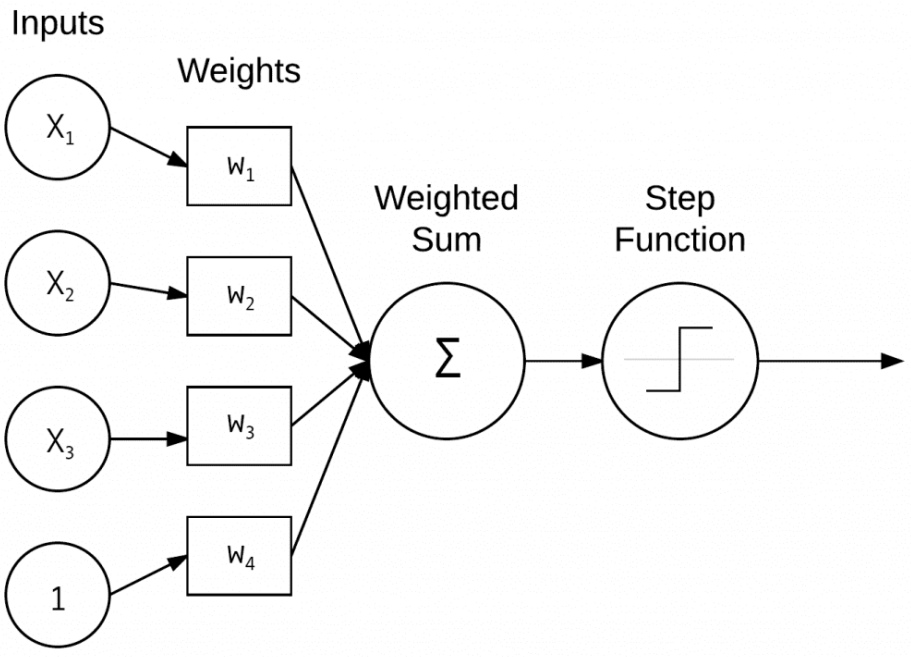


Рисунок 3: Простой NN, принимающий взвешенную сумму ввода xx и веса w. Затем эта взвешенная сумма передается через функцию активации, чтобы определить, срабатывает ли нейрон.

На практике эти входные данные могут быть векторами, используемыми для количественной оценки содержимого изображения систематическим, предопределенным способом (например, гистограммы цвета, изображения), Гистограмма Ориентированных Градиентов, Местные Бинарные Узоры, и т.д.). В контексте глубокого обучения эти входы являются интенсивность сырых пикселей самих изображений.

Каждый xx подключен к нейрону через вектор веса WW состоит из w1, w2, ..., wn, это означает, что для каждого входа xn у нас также есть связанный вес wn.

Наконец-то выходной узел справа от Рисунок 3 берет взвешенную сумму, применяет функцию активации ф (используется для определения того, запускается ли нейрон “ ” или нет), и выводит значение. Выражая вывод математически, вы обычно сталкиваетесь со следующими тремя формами:

• f(w1x1 +w2x2 + ⋯ +wnxn)

• f(∑ni=1wixi)

• Или проще, f(net), где net = ∑ni=1wixi

Независимо от того, как выражено выходное значение, поймите, что мы просто берем взвешенную сумму входов, а затем применяем функцию активации f.

Типы нейронных сетей

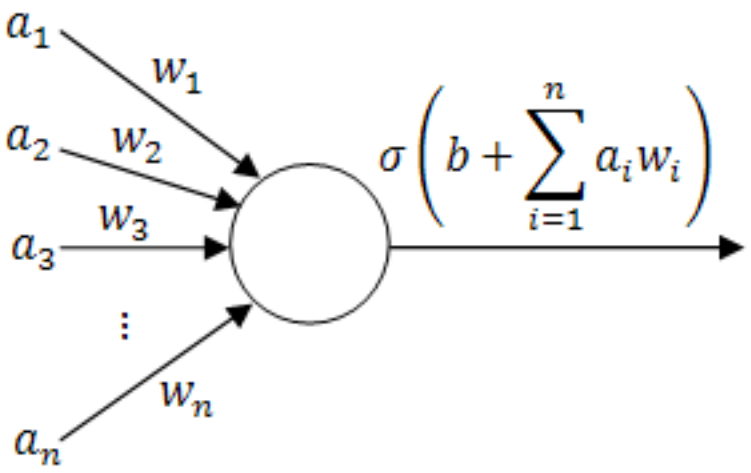
Существует множество классов нейронных сетей, и у этих классов также есть подклассы. Здесь я перечислю наиболее используемые из них и упрощу дальнейшее путешествие по изучению нейронных сетей.

1. Нейронная сеть прямого распространения

Нейронная сеть прямого распространения — это искусственная нейронная сеть, в которой связи между элементами не *образуют* цикл. В этой сети информация движется только в одном направлении — вперед, от входных узлов через скрытые узлы (если таковые имеются) к выходным узлам. В сети нет циклов и петель.
Мы можем выделить два типа нейронных сетей прямого распространения:

1.1. Однослойный персептрон

Это простейшая нейронная сеть прямого распространения, которая не содержит скрытых слоев. Это означает, что она состоит только из одного слоя выходных узлов. Это называется единичным, потому что при подсчете слоев мы не включаем входной слой, причина этого в том, что на входном слое не выполняются никакие вычисления, входные данные передаются непосредственно на выходные через ряд весов.



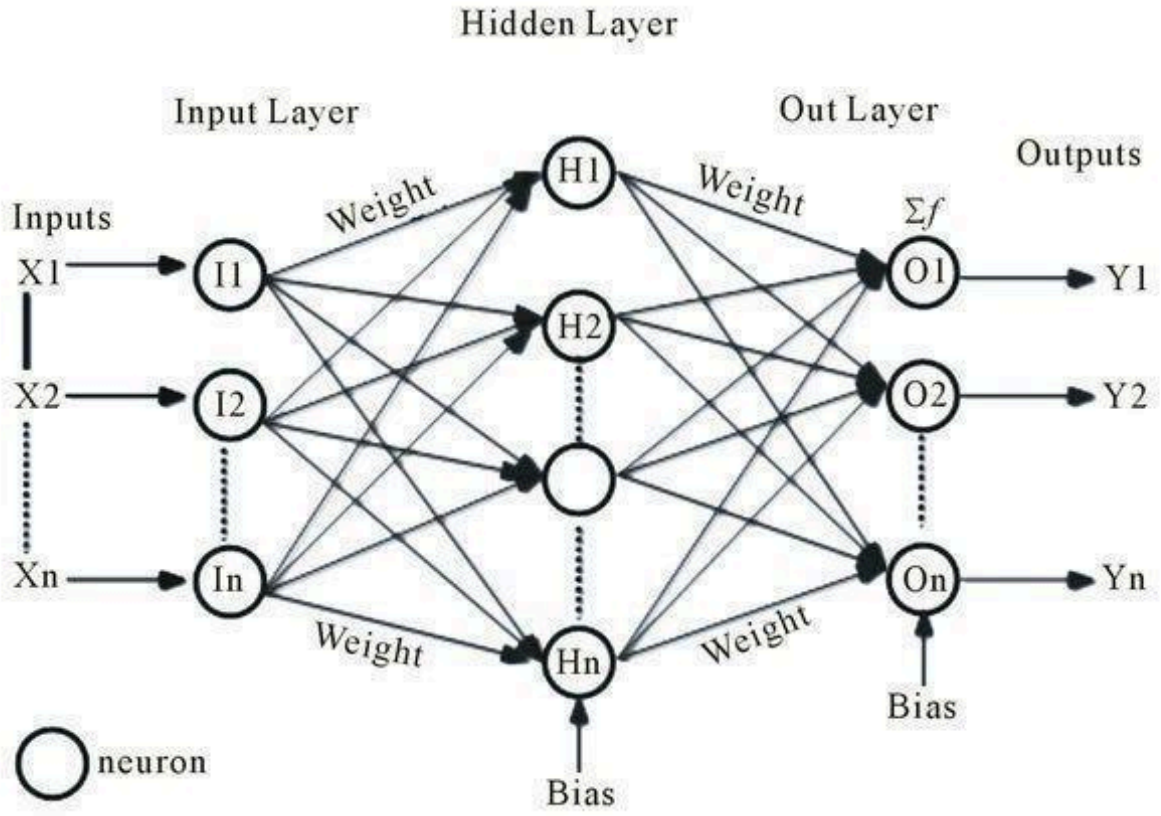
Простой персептрон

1.2. Многослойный перцептрон (MLP)
По своей сути сети прямого распространения организуют однонаправленный путь информации. Все начинается с входного слоя, содержащего n нейронов, куда первоначально принимаются данные. Этот уровень служит точкой входа в сеть, действуя как приемник входных объектов, которые необходимо обработать. Оттуда данные отправляются в преобразующее путешествие по скрытым слоям сети.

Одним из важных аспектов сетей прямого распространения является их связная структура, а это означает, что каждый нейрон в слое сложно связан со всеми остальными нейрономи в этом слое. Эта взаимосвязь позволяет сети выполнять вычисления и фиксировать взаимосвязи внутри данных. Это похоже на сеть связи, где каждый узел играет роль в обработке информации.

Когда данные проходят через скрытые слои, они подвергаются серии вычислений. Каждый нейрон в скрытом слое получает входные данные от всех нейронов предыдущего слоя, применяет к этим входным данным взвешенную сумму, добавляет вес смещения, а затем передает результат через функцию активации (обычно ReLU, Sigmoid или tanH). Эти математические операции позволяют сети извлекать соответствующие закономерности из входных данных и фиксировать сложные нелинейные связи внутри данных. Именно здесь FFNN(нейронные сети прямого распространения) действительно превосходят более поверхностные

модели машинного обучения.



Однако на этом все не заканчивается. Настоящая сила FFNN заключается в их способности адаптироваться. Во время обучения сеть корректирует свои веса, чтобы минимизировать разницу между ее прогнозами и фактическими целевыми значениями. Этот итерационный процесс, часто основанный на алгоритмах оптимизации, таких как градиентный спуск, называется обратным распространением ошибки. Обратное распространение ошибки позволяет FFNN фактически учиться на данных и повышать точность прогнозов или классификаций. Будучи мощными и универсальными, FFNN имеют некоторые существенные ограничения. Например, им не удается уловить последовательность и временные/синтаксические зависимости в данных – два важнейших аспекта для задач языковой обработки и анализа временных рядов. Необходимость преодоления этих ограничений побудила к развитию нового типа архитектуры нейронных сетей. Этот переход проложил путь к рекуррентным нейронным сетям (RNN), которые представили концепцию циклов обратной связи для лучшей обработки последовательных данных.

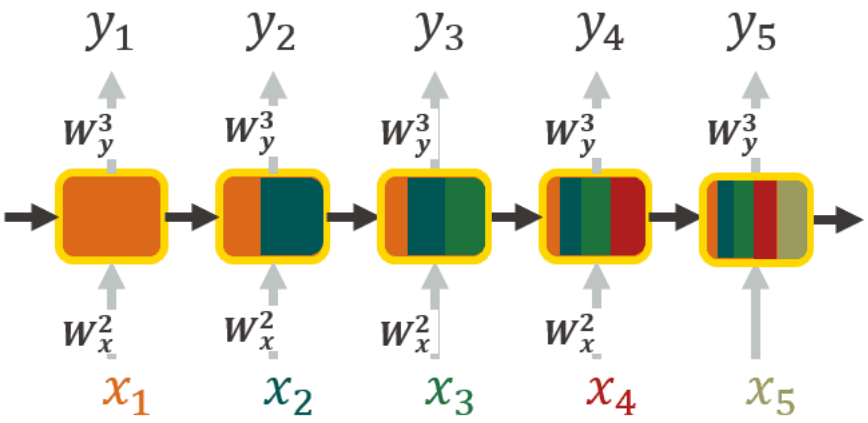
2. Рекуррентные нейронные сети

По своей сути RNN имеют некоторое сходство с FFNN. Они также состоят из слоев взаимосвязанных узлов, обрабатывающих данные для прогнозирования или классификации. Однако их ключевое отличие заключается в способности обрабатывать последовательные данные и фиксировать временные зависимости.

В FFNN информация течет по одному однонаправленному пути от входного уровня к выходному слою. Это подходит для задач, где порядок данных не имеет большого значения. Однако при работе с такими последовательностями, как данные

временных рядов, язык или речь, решающее значение имеет поддержание контекста и понимание порядка данных. Вот где проявляют себя RNN.

RNN вводят концепцию петель обратной связи. Они действуют как своего рода «память» и позволяют сети поддерживать скрытое состояние, которое фиксирует информацию о предыдущих входных данных и влияет на текущие входные и выходные данные. В то время как традиционные нейронные сети предполагают, что входные и выходные данные независимы друг от друга, выходные данные рекуррентных нейронных сетей зависят от предшествующих элементов в последовательности. Этот механизм рекуррентных соединений делает RNN особенно подходящими для обработки последовательностей путем «запоминания» прошлой информации.
Еще одной отличительной особенностью рекуррентных сетей является то, что они имеют один и тот же весовой параметр на каждом уровне сети, и эти веса корректируются с использованием алгоритма обратного распространения ошибки во времени (BPTT), который немного отличается от традиционного обратного распространения ошибки, поскольку он специфичен для данных последовательности. .



Развернутое представление RNN, где каждый входной сигнал обогащен контекстной информацией, поступающей из предыдущих входных данных. Цвет представляет распространение контекстной информации (изображение автора).

Однако традиционные RNN имеют свои ограничения. Хотя теоретически они должны быть в состоянии фиксировать долгосрочные зависимости, в действительности им трудно сделать это эффективно, и они могут даже страдать от проблемы исчезающего градиента, которая препятствует их способности изучать и запоминать информацию на протяжении многих временных шагов.
Именно здесь в игру вступают блоки долговременной краткосрочной памяти (LSTM). Они специально разработаны для решения этих проблем путем включения в свою структуру трех вентилей: вентиля забывания, входного вентиля и выходного вентиля.

- **Ворота забывания** : эти ворота решают, какую информацию из временного шага следует отбросить или забыть. Изучая состояние ячейки и текущие

входные данные, он определяет, какая информация не важна для прогнозирования в настоящем.

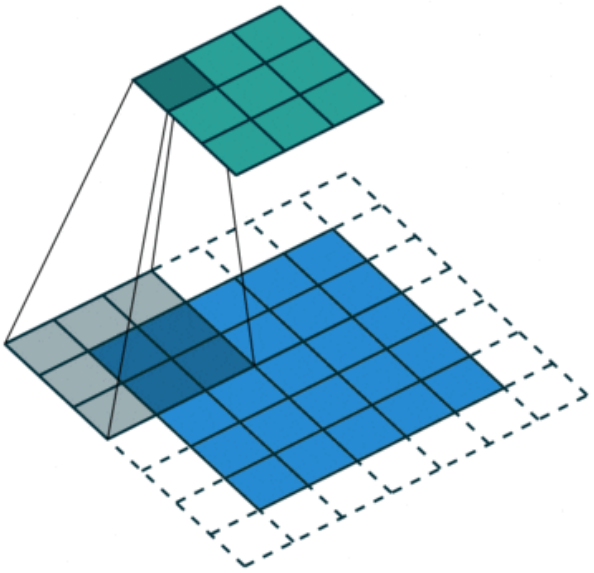
- **Входные ворота** : эти ворота отвечают за включение информации в состояние ячейки. Он учитывает как входные данные, так и предыдущее состояние ячейки, чтобы решить, какую новую информацию следует добавить для улучшения ее состояния.
- **Выходной вентиль** : этот вентиль определяет, какой вывод будет генерироваться модулем LSTM. Он учитывает как текущие входные данные, так и обновленное состояние ячейки, чтобы получить выходные данные, которые можно использовать для прогнозов или передать на временные шаги.

Таким образом, RNN и особенно модули LSTM предназначены для последовательных данных, что позволяет им сохранять память и фиксировать временные зависимости, что является критически важной возможностью для таких задач, как обработка естественного языка, распознавание речи и прогнозирование временных рядов.
По мере того как мы отходим от RNN, фиксирующих последовательные зависимости, эволюция продолжается с помощью сверточных нейронных сетей (CNN). В отличие от RNN, CNN превосходно извлекают пространственные признаки из структурированных данных в виде сетки, что делает их идеальными для задач распознавания изображений и образов. Этот переход отражает разнообразные применения нейронных сетей для разных типов и структур данных.

3. Сверточная нейронная сеть (CNN)
CNN — это особый вид нейронных сетей, особенно хорошо подходящий для обработки данных изображений, таких как 2D-изображения или даже 3D-видеоданные. Их архитектура основана на многослойной нейронной сети прямого распространения как минимум с одним сверточным слоем.
Что выделяет CNN, так это их сетевое подключение и подход к извлечению признаков, который позволяет им автоматически выявлять соответствующие закономерности в данных. В отличие от традиционных FFNN, которые соединяют каждый нейрон одного слоя с каждым нейроном следующего, CNN используют скользящее окно, известное как ядро или фильтр . Это скользящее окно сканирует входные данные и особенно полезно для задач, где важны пространственные отношения, например, идентификация объектов на изображениях или отслеживание движения в видео. При перемещении ядра по изображению между ядром и значениями пикселей выполняется операция свертки (со строго математической точки зрения эта операция представляет собой взаимную корреляцию) и применяется нелинейная функция активации, обычно ReLU. Это дает высокое значение, если объект присутствует в патче изображения, и маленькое значение, если его нет.

Вместе с ядром добавление и точная настройка гиперпараметров, таких как шаг (т. е. количество пикселей, на которое мы сдвигаем ядро) и скорость расширения (т. е. промежутки между каждой ячейкой ядра), позволяет сети фокусироваться на

конкретных особенностях, распознавая закономерности и детали в конкретных регионах, не рассматривая всю входную информацию сразу.



Операция свертки с длиной шага = 2 (ссылка - GIF).

Некоторые ядра могут специализироваться на обнаружении краев или углов, в то время как другие могут быть настроены на распознавание более сложных объектов, таких как кошки, собаки или уличные знаки, на изображении. Объединяя несколько сверточных и объединяющих слоев, CNN создают иерархическое представление входных данных, постепенно абстрагируя функции от низкого уровня к высокому, точно так же, как наш мозг обрабатывает визуальную информацию.
Хотя CNN преуспевают в извлечении признаков и произвели революцию в задачах компьютерного зрения, они действуют как пассивные наблюдатели, поскольку не предназначены для генерации новых данных или контента. Это не является внутренним ограничением сети как такового, но наличие мощного двигателя и отсутствие топлива делает быструю машину бесполезной. Действительно, сбор реальных и значимых изображений и видеоданных, как правило, труден и дорог, и они часто сталкиваются с ограничениями в области авторских прав и конфиденциальности данных. Это ограничение привело к разработке новой парадигмы, основанной на CNN, но знаменующей переход от классификации изображений к творческому синтезу: генеративно-состязательные сети (GAN).

4. Генеративно-состязательные сети (GAN)

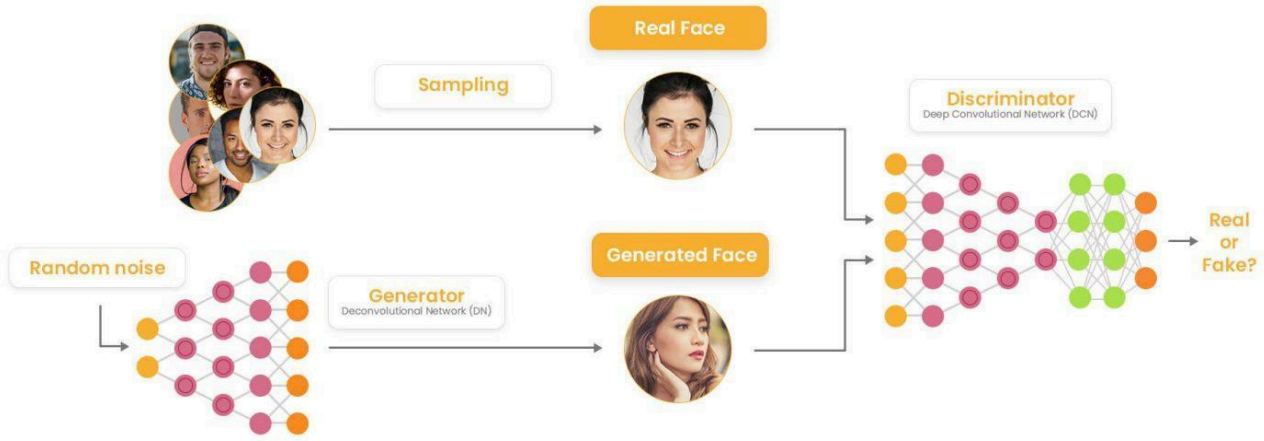
GAN — это особое семейство нейронных сетей, основной, но не единственной целью которых является создание синтетических данных, которые точно имитируют заданный набор данных реальных данных. В отличие от большинства нейронных сетей, гениальная архитектура GAN состоит из двух основных моделей:

- **Модель генератора** . Первым игроком в этом дуэте нейронных сетей является модель генератора. Перед этим компонентом стоит увлекательная миссия: учитывая случайный шум или входные векторы, он стремится создать искусственные образцы, максимально похожие на реальные образцы. Представьте себе его как фальсификатора

произведений искусства, пытающегося создать картины, неотличимые от шедевров.

- **Модель дискриминатора** . Роль противника выполняет модель дискриминатора. Его задача — отличить сгенерированные образцы, созданные генератором, от подлинных образцов из исходного набора данных. Думайте об этом как о знатоке искусства, пытающемся обнаружить подделки среди подлинных произведений искусства.

Вот где происходит волшебство: GAN участвуют в непрерывном состязательном танце. Генератор стремится улучшить свое мастерство, постоянно совершенствуя свои творения, чтобы они стали более убедительными. Тем временем дискриминатор становится более проницательным детективом, оттачивая свою способность отличать настоящее от подделки.



Архитектура GAN (ссылка на источник)

По мере обучения это динамическое взаимодействие между генератором и дискриминатором приводит к потрясающим результатам. Генератор стремится генерировать настолько реалистичные образцы, что даже дискриминатор не сможет отличить их от подлинных. Это соревнование заставляет оба компонента постоянно совершенствовать свои способности.

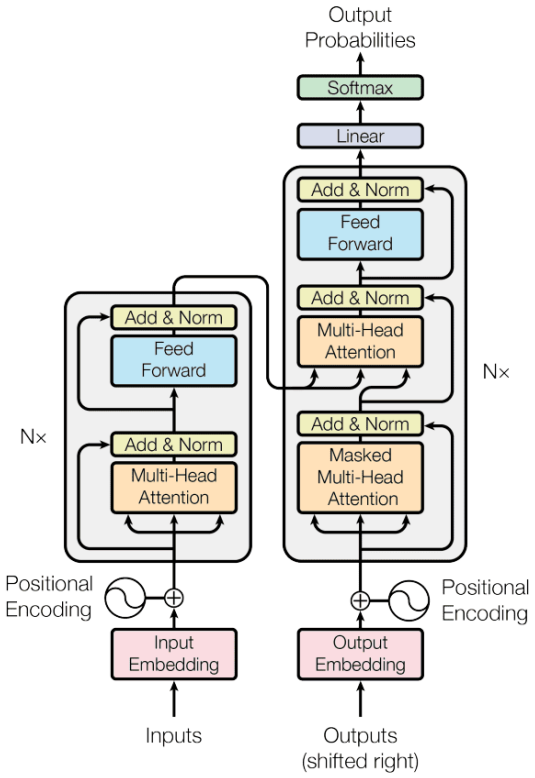
Результат? Генератор, который становится удивительно искусным в создании данных, которые кажутся подлинными, будь то изображения, музыка или текст. Эта возможность привела к появлению замечательных приложений в различных областях, включая синтез изображений, увеличение данных, перевод изображений в изображения и редактирование изображений.

GAN стали пионерами в создании реалистичного изображения и видеоконтента, противопоставив генератор дискриминатору. Расширяя потребность в творчестве и расширенных операциях от изображений до последовательных данных, были представлены модели для более сложного понимания естественного языка, машинного перевода и генерации текста. Это положило начало разработке Transformers, замечательной архитектуры глубоких нейронных сетей, которая не только превзошла предыдущие архитектуры за счет эффективного улавливания языковых зависимостей и семантического контекста на больших расстояниях, но также стала бесспорной основой новейших приложений, управляемых искусственным интеллектом.

5. Трансформеры

Разработанные в 2017 году, Transformers могут похвастаться уникальной функцией, которая позволяет им заменять традиционные повторяющиеся слои: механизмом самообслуживания, который позволяет им моделировать сложные отношения между всеми словами в документе, независимо от их положения. Это делает Трансформеры превосходными в решении проблемы долгосрочных зависимостей на естественном языке. Архитектура трансформатора состоит из двух основных строительных блоков:

- **Кодировщик.** Здесь входная последовательность внедряется в векторы, а затем подвергается воздействию механизма внутреннего внимания. Последний вычисляет оценки внимания для каждого токена, определяя его важность по отношению к другим. Эти оценки используются для создания взвешенных сумм, которые передаются в FFNN для создания контекстно-зависимых представлений для каждого токена. Несколько слоев кодировщика повторяют этот процесс, расширяя возможности модели захватывать иерархическую и контекстную информацию.
- **Декодер.** Этот блок отвечает за генерацию выходных последовательностей и выполняет тот же процесс, что и кодировщик. Он способен уделять должное внимание и понимать выходные данные кодировщика и свои собственные прошлые токены на каждом этапе, обеспечивая точную генерацию, учитывая как входной контекст, так и ранее сгенерированный выходной сигнал.



Архитектура модели трансформатора (изображение : Vaswani et al., 2017).

Рассмотрим это предложение: «Я не могу найти ключ от квартиры». Слово « ключ» может иметь два значения – либо металлический предмет, либо источник воды. Вот где трансформеры сияют. Они могут быстро сфокусироваться на слове « «ключ», чтобы устранить неоднозначность слова «ключ», сравнивая слово «ключ» с каждым другим словом в предложении и присваивая оценки внимания. Эти оценки

определяют влияние каждого слова на следующее представление слова «ключ». В этом случае слово «квартира» получает более высокий балл, эффективно проясняющий предполагаемый смысл.
Чтобы работать так хорошо, Трансформеры полагаются на миллионы обучаемых параметров, требуют больших массивов текстов и сложных стратегий обучения. Одним из примечательных подходов обучения, используемых с Трансформерами, является *моделирование языка в масках (MLM)* . Во время обучения определенные токены во входной последовательности случайным образом маскируются, и цель модели — точно предсказать эти замаскированные токены. Эта стратегия побуждает модель улавливать контекстуальные связи между словами, поскольку для точных прогнозов она должна полагаться на окружающие слова. Этот подход, популяризированный моделью BERT, сыграл важную роль в достижении самых современных результатов в различных задачах НЛП.
Альтернативой MLM для Transformers является *авторегрессионное моделирование* . В этом методе модель обучается генерировать по одному слову за раз, при этом учитывая ранее сгенерированные слова. Модели авторегрессии, такие как GPT (генеративный предварительно обученный преобразователь), следуют этой методологии и превосходно справляются с задачами, целью которых является однонаправленное предсказание следующего наиболее подходящего слова, например, генерация произвольного текста, ответы на вопросы и завершение текста.
Кроме того, чтобы компенсировать потребность в обширных текстовых ресурсах, преобразователи превосходно справляются с распараллеливанием, что означает, что они могут обрабатывать данные во время обучения быстрее, чем традиционные последовательные подходы, такие как модули RNN(рекурентная нейронная сеть) или LSTM(сети долгой-кроткосрочной памяти). Эти эффективные вычисления сокращают время обучения и привели к появлению революционных приложений в области обработки естественного языка, машинного перевода и многого другого.
Основная модель Transformer, разработанная Google в 2018 году и оказавшая существенное влияние, — это BERT (представления двунаправленного кодировщика от Transformers). BERT опирался на обучение MLM и представил концепцию двунаправленного контекста, то есть при прогнозировании замаскированного токена он учитывает как левый, так и правый контекст слова. Этот двунаправленный подход значительно улучшил понимание моделью значений слов и контекстуальных нюансов, установив новые стандарты понимания естественного языка и широкого спектра последующих задач НЛП.

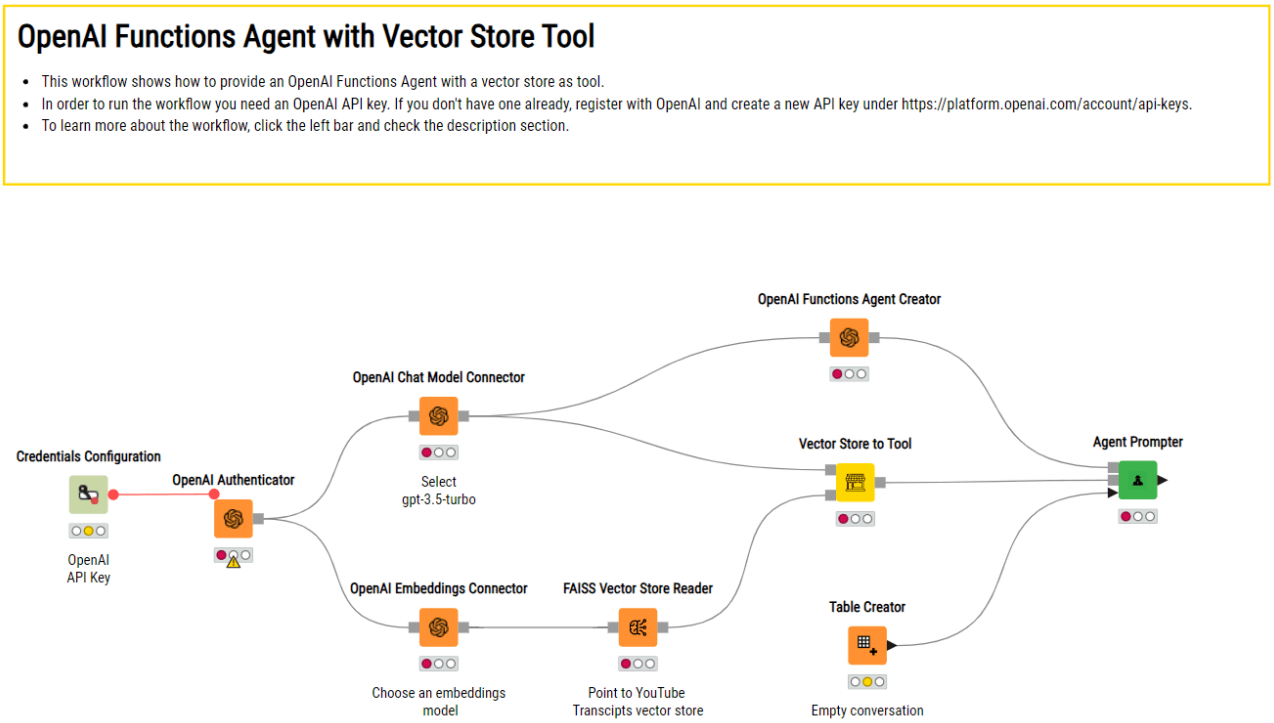
Вслед за Трансформерами, которые представили мощные механизмы самообслуживания, растущий спрос на универсальность приложений и выполнение сложных задач на естественном языке, таких как обобщение документов, редактирование текста или генерация кода, потребовало разработки больших языковых моделей . В этих моделях используются глубокие нейронные сети с миллиардами параметров, позволяющие преуспеть в таких задачах и удовлетворить растущие требования индустрии анализа данных.

6. Большие языковые модели (LLM)

Большие языковые модели (LLM) — это революционная категория многоцелевых и мультимодальных (принимающих входные изображения, аудио и текст) глубоких нейронных сетей, которые в последние годы привлекли значительное внимание. Прилагательное «большой» связано с их огромным размером, поскольку они охватывают миллиарды обучаемых параметров. Некоторые из наиболее известных примеров включают ChatGTP от OpenAI, Bard от Google или LLaMa от Meta.

Что отличает LLM, так это их беспрецедентная способность и гибкость обрабатывать и генерировать текст, похожий на человеческий. Они преуспевают в понимании естественного языка и выполнении задач, начиная от завершения и перевода текста и заканчивая ответами на вопросы и обобщением содержания. Ключ к их успеху заключается в обширной подготовке к работе с огромными текстовыми корпусами, что позволяет им получить глубокое понимание языковых нюансов, контекста и семантики.

Эти модели используют глубокую нейронную архитектуру с несколькими уровнями механизмов самообслуживания, что позволяет им взвешивать важность различных слов и фраз в данном контексте. Эта динамическая адаптируемость делает их исключительно опытными в обработке входных данных различных типов, понимании сложных языковых структур и генерации результатов на основе заданных человеком подсказок.



Пример рабочего процесса KNIME по созданию ИИ-помощника, который использует ChatGPT OpenAI и векторное хранилище с пользовательскими документами для ответов на вопросы, специфичные для предметной области.

LLM проложили путь для множества приложений в различных отраслях: от здравоохранения и финансов до развлечений и обслуживания клиентов. Они даже открыли новые горизонты в творческом писательстве и рассказывании историй.

Однако их огромный размер, ресурсоемкие процессы обучения и потенциальные нарушения авторских прав на создаваемый контент также вызывают обеспокоенность по поводу этического использования, воздействия на окружающую среду и доступности. Наконец, хотя LLM все более совершенствуется, они могут содержать некоторые серьезные недостатки, такие как «галлюцинации» неверных фактов, предвзятость, легковерность или убежденность в создании токсичного контента.

Функции активации

Каждая функция активации (или нелинейности) принимает одно число и выполняет над ним определенную фиксированную математическую операцию. Вот некоторые функции активации, которые вы часто встретите на практике:

- **Sigmoid**
- **Tanh**
- **ReLU**
- **Leaky ReLU**

Функции активации мы с вами разберем на следующих уроках.

Сферы применения ИНС

Искусственные нейронные сети (ИНС) представляют собой мощный инструмент в современной информационной технологии, который нашел широкое применение во множестве областей. Давайте рассмотрим некоторые сферы, где ИНС проявили себя наилучшим образом.

1. **Обработка изображений**: ИНС активно применяются в области компьютерного зрения. Например, сверточные нейронные сети (CNN) используются для распознавания объектов, сегментации изображений и автоматического описания фотографий. Примером может служить система, способная автоматически распознавать лица на фотографиях в социальных сетях.

2. **Обработка естественного языка**: Рекуррентные нейронные сети (RNN) и трансформеры нашли свое применение в задачах обработки текста, включая машинный перевод, анализ тональности, генерацию текста и диалоговых системах. Примером может служить Google Translate, который использует нейронные сети для автоматического перевода между разными языками.

3. **Медицина**: ИНС применяются для анализа медицинских изображений, диагностики заболеваний, прогнозирования эпидемий и даже создания новых лекарств. Нейронные сети могут анализировать рентгеновские

снимки, МРТ и компьютерные томограммы, помогая врачам выявлять патологии.

4. **Финансы**: ИНС используются для прогнозирования финансовых рынков, определения рисков и мошенничества, а также управления портфелем инвестиций. Например, нейронные сети могут анализировать большие объемы данных, чтобы предсказать будущие тренды на фондовом рынке.

5. **Транспорт и автономные транспортные средства**: ИНС играют ключевую роль в развитии автономных автомобилей, позволяя им распознавать дорожные знаки, пешеходов и другие транспортные средства. Это снижает риски дорожных происшествий и увеличивает безопасность на дорогах.

6. **Промышленность**: В производстве нейронные сети могут использоваться для мониторинга и управления процессами, предотвращения сбоев оборудования и оптимизации производственных операций. Например, они могут прогнозировать временные интервалы замены оборудования, что помогает в снижении затрат на обслуживание.

Эти примеры лишь небольшая часть сфер, где искусственные нейронные сети демонстрируют свой потенциал. Их способность адаптироваться к разнообразным задачам и обрабатывать большие объемы данных делает их незаменимым инструментом в современном мире информационных технологий.

Искусственные нейронные сети и мозг

Искусственные нейронные сети работают не так, как наш мозг, ИНС — это простое грубое сравнение, связи между биологическими сетями гораздо сложнее, чем те, которые реализуются с помощью **Искусственных нейронных сетей** Помните, что наш мозг гораздо сложнее, и нам нужно многому у него научиться. Есть много вещей, которые мы не знаем о нашем мозге, и это также затрудняет понимание того, как нам следует моделировать искусственный мозг, чтобы рассуждать на человеческом уровне. Всякий раз, когда мы тренируем нейронную сеть, мы хотим, чтобы наша модель выучила оптимальные веса (w), которые лучше всего предсказывают желаемый результат (y) с учетом входных сигналов или информации (x).

Заключение

В заключение нашей обзорной лекции о искусственных нейронных сетях (ИНС), мы можем подытожить, что ИНС представляют собой удивительное достижение в мире информационных технологий, которое оказало значительное влияние на множество сфер жизни. Эти вычислительные системы, вдохновленные биологическими нейронами, способны обучаться, адаптироваться и решать сложные задачи, которые ранее казались невыполнимыми.

Мы рассмотрели разнообразные сферы применения ИНС, начиная от обработки изображений и обработки текста до медицины, финансов, автономных транспортных средств и промышленности. Их способность анализа больших объемов данных, автоматизации процессов и принятия решений на основе данных делает их бесценным инструментом для современного общества.

Тем не менее, важно помнить, что ИНС - это несомненно мощный инструмент, но его применение также сопряжено с вызовами, включая этические и безопасностные вопросы, а также необходимость постоянного обучения и обновления моделей. В развитии этой технологии неотъемлемо участие специалистов, обеспечивающих ее эффективное и ответственное использование.

ИНС, продолжают изменять мир в лучшую сторону. С каждым новым днем открываются новые возможности для улучшения жизни и решения сложных задач. В будущем, развитие и применение искусственных нейронных сетей будет оставаться ключевым фактором в технологическом прогрессе и улучшении качества жизни людей.