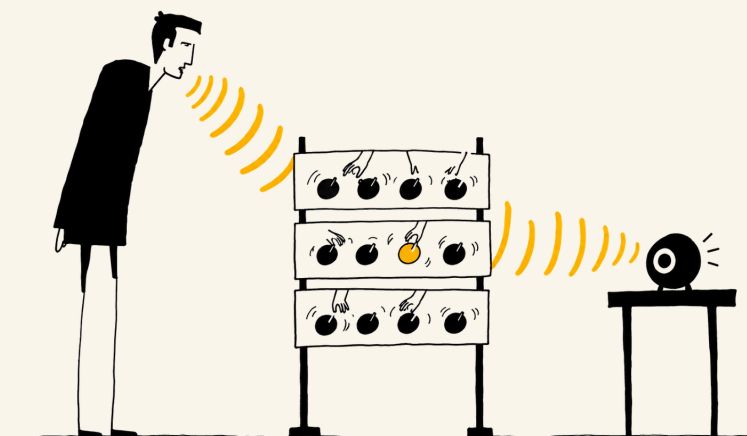


Погружение в глубокое обучение



ton Zhang, Zachary C. Lipton, Mu Li, Alexander J. Smola · 20

Иллюстрированные аудиолекции

Lecturized

Погружение в глубокое обучение

Иллюстрированные аудиолекции

Aston Zhang, Zachary C. Lipton, Mu Li, Alexander J. Smola · 2023

Пересказано в иллюстрированных лекциях Lecturized
lecturized.com

По оригиналу: Dive into Deep Learning (Cambridge University Press, 2023), d2l.ai

Содержание

Программы, которые учатся на примерах

Из чего состоит любая обучающаяся система

Как оценить модель и какой экзамен она должна сдать

Сколько или какой?

Много меток, ранжирование и последовательности

Обучение без начальника и меняющийся мир

Обучение действием и с чего всё началось

Что изменилось: данные, чипы и инструменты

Где это уже работает и что делает его глубоким

Программы, которые учатся на примерах

Поднесите телефон поближе ко рту и скажите слова, которые будят его помощника: «Привет, Siri» или «Окей, Google». Потом скажите что-то похожее, но неверное, например «Привет, Сирия» или «Окей, Гугл-мугл», и посмотрите, что будет. Чаще всего телефон просыпается на настоящие слова и продолжает спать на похожие, даже на шумной кухне и даже когда говорит кто-то другой. За этим маленьким фокусом скрывается вопрос, ответу на который посвящена вся эта книга: как вообще кто-то написал такую программу?

ПРАВИЛА



Магазин работает по правилам

Почти всю историю вычислительной техники программа была жёстким списком правил, написанных людьми. Возьмите интернет-магазин: когда покупатель кладёт товар в корзину, программа записывает строку, связывающую этого покупателя с этим товаром, а разработчики продумывают каждый странный случай, например когда кто-то нажимает «купить» при пустой корзине. Такую программу можно спланировать у доски и уверенно запустить ещё до того, как придёт первый настоящий покупатель. Это замечательное достижение людей, и когда такие правила срабатывают каждый раз, авторы прямо говорят, что машинное обучение здесь вообще не нужно.

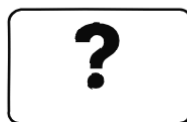
А теперь попробуйте написать правила для слова пробуждения. Микрофон записывает примерно сорок четыре тысячи чисел в секунду, и каждое из них — высота звуковой волны в этот миг. Какое правило превратит секунду-другую таких чисел в уверенное «да» или «нет»? Этого никто не знает; авторы признаются, что и сами не смогли бы его написать. Та же стена стоит перед многими задачами: предсказать завтрашнюю погоду, ответить на вопрос, заданный своими словами, найти всех людей на фотографии или предложить товар, который вам понравится, но который вы сами никогда бы не нашли. Иногда правильный ответ всё время меняется вместе с миром, а иногда путь от пикселей или звука к смыслу слишком запутан, чтобы кто-то мог его расписать, хотя наш собственный мозг проходит его без усилий. Спам-фильтр в вашей почте сталкивается с обеими трудностями сразу, потому что спамеры меняют свои уловки каждую неделю.

ПРИВЕТ, SIRI



44 000 / с

ПРАВИЛО

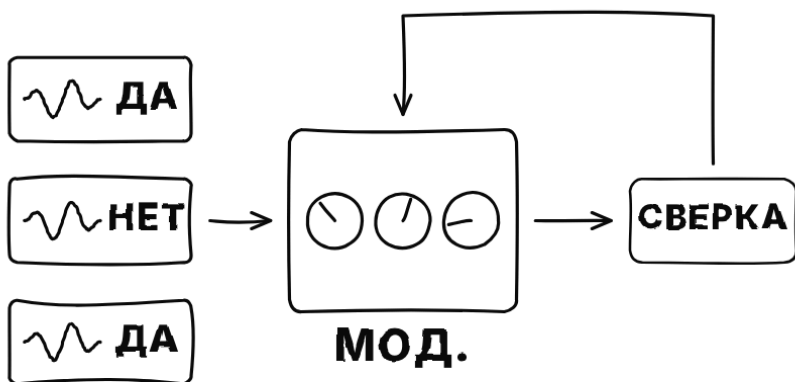


ДА

НЕТ

Для голоса правила нет

Вот выход, и это главная мысль главы. Вы не можете написать правило, по которому узнаётся «Привет, Siri», но сами вы его узнаете. Поэтому люди собирают огромный набор записей и помечают каждую: в этой есть слово пробуждения, в этой нет. Затем вместо жёсткой программы они строят гибкую, поведение которой зависит от огромного числа настраиваемых чисел, называемых параметрами; их можно представить себе как ручки. Если ручки выставлены наугад, программа говорит «да» и «нет» как придётся. Затем обучающий алгоритм понемногу крутит ручки так, чтобы ответы программы на помеченных записях становились лучше, и продолжает, пока программа не станет хорошей. Когда ручки закреплены, программу называют моделью.



Крутим ручки

Авторы называют это программированием с помощью данных. Вы не объясняете машине, как услышать слово пробуждения; вы показываете ей много примеров и даёте ручкам найти положение, при котором всё работает. Модель того же рода, с другим положением ручек, можно так же хорошо научить другому слову, но для совсем другой работы, например подписей к фотографиям или перевода с английского на китайский, может понадобиться модель совсем другого рода. Голосовой помощник в вашем телефоне, предсказанное время в пути в приложении с картами и лента, угадывающая, какое видео вы посмотрите следующим, — всё это сделано так, из примеров, а не из правил, написанных вручную.

Вот что книга называет машинным обучением: программы, которые работают лучше с опытом, обычно в виде данных. Правила интернет-магазина сами по себе не становятся лучше, сколько бы покупателей через него ни прошло; обучающаяся программа становится. Глубокое обучение, предмет этой книги, — одно семейство методов внутри машинного обучения, и сегодня оно движет большей частью успехов в распознавании изображений и речи, в понимании языка, в медицине и в чтении геномов. Чат-бот, которому вы печатаете вопросы, — это модель глубокого обучения, ручки которой настроили, показав ей огромное количество текста. В этой части мы не говорили о том, как именно крутят ручки и как измеряют качество модели; это дальше.

Опыт

Попросите друга тихо произнести слово пробуждения вашего телефона с другого конца комнаты, а потом похожее слово тем же голосом. Посчитайте, как часто телефон просыпается на каждое. Вы увидите, что обычно он просыпается на настоящее слово и редко на похожее, хотя и не безупречно: обученная модель права в большинстве случаев, а не всегда, потому что она училась на примерах, а не по правилу.

Запомним

Некоторые задачи можно решить правилами, написанными людьми, и тогда машинное обучение не нужно. Другие задачи, например услышать слово пробуждения или узнать лицо, нам даются легко, но в правила их не уложить. Для них мы собираем много помеченных примеров и даём обучающему алгоритму настраивать ручки гибкой программы, пока она не начнёт хорошо справляться. Настроенную программу называют моделью, а такой способ работы — программированием с помощью данных. Глубокое обучение — одно мощное семейство таких методов, и именно ему учит остальная часть книги.

Вопросы

1. Почему слово пробуждения вроде «Привет, Siri» трудно распознать правилами, написанными вручную?
 - a. Микрофон собирает слишком мало чисел, чтобы различать слова
 - b. Никто не может сказать, какое правило превращает десятки тысяч звуковых измерений в «да» или «нет», хотя люди слышат это слово легко
 - c. Телефоны недостаточно быстры, чтобы следовать длинному списку правил
 - d. Слово пробуждения меняется при каждом обновлении телефона

2. Когда, по словам книги, машинное обучение вообще не нужно?
 - a. Когда задача связана с большой базой данных покупателей
 - b. Когда программа должна работать на телефоне
 - c. Когда можно написать правила, которые срабатывают каждый раз, как для корзины интернет-магазина
 - d. Когда данные записаны числами, а не словами
3. Обучающий алгоритм только что начал работу с новой программой, ручки которой выставлены наугад. Что делает программа?
 - a. Отвечает «да» и «нет» примерно наугад
 - b. Уже распознаёт слово пробуждения, но медленно
 - c. Отказывается отвечать, пока не увидит все данные
 - d. Одинаково хорошо распознаёт любое английское слово
4. Чем обучающаяся программа со временем отличается от программы интернет-магазина?
 - a. Программа магазина быстрее, потому что её правила жёсткие
 - b. Обучающаяся программа становится лучше с опытом, а правила магазина остаются прежними, пока их не переписут разработчики
 - c. Обучающейся программе после запуска данные больше не нужны
 - d. Программа магазина учится на кликах каждого покупателя

Ответы

1. **b)** Телефон записывает около сорока четырёх тысяч чисел в секунду, и никто не знает правила, которое надёжно переводило бы их в «да» или «нет», хотя наши уши делают это без усилий.
2. **c)** Если правила, написанные людьми, работают во всех случаях, как это возможно для корзины магазина, обучению нечего добавить.
3. **a)** Со случайными ручками программа бесполезна; хорошей её делает только поворот ручек так, чтобы она лучше справлялась с помеченными примерами.
4. **b)** Машинное обучение — это работа, которая становится лучше с опытом; правила магазина сами по себе не улучшаются, сколько бы покупателей ни прошло.

Задачи

1. Книга перечисляет задачи, для которых даже лучшим программистам трудно написать правила, например предсказать завтрашнюю погоду или предложить товары, которые вам понравятся. Выберите две из них и объясните для каждой, какая из двух причин здесь действует: правильный ответ всё время меняется вместе с миром или путь от входных данных к смыслу слишком запутан, чтобы его расписать.

Подсказка: Спросите себя про каждую задачу: осталось бы идеальное правило, написанное сегодня, верным через год? И могли бы вы шаг за шагом описать, как делаете это в голове?

Ответ: Подбор товаров относится к первой причине: вкусы людей и ассортимент меняются, поэтому любое жёсткое правило устаревает и решение должно подстраиваться. Найти всех людей на фотографии — ко второй: мы делаем это без усилий, но шаги от пикселей к «это

человек» никто не может записать. В прогнозе погоды есть немного и того, и другого.

2. Книга описывает, как можно «запрограммировать» распознаватель кошек, не написав ни одного правила об усах или ушах. Объясните своими словами, что вы дали бы системе и что она научилась бы выдавать.

Подсказка: Вспомните рецепт со словом пробуждения: на входе помеченные примеры, ручки настраиваются, на выходе модель.

Ответ: Вы даёте ей много картинок с пометками «кошка» или «собака». Обучающий алгоритм настраивает ручки, пока она не станет выдавать большое положительное число для кошки, большое отрицательное для собаки и что-то около нуля, когда не уверена. Поведение берётся из примеров — это то, что книга называет программированием с помощью данных.

3. Модель обучили просыпаться на «Alexa». Теперь создатели хотят, чтобы она просыпалась на «Apricot», а отдельно хотят программу, которая пишет подписи к фотографиям. Для какой из двух задач, скорее всего, подойдёт модель того же рода и почему?

Подсказка: Сравните, что поступает на вход и что выходит в каждой работе.

Ответ: Для «Apricot»: это работа того же рода — на входе звук, на выходе «да» или «нет», поэтому подойдёт то же семейство моделей с другим положением ручек. Подписи к фотографиям — это изображения на входе и предложения на выходе, принципиально другая работа, для которой может понадобиться модель совсем другого рода.

Из чего состоит любая обучающаяся система

Возьмите любую фотографию в телефоне и увеличивайте её, пока она не распадётся на маленькие квадратики. Каждый квадратик, пиксель, — это всего лишь три числа: насколько ярок в этой точке красный, зелёный и синий свет. Поэтому маленькая фотография шириной двести пикселей и высотой двести — это список из ста двадцати тысяч чисел. Именно это на самом деле видит обучающаяся программа, и это первая из четырёх частей, которые есть у любой системы машинного обучения, какой бы ни была её работа.

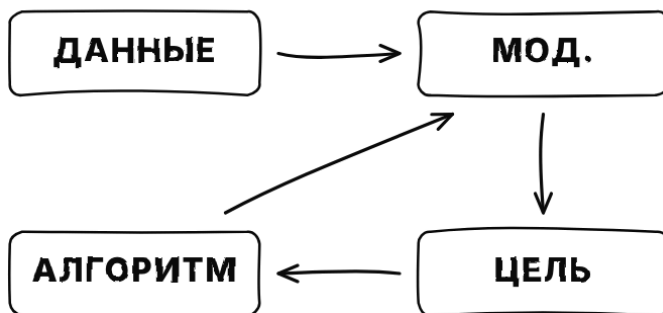


Фото — это числа

Вот эти четыре части: данные, на которых она учится; модель, которая преобразует данные; целевая функция, которая оценивает, насколько хорошо модель справляется; и алгоритм, который подстраивает ручки модели, чтобы улучшить эту оценку. В примере со словом пробуждения из первой части были все четыре, и он относился к самому распространённому виду задач — обучению с учителем, где к каждому примеру приложен ответ, который программа должна научиться давать.

Данные — это собрание примеров, каждый из которых превращён в числа. Числа, описывающие пример, — это его признаки, а в обучении с учителем ответ, который нужно предсказать, — это его метка. В больнице признаками пациента

могут быть возраст, пульс, лекарства и недавние операции, а меткой — выжил ли этот пациент в следующие тридцать дней. Когда у каждого примера одинаковое число признаков, данные аккуратны; но фотографии из интернета бывают любых размеров, а с текстом ещё хуже: один отзыв говорит только «отстой!», а другой тянется на несколько страниц. Умение спокойно работать с данными разной длины — одно из больших преимуществ глубокого обучения перед старыми методами. Письма, которые читает спам-фильтр, — как раз такие данные: одни в одну строку, другие — длинные послания.

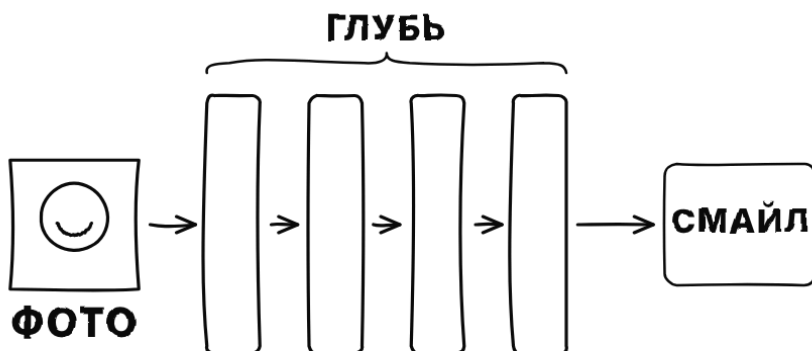


Признаки и метка

Ещё две вещи о данных очень важны. Во-первых, больше — лучше: на большем числе примеров можно обучить более

мощные модели, и переход от малых данных к большим — одна из главных причин успеха глубокого обучения. Во-вторых, данные должны быть правильными, потому что из мусора на входе выходит мусор на выходе. Если какой-то группы людей нет среди примеров, как у распознавателя рака кожи, который никогда не видел тёмной кожи, система их подводит; а если прошлые решения были несправедливы, как старые решения о найме, по которым отбирают новые заявки, модель может тихо выучить и повторить эту несправедливость, хотя никто этого не хотел. Лента рекомендаций, обученная только на том, что смотрела одна группа пользователей, будет плохо служить всем остальным.

Модель — это механизм, который принимает данные одного рода и выдаёт предсказания, возможно, другого рода: на входе фотографии, на выходе оценка улыбки; на входе показания датчиков, на выходе предупреждение о чём-то необычном. Простые модели годятся для простых задач, но задачи этой книги им не по силам. Глубокое обучение выделяется тем, что его модели — это цепочки из многих преобразований, идущих одно за другим, каждое передаёт свой результат следующему, и эта глубина дала глубокому обучению его имя. Как работают целевая функция и обучающий алгоритм, мы оставим для следующей части.



Цепочка преобразований

Запомним

У любой системы машинного обучения четыре части: данные, модель, целевая функция и алгоритм, улучшающий модель. Данные — это примеры, превращённые в числа и описанные признаками, а в обучении с учителем у каждого есть метка. Больше данных помогает, но неверные или однобокие данные ведут к провалу и могут повторить старую несправедливость. Глубокая модель — это длинная цепочка преобразований данных, и именно эта цепочка делает её глубокой.

Вопросы

1. Команда строит для больницы модель, которая предсказывает, выживет ли пациент в следующие тридцать дней. Что из этого — метка?
 - a. Возраст и пульс пациента
 - b. Выжил ли пациент в эти тридцать дней
 - c. Список лекарств пациента
 - d. Число пациентов в записях
2. Почему текст доставляет обучающимся системам больше хлопот, чем снимки с одного стандартного микроскопа?
 - a. В тексте вообще нет чисел
 - b. Отзывы и сообщения очень сильно различаются по длине, поэтому не укладываются в фиксированное число признаков
 - c. Текст всегда полон ошибок
 - d. Снимкам с микроскопа не нужны признаки
3. Модель для отбора резюме обучают на прошлых решениях компании о найме, которые отдавали предпочтение одной группе. В чём опасность?
 - a. Модель будет слишком медленной
 - b. Модель откажется кого-либо ранжировать
 - c. Модель может выучить и повторить старую несправедливость, даже если никто этого не хочет
 - d. Модели понадобится меньше примеров, чем обычно

4. Что делает модель глубокого обучения «глубокой»?
- a. Она — длинная цепочка преобразований, каждое из которых передаёт свой результат следующему
 - b. Её обучают на очень больших объёмах данных
 - c. У неё самая точная целевая функция
 - d. Она работает только с фотографиями

Ответы

1. **b)** Метка — это ответ, который модель учится предсказывать; возраст, пульс и лекарства — это признаки, которые подаются на вход.
2. **b)** У аккуратных данных одинаковое число признаков у каждого примера; один отзыв говорит «отстой!», а другой тянется на страницы, и сила глубокого обучения — в умении работать с такой разной длиной.
3. **c)** Мусор на входе — мусор на выходе: если примеры отражают прошлые предубеждения, модель тихо их усваивает.
4. **a)** Глубина — это число последовательных преобразований, соединённых в цепочку; отсюда и название.

Разобранные примеры

Фотография как числа Цветная фотография имеет ширину 200 пикселей и высоту 200 пикселей. Каждый пиксель хранится как яркость красного, зелёного и синего света. Сколько чисел получает обучающаяся программа для одной этой фотографии?

Считаем пиксели: 200 по ширине умножить на 200 по высоте — 40 000 пикселей.

Каждый пиксель несёт три числа, по одному на каждый цвет света.

Значит, фотография — это 40 000 умножить на 3 чисел.

Ответ: 120 000 чисел

Задачи

1. Цветная фотография шириной 300 пикселей и высотой 200 хранится так же — как яркость красного, зелёного и синего для каждого пикселя. Сколько признаков она даёт модели?

Подсказка: Сначала посчитайте пиксели, потом вспомните, сколько чисел несёт каждый пиксель.

Ответ: 180 000. Пикселей 300 на 200, то есть 60 000, и каждый даёт три числа.

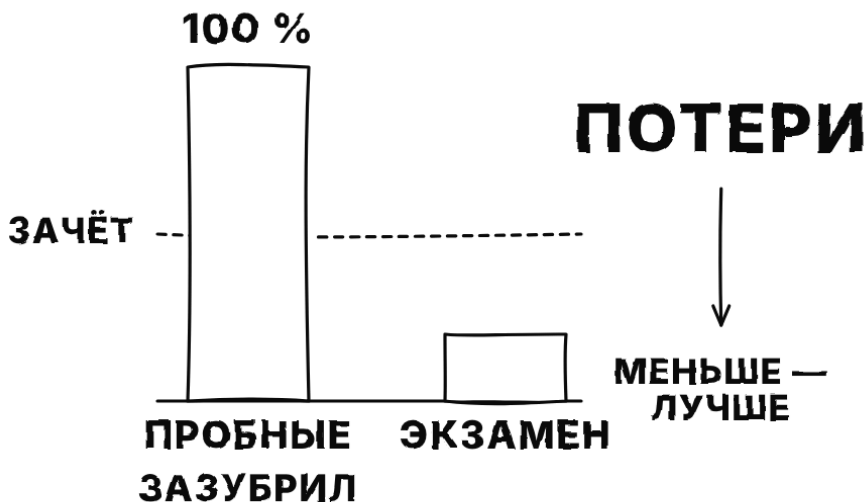
2. Распознаватель рака кожи хорошо работает на испытаниях, но ошибается на многих пациентах, когда его начинают использовать в клинике. Опираясь на мысль о «правильных данных», назовите вероятную причину и скажите, что нужно изменить.

Подсказка: Спросите себя, кто был и кого не было среди примеров, на которых учился распознаватель.

Ответ: В обучающих примерах, вероятно, не было какой-то группы пациентов, например людей с тёмной кожей, поэтому распознаватель так и не узнал, как выглядит их кожа. Больше данных того же рода не помогут; в примерах должны быть те люди, на которых систему будут применять.

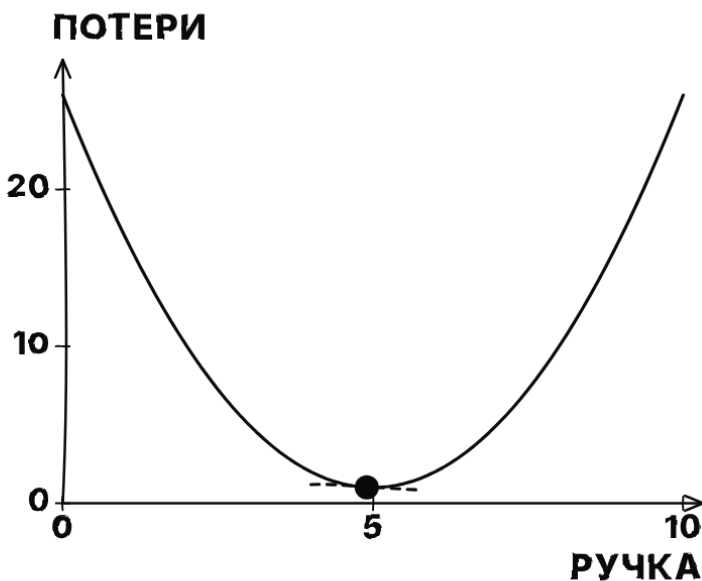
Как оценить модель и какой экзамен она должна сдать

Представьте студента, который готовится к выпускному экзамену по стопке пробных билетов. Через неделю он решает каждый пробный билет на высший балл. Потом приходит настоящий экзамен с вопросами, которых он никогда не видел, и он проваливается. Он не выучил предмет; он запомнил ответы на пробные билеты. Обучающиеся машины попадают ровно в ту же ловушку, и то, как книга от неё защищается, — суть этой части.



Пробный билет — не экзамен

Прежде всего, машина не может стать лучше, пока кто-то не скажет, что значит «лучше». Поэтому у каждой обучающейся системы есть целевая функция — одно число, которое оценивает, насколько плохо справляется модель, и по договорённости чем оно меньше, тем лучше; поэтому её ещё называют функцией потерь. Когда модель предсказывает число, обычная функция потерь наказывает за величину каждого промаха, причём большие промахи весят гораздо больше маленьких. Когда она предсказывает категорию, естественная оценка — доля примеров, на которых она ошибается, но такую оценку неудобно улучшать напрямую, поэтому на практике берут более гладкую замену. Для спам-фильтра функцией потерь может быть просто число писем, которые он положил не в ту папку.



Спуск с холма

Дальше идёт обучающий алгоритм — та часть, что крутит ручки. Метод, на котором держится почти всё глубокое обучение, называется градиентным спуском. На каждом шаге он спрашивает про каждую ручку, поднимет ли крошечный её поворот функцию потерь или опустит, а затем немного сдвигает каждую ручку в ту сторону, где потери меньше. Шаг за шагом, как будто спускаешься с холма в тумане, нащупывая склон ногами, модель становится лучше на своих примерах.

А теперь ловушка. Успех на примерах, на которых модель обучали, не доказывает, что она справится с новыми. Поэтому данные делят на две части: обучающую выборку, на которой настраивают ручки, и тестовую выборку, которую откладывают и показывают модели только в конце, как настоящий экзамен. Модель, которая блестяще работает на обучающей выборке и проваливается на тестовой, переобучена: она запомнила, а не научилась.



Обучающая и тестовая выборки

Это обучение с учителем — тот вид, на котором держится большинство успехов машинного обучения в промышленности. Соберите примеры, получите для каждого правильный ответ — либо из уже существующих записей, либо от людей, которые размечают их вручную, — дайте их обучающему алгоритму, и на выходе получится модель. Затем подайте модели входные данные, которых она никогда не видела, и используйте её ответы как предсказания: рак или нет по снимку, французское предложение по английскому, цена акции по отчёту компании. Когда вы помечаете письмо как спам, вы добавляете ещё один размеченный пример в чью-то обучающую выборку. Как разные виды ответов, числа или категории, меняют задачу, — это для следующей части.

Запомним

Модель может становиться лучше только относительно оценки — функции потерь, которая говорит, насколько плохо она справляется; чем меньше, тем лучше.

Градиентный спуск улучшает модель, слегка сдвигая каждую ручку в ту сторону, где потери меньше. Хорошие оценки на обучающих примерах мало что доказывают, поэтому отдельную тестовую выборку держат в стороне, как настоящий экзамен. Модель, которая хороша на обучении и плоха на новых данных, переобучена. Обучение с учителем — это обучение на примерах с известными ответами, а затем предсказание ответов для новых входных данных.

Вопросы

1. Модель почти безупречна на обучающей выборке, но плохо работает на тестовой. Что случилось?
 - a. Тестовая выборка была слишком мала, чтобы что-то значить
 - b. Модель переобучилась: она запомнила свои примеры, а не научилась
 - c. Функцию потерь определили так, что больше значит лучше
 - d. Градиентный спуск сдвинул ручки не в ту сторону

2. Зачем тестовую выборку не показывают модели до самого конца?
- a. Чтобы она показала, как модель справляется с данными, которых никогда не видела
 - b. Потому что у тестовой выборки нет меток
 - c. Потому что градиентный спуск не умеет работать с большим объёмом данных
 - d. Чтобы обучение закончилось быстрее
3. Что делает градиентный спуск на каждом шаге?
- a. Пробует случайные положения всех ручек и оставляет лучшее
 - b. Добавляет примеры в обучающую выборку
 - c. Проверяет, как небольшой поворот каждой ручки изменит потери, и сдвигает ручки в ту сторону, где потери меньше
 - d. Убирает примеры, на которых модель ошибается
4. Модель предсказывает цены домов. Что при обычной функции потерь для чисел наказывается сильнее?
- a. Десять маленьких промахов той же общей величины
 - b. Одинаково, ведь важна только сумма
 - c. Ни то, ни другое, потому что цены — это категории
 - d. Один большой промах

Ответы

1. **b)** Блестяще работать на обучающих примерах и проваливаться на новых — это и есть переобучение, как у студента, который зазубрил пробные билеты.

2. а) Тестовая выборка играет роль настоящего экзамена; если бы модель на ней училась, уже нельзя было бы отличить обучение от зубрёжки.

3. с) Как при спуске с холма в тумане, он нащупывает склон для каждой ручки и делает маленький шаг вниз.

4. d) Обычная функция потерь для чисел заставляет большие промахи весить гораздо больше маленьких.

Разобранные примеры

Доля ошибок спам-фильтра Спам-фильтр проверяют на 500 письмах, которых он не видел при обучении, и 15 из них он кладёт не в ту папку. Какова его доля ошибок на тестовой выборке?

Доля ошибок — это доля примеров, на которых предсказание расходится с правильным ответом.

Здесь ошибочны 15 писем из 500.

15, делённое на 500, — это 0,03.

Ответ: 0,03, то есть 3 процента

Задачи

1. При квадратичной ошибке предсказание, промахнувшееся на 2, стоит 2 умножить на 2, то есть 4. Сколько стоит промах на 6 и во сколько раз это больше, чем промах на 2?

Подсказка: Возведите в квадрат разность между предсказанием и верным значением.

Ответ: 36, то есть в 9 раз больше, чем 4: если промах втрое больше, цена вырастает в девять раз, поэтому большие промахи весят гораздо больше.

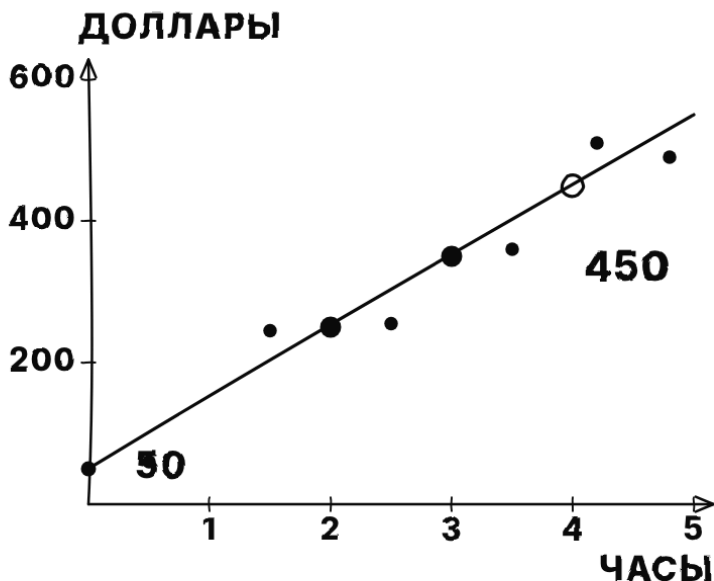
2. Две модели обучили распознавать слова пробуждения. Модель А верна в 99 процентах случаев на обучающей выборке и в 70 процентах на тестовой; модель В — в 92 процентах на обучении и в 91 на тесте. Какую вы поставили бы в телефон и почему?

Подсказка: Какое число говорит о том, как модель справится с голосами, которых она никогда не слышала?

Ответ: Модель В. Тестовая выборка представляет новые данные, и на ней В гораздо лучше; высокая оценка А на обучении при намного более низкой на тесте — признак переобучения.

Сколько или какой?

Представьте: сантехник три часа прочищает вам трубы и присылает счёт на триста пятьдесят долларов. Ваш друг вызывает того же сантехника, тот работает два часа, и счёт — двести пятьдесят. Сосед спрашивает, сколько будет стоить работа на четыре часа. Большинство людей считают это без ручки: лишний час добавил сто долларов, значит, сантехник берёт сто в час плюс пятьдесят долларов за вызов, и четыре часа должны стоить четыреста пятьдесят. Если вы за этим проследили, говорит книга, вы уже понимаете идею, лежащую в основе простейшего вида обучения с учителем.



Ценовая прямая сантехника

Этот вид называется регрессией, и задачу делает регрессией ответ, которого она требует: число, которое может принимать любое значение. Примеры обычно — это строки таблицы, и признаки каждой строки образуют вектор признаков. Для домов в строке может быть площадь, число спален, число ванных и сколько минут пешком до центра города, а метка — цена, за которую дом продали. Хорошее правило: любой вопрос «сколько?» — это регрессия: сколько часов займёт операция, сколько дождя выпадет за следующие шесть часов, сколько дней пациент пробудет в больнице. Звёздная оценка, которую стриминговый сервис угадывает за вас для фильма, — тоже регрессия; книга вспоминает, что за хорошее предсказание таких

оценок в 2009 году можно было выиграть миллионный приз Netflix.

С сантехником два примера определили цену точно. Настоящие данные никогда не бывают такими чистыми: на ответы влияют вещи, которых признаки не учитывают. Поэтому от модели не требуют попасть в каждый пример, а только подойти как можно ближе в целом, и обычная мера близости — квадрат промаха из прошлой части.

Другое большое семейство — классификация, и её вопрос — «какой?». Приложение банка фотографирует чек и должно решить для каждого рукописного знака, какой это символ; распознаватель цифр выбирает из десяти классов, от нуля до девяти; в простейшем случае классов всего два, например кошка или собака. Твёрдые ответы «да» или «нет» оказывается трудно обучать, поэтому классификатор обычно выдаёт вероятности: увидев фото, он может сказать, что на девяносто процентов уверен, что это кошка. Это число говорит ещё и о том, насколько модель не уверена, а это очень важно для того, что будет дальше. Ваш спам-фильтр — классификатор с двумя классами, спам и не спам, и он тоже учитывает свою уверенность, прежде чем переложить письмо.

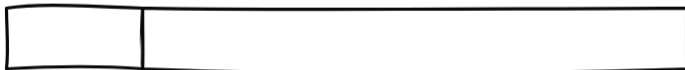


Классификатор даёт шансы

Книга показывает это на грибе, найденном в саду.

Классификатор, обученный на фотографиях, говорит, что с вероятностью двадцать процентов это бледная поганка, самый смертоносный гриб на свете. Значит, он на восемьдесят процентов уверен, что гриб безопасен, — и всё же съесть его было бы глупо. Самый вероятный ответ — не всегда тот, по которому надо действовать: решение должно взвешивать шансы против цены каждого исхода, а малый шанс смерти перевешивает любой ужин. Гриб и правда был бледной поганкой.

ПОГАНКА 20 % БЕЗОПАСНО 80 %



СЪЕСТЬ

ВЫБРОСИТЬ

РИСК: СМЕРТЬ

ЦЕНА: УЖИН



Взвесить шанс и цену

Ошибки неравны и ещё в одном смысле. Когда классы образуют родословное древо, как древо живых существ у Линнея, спутать пуделя со шнауцером — небольшая оплошность, а спутать пуделя с динозавром — нелепость, поэтому модель можно построить так, чтобы она предпочитала близкие промахи. Но какое древо важно, зависит от применения: гремучая змея и безобидный подвязочный уж — близкие родственники, но спутать их может стоить жизни. Предсказания со многими метками и более сложные виды предсказаний — в следующей части.

Опыт

Запишите два счёта сантехника: три часа за триста пятьдесят долларов и два часа за двести пятьдесят. Найдите по разнице цену за час, затем плату за вызов и предскажите работу на пять часов. Должно получиться сто в час, пятьдесят за вызов и пятьсот пятьдесят долларов: вы только что подогнали модель к данным и использовали её для предсказания.

Запомним

Регрессия отвечает на вопрос «сколько?» числом, например ценой или числом часов. Классификация отвечает на вопрос «какой?», выбирая из категорий, например кошка или собака, или цифры от нуля до девяти. Классификаторы обычно выдают вероятности, которые показывают, насколько они уверены. Решение должно взвешивать эти шансы против цены каждого исхода, поэтому гриб, безопасный на восемьдесят процентов, всё равно отправляется в мусор. Одни ошибки хуже других, и какие именно — зависит от того, как будут использовать модель.

Вопросы

1. Какой из этих вопросов — задача регрессии?
 - а. Это письмо — спам или нет?

- b. Какая цифра написана на этом чеке?
 - c. Сколько часов займёт эта операция?
 - d. Этот гриб — бледная поганка?
2. Классификатор на 80 процентов уверен, что гриб безопасен. Почему его всё равно не стоит есть?
- a. Потому что 80 процентов — наименее вероятный ответ
 - b. Потому что решение должно взвешивать шанс в 20 процентов против цены, а эта цена — смерть
 - c. Потому что классификаторы никогда не дают вероятностей
 - d. Потому что классификатор обучали на кошках и собаках
3. Почему классификаторы обычно выдают вероятности, а не твёрдый ответ?
- a. Твёрдые ответы трудно обучать, а вероятность ещё и показывает, насколько модель не уверена
 - b. Вероятностям нужно меньше данных
 - c. Твёрдые ответы нельзя показать на экране
 - d. Вероятности делают все ошибки одинаково дорогими
4. Какая ошибка важнее всего для модели, различающей змей для туристов?
- a. Назвать подвязочного ужа обыкновенным ужом
 - b. Назвать одну безобидную змею именем другой безобидной
 - c. Назвать гремучую змею подвязочным ужом
 - d. Назвать подвязочного ужа гремучей змеей

Ответы

1. **с)** Вопрос «сколько?» требует числа, и это делает его регрессией; остальные выбирают из категорий.
2. **b)** Самый вероятный ответ — не всегда тот, по которому надо действовать; малый шанс смертельного исхода перевешивает пользу от ужина.
3. **а)** Твёрдые ответы «да» или «нет» трудно оптимизировать, а вероятность вроде «девяносто процентов кошка» несёт неуверенность модели, которая нужна для решений.
4. **с)** Эти змеи — близкие родственники, но принять опасную гремучую змею за безобидного ужа может стоить жизни; какие ошибки важны, зависит от применения.

Разобранные примеры

Цены сантехника Мастер работает 3 часа и выставляет счёт на 350 долларов; у другого клиента он работает 2 часа и выставляет 250 долларов. Пусть есть фиксированная плата за вызов плюс цена за час. Найдите обе и предскажите счёт за 4 часа.

Лишний час, от 2 до 3, добавил 350 минус 250, то есть 100 долларов, значит, цена за час — 100 долларов.

Два часа стоят 2 умножить на 100, то есть 200 долларов работы, значит, остаток счёта в 250 долларов, 50 долларов, — это фиксированная плата.

Проверка по другому счёту: 50 плюс 3 умножить на 100 — это 350, как и в счёте.

За 4 часа: 50 плюс 4 умножить на 100.

Ответ: 100 долларов в час плюс 50 долларов за вызов; 450 долларов за 4 часа

Съесть или выбросить Классификатор даёт вероятность 0,2, что гриб — бледная поганка, и 0,8, что он безопасен. Выбросить безопасный гриб стоит 1 (потерянный ужин); выбросить поганку ничего не стоит; съесть поганку стоит жизни, которую не перевесит никакой ужин. Какова ожидаемая цена того, чтобы выбросить гриб, и как следует поступить?

Выбросить: 0,2 умножить на 0 за поганку плюс 0,8 умножить на 1 за потерянный ужин.

Это даёт 0,8.

Съесть: шанс 0,2 на цену без предела, поэтому и ожидаемая цена не имеет предела.

Конечная цена лучше неограниченной, так что гриб надо выбросить.

Ответ: Выбросить стоит в среднем 0,8; съесть — без предела, поэтому гриб выбрасывают.

Задачи

1. Мойщик окон берёт 380 долларов за работу на 4 часа и 140 долларов за работу на 1 час: фиксированная плата плюс цена за час. Найдите цену за час, плату и счёт за 3 часа.

Подсказка: На сколько выросла сумма за три лишних часа?

Ответ: Три лишних часа добавили 240 долларов, значит, 80 долларов в час; тогда 140 минус 80 оставляет плату 60 долларов; 3 часа стоят 60 плюс 240, то есть 300 долларов.

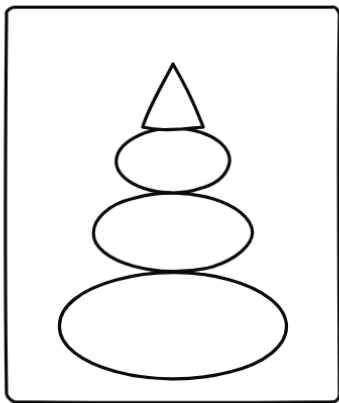
2. Разделите на регрессию и классификацию и объясните почему: цена, за которую продадут квартиру; показывает ли снимок рак; сколько дождя выпадет за следующие шесть часов; какая это буква, написанная от руки.

Подсказка: Спросите себя, ответ — это число или выбор из категорий.

Ответ: Цена квартиры и количество дождя — числа, значит, это регрессия; рак или нет и какая буква — выбор из категорий, значит, это классификация.

Много меток, ранжирование и последовательности

Откройте на телефоне приложение с фотографиями и наберите в поиске «собака». Появятся ваши снимки собаки, но и тот, из парка, где в кадре вместе собака, ребёнок, велосипед и дерево. Эта фотография не может относиться только к одному классу. Телефон пометил её сразу несколькими метками, и этот маленький факт выводит нас за пределы вопроса «какой?» из прошлой части.



ОДИН КЛАСС ✕

ОСЁЛ ✓

ПЁС ✓

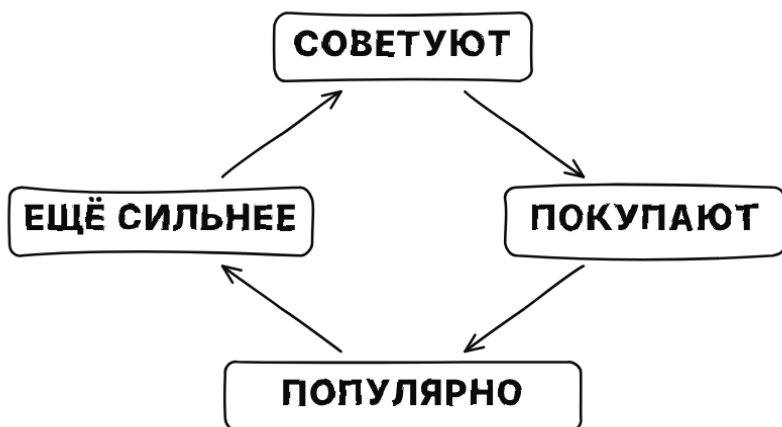
КОТ ✓

ПЕТУХ ✓

Одно фото, много меток

Пример в книге — сказка о бременских музыкантах: осёл, собака, кот и петух, стоящие друг у друга на спинах. Классификатор, вынужденный выбрать одно животное, непременно ошибётся, потому что честный ответ — все четыре. Предсказание классов, которые не исключают друг друга, называется классификацией с несколькими метками, и так работает автоматическая расстановка тегов. Запись в технологическом блоге может нести от пяти до десяти тегов, и теги ходят вместе: записи об облачных вычислениях обычно упоминают какого-нибудь крупного облачного провайдера, записи о машинном обучении — графические процессоры. Крайний случай — Национальная медицинская библиотека США, которая помечает каждую индексируемую медицинскую статью терминами примерно из двадцати восьми тысяч рубрик; у экспертов-людей эта работа занимает до года, поэтому обученная модель может пока ставить временные метки.

Поиск задаёт снова другой вопрос: не подходит ли страница, а в каком порядке показывать подходящие страницы. Первый секретный ингредиент Google, PageRank, давал каждой странице оценку авторитетности, которая, как ни странно, вообще не зависела от запроса; сначала простой фильтр находил подходящие страницы, а затем оценка выстраивала их по порядку. Сегодня поисковики учат оценки, которые зависят от запроса и от того, как ведут себя люди.

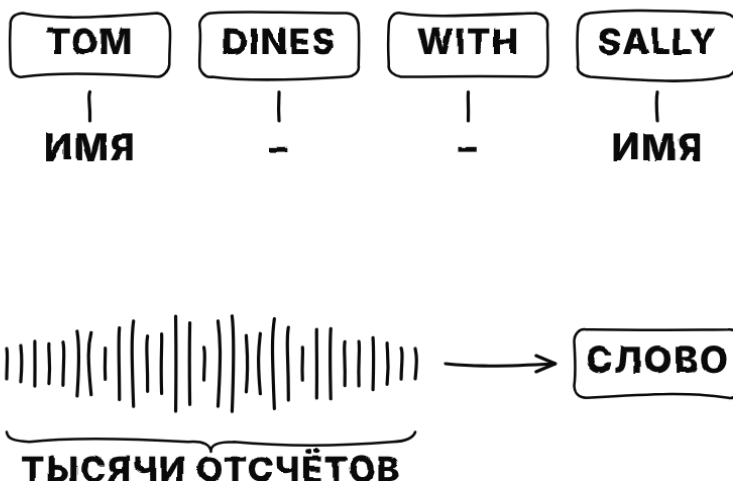


Ранжирование и его петля

Рекомендательные системы — близкие родственники поиска, с одним большим отличием: список составляется лично для вас. Любитель научной фантастики и поклонник старых комедий с Питером Селлерсом должны видеть очень разные страницы. Одни отзывы явные, как звёздные оценки и рецензии; другие неявные, как пропуск песни, который может означать неприязнь, а может — просто неподходящий момент. Лента в видеоприложении, которая будто знает, что вы посмотрите дальше, — рекомендательная система именно такого рода: она угадывает оценку для каждого видео и показывает вам лучшие. Эти системы стоят целое состояние, но у них глубокие изъяны. Люди в основном оценивают то, что любят или ненавидят, поэтому три звезды встречаются на удивление редко, а

собственные советы системы влияют на то, что люди покупают, так что товар, который она продвигает, выглядит популярным и продвигается ещё сильнее — это петля обратной связи, которая до сих пор остаётся открытой задачей исследований.

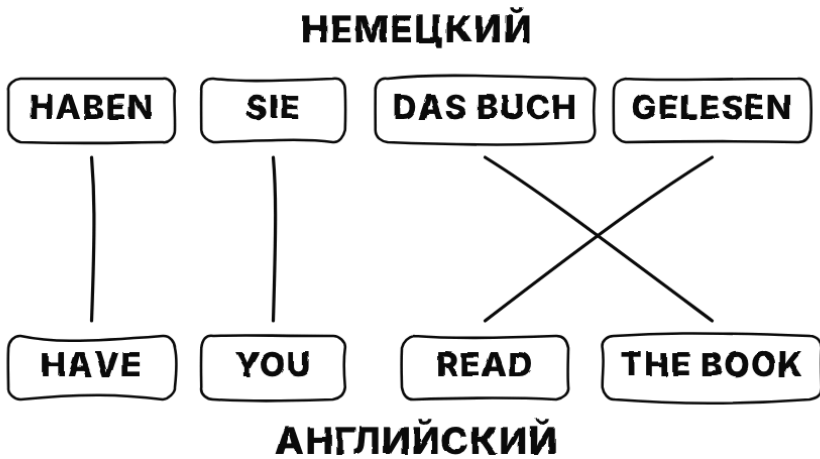
До сих пор каждый пример обрабатывался отдельно и забывался, как только на него ответили. Но видео — это череда кадров, и то, что происходит в одном кадре, гораздо понятнее, когда видишь соседние. За пациентом в реанимации следят час за часом, и модель, предупреждающая об опасности, должна помнить всю историю, а не только последнее показание. Задачи, где вход, выход или оба являются последовательностями разной длины, называются обучением на последовательностях. Чат-бот читает всё ваше сообщение как последовательность слов и отвечает другой последовательностью, так что это обучение на последовательностях от начала до конца.



Как совпадают последовательности

Книга называет несколько видов. При разметке каждое слово получает пометку, например является ли оно частью имени, так что вход и выход совпадают один к одному. При распознавании речи тысячи звуковых измерений образуют одно слово, так что длинный вход становится коротким выходом; синтез речи идёт в обратную сторону — из короткого текста в длинный звук. Когда вы диктуете сообщение голосовому помощнику, он делает первое, а когда читает ответ вслух — второе. Машинный перевод ещё труднее, потому что меняется порядок: в немецком глагол любит стоять в конце, поэтому дословный английский перевод звучал бы как «Have you yourself already this excellent textbook looked at?» — «Вы сами уже этот отличный учебник посмотрели?» Понимать вёрстку веб-страницы и вести разговор,

помня, что было сказано давно и как устроен мир, ещё труднее, и это пока исследуют. Обучение вовсе без меток — в следующей части.



Перевод меняет порядок

Запомним

Классификация с несколькими метками позволяет одному примеру нести сразу несколько меток; так работает автоматическая расстановка тегов. Поиск и рекомендации — это о том, как расставить объекты по порядку, а рекомендации добавляют личный вкус. Рекомендательные системы сталкиваются с перекошенными отзывами и петлями, в которых их собственные советы делают товары более популярными на вид. Обучение на последовательностях работает со входами или выходами разной длины, где важны порядок и контекст. Разметка, распознавание речи, синтез речи и перевод — задачи на последовательностях, которые различаются тем, как сопоставляются вход и выход.

Вопросы

1. Почему обычный многоклассовый классификатор плохо подходит для картинки с бременскими музыкантами?
 - a. На картинке слишком много пикселей для классификатора
 - b. Модель не может различать животных
 - c. Он должен выбрать один класс, а честный ответ — четыре животных сразу
 - d. Сказочных картинок никогда не бывает в обучающих данных

2. В чём главное отличие рекомендательной системы от поисковика?
- a. Рекомендательная система упорядочивает объекты под вкус конкретного человека
 - b. Рекомендательная система никогда не даёт объектам оценок
 - c. Поисковик работает только с фильмами
 - d. Поисковик не умеет ранжировать страницы
3. Рекомендательная система магазина продвигает один блендер, поэтому его покупают больше, поэтому его продвигают ещё сильнее. Что это за проблема?
- a. Переобучение на тестовой выборке
 - b. Петля обратной связи, когда собственные советы системы формируют данные, на которых она учится
 - c. Классификация с несколькими метками
 - d. Слишком мало оценок в три звезды
4. Какая задача превращает длинный вход в короткий выход?
- a. Синтез речи
 - b. Разметка каждого слова в предложении
 - c. Перевод с немецкого на английский
 - d. Распознавание речи

Ответы

1. c) Когда верны сразу несколько классов, это задача классификации с несколькими метками, которая позволяет модели сказать «осёл, собака, кот и петух» вместе.

- 2. a)** Обе расставляют подходящие объекты по порядку, но рекомендательная система личная: любитель фантастики и поклонник комедий видят разные списки.
- 3. b)** Покупки отчасти отражают сами рекомендации, поэтому товар кажется лучше, чем есть, и его рекомендуют ещё больше.
- 4. d)** При распознавании речи тысячи звуковых измерений образуют одно слово, поэтому текст гораздо короче записи; синтез речи идёт в обратную сторону.

Задачи

1. Книга размечает предложение «Tom has dinner in Washington with Sally» («Том ужинает в Вашингтоне с Салли»), пометая каждое слово, которое входит в имя. Какие слова получают пометку и почему говорят, что вход и выход в этой задаче совпадают?

Подсказка: Ищите людей и места и сравните число пометок с числом слов.

Ответ: Пометку получают Tom, Washington и Sally, а остальные четыре слова — нет. На каждое слово приходится ровно одна пометка в том же порядке, поэтому выход совпадает со входом один к одному, в отличие от перевода.

2. Музыкальное приложение учится на том, какие песни люди пропускают. Объясните, почему пропуск — более слабое свидетельство, чем оценка в одну звезду, и что это значит для рекомендательной системы.

Подсказка: Подумайте, по каким ещё причинам, кроме неприязни, пропускают песню.

Ответ: Пропуск — неявный отзыв: он может означать неприязнь, а может — только то, что песня не подошла к моменту. Оценка в одну

звезду прямо говорит, что человеку не понравилось. Поэтому рекомендательная система должна считать пропуски неуверенными подсказками, а не ясными приговорами.

3. Почему модель, предупреждающая об опасности в реанимации, должна быть моделью последовательностей, а не смотреть на показания каждого часа по отдельности?

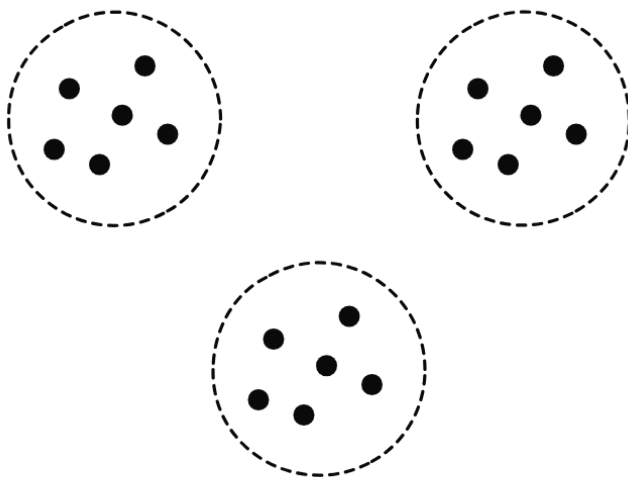
Подсказка: Как выглядит тревожная тенденция в одном показании?

Ответ: Опасность часто видна в истории, например в показаниях, медленно смещающихся за много часов, а этого не покажет ни одно отдельное показание. Модель, которая забывает данные каждого часа, не увидит такой тенденции, поэтому она должна учитывать всю последовательность измерений пациента.

Обучение без начальника и меняющийся мир

Откройте галерею на телефоне и найдите альбомы «люди» или «места». Никто не говорил телефону, чьи это лица и как выглядит пляж, и всё же он разложил тысячи снимков по стопкам, которые в основном имеют смысл. Это обучение другого рода, чем всё, что было до сих пор, потому что меток не было вовсе.

КЛАСТЕРИЗАЦИЯ

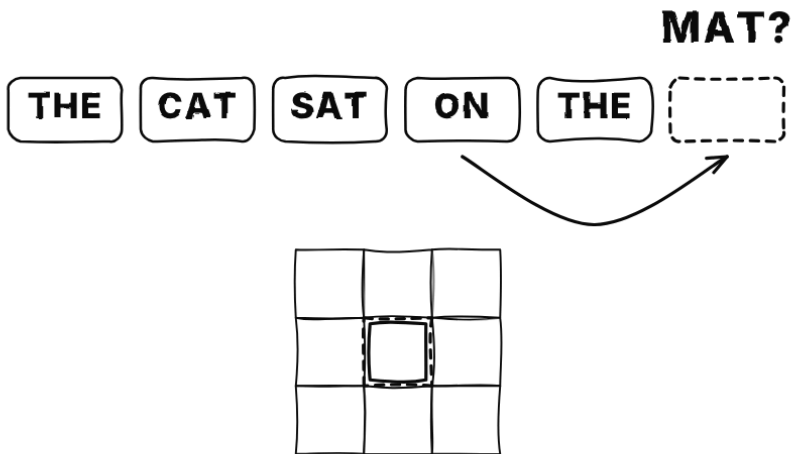


Группы без меток

Книга изображает обучение с учителем как работу с властным начальником, который стоит у вас за плечом и в каждом случае точно говорит, что делать. Скучно, но легко: вы копируете начальника, пока не уловите закономерность. А теперь представьте начальника, который сваливает вам на стол гору данных и говорит только: «Сделай с этим какую-нибудь науку о данных». Это обучение без учителя, и вопросы, которые можно задать, ограничены лишь вашим воображением. Можно разложить объекты по группам, как это делает галерея; это называется кластеризацией. Можно искать несколько чисел, которые улавливают главное, как портной описывает всю фигуру горсткой мерок. Можно поместить слова в пространство, где их смыслы ведут себя как положения, так что если от Рима отнять Италию и прибавить Францию, окажешься рядом с Парижем. Можно искать причины, спрятанные в таблицах цен, загрязнения и зарплат, или строить порождающие модели, которые так хорошо узнают, как выглядят данные, что могут создавать собственные новые примеры. Инструменты, которые рисуют новую картинку по набранной вами фразе, — порождающие модели именно такого рода.

Хитрый промежуточный путь — самообучение, при котором данные сами задают себе упражнения. Спрячьте слово в предложении и попросите модель угадать его по соседним словам; когда для практики есть все тексты мира, людям ничего не нужно размечать. С картинками можно закрыть кусочек и попросить модель его дорисовать. То, чему модель так научилась, потом дообучают для той работы, которая вам действительно нужна. Чат-бот, с которым вы разговариваете, начинал именно

так: учился угадывать пропущенные или следующие слова в огромном количестве текста.



Данные задают упражнение

У всех этих методов есть одно негласное допущение: сначала собирают данные, потом идёт обучение, и пока модель учится, она никак не касается мира. Это называется офлайн-обучением, и оно простое, но мир — не куча данных. Настоящие системы действуют, и их действия меняют то, что они увидят дальше. Одни среды помнят, что вы сделали, другие вам помогают, как человек, который отчётливо говорит в приложение для диктовки, а третьи с вами борются. Ваш спам-фильтр имеет дело с противником: спамеры изучают, что блокируется, и меняют письма, чтобы проскользнуть. Когда данные, которые модель

встречает в работе, отличаются от данных, на которых она обучалась, это сдвиг распределения. Пример из книги — экзамен, который составляет лектор после семестра домашних заданий, которые давали ассистенты. Как обучающаяся система может действовать в таком мире, — в следующей части.



Когда данные сдвигаются

Запомним

Обучение без учителя находит структуру в данных без меток: группы, несколько ключевых чисел, скрытые причины или новые примеры. Самообучение само делает метки из данных, например угадывает скрытые слова, а затем его дообучают для задачи. Большая часть обучения — офлайн: сначала данные, и модель никак не действует, пока учится. Когда мир, с которым встречается модель, отличается от данных, на которых она обучалась, — это сдвиг распределения.

Вопросы

1. Телефон раскладывает фотографии из отпуска по группам — пляжи, горы, лица, — хотя их никто не размечал. Что это за обучение?
 - a. Обучение с учителем, потому что группы верны
 - b. Обучение без учителя, а именно кластеризация
 - c. Обучение с подкреплением
 - d. Регрессия
2. Почему самообучению не нужна разметка людьми?
 - a. Данные сами задают себе упражнения, например скрытое слово, которое должны выдать соседние слова
 - b. Оно использует только маленькие наборы данных

- c. Оно копирует метки у модели, обученной с учителем
 - d. Оно учится только на изображениях
3. Спам-фильтр, обученный в прошлом году, теперь пропускает много нового спама. Что лучше всего это объясняет?
- a. Фильтр обучали онлайн
 - b. Фильтр использует кластеризацию
 - c. Сдвиг распределения: спамеры изменили письма, и новые данные отличаются от обучающих
 - d. У фильтра слишком мало параметров, чтобы читать почту
4. Какое допущение, по словам этой части, разделяют обучение с учителем и без учителя?
- a. Что у каждого примера есть метка
 - b. Что модель действует в мире, пока учится
 - c. Что тестовые данные составил тот же человек, что и обучающие
 - d. Что сначала собирают данные, а модель никак не действует, пока учится

Ответы

1. b) Меток не давали; поиск групп похожих примеров в неразмеченных данных — это кластеризация, вид обучения без учителя.
2. a) Если спрятать часть данных и попросить модель угадать её, любой текст или картинка превращается в материал для упражнений, а спрятанная часть служит ответом.

3. c) Среда сопротивляется; когда данные в работе отличаются от обучающих, качество модели падает.

4. d) Оба — офлайн-обучение: сначала куча данных, и ответы модели никак не меняют того, какие данные она видит.

Задачи

1. Книга сравнивает сдвиг распределения с экзаменом, который составил лектор после домашних заданий, которые давали ассистенты. Объясните своими словами, почему студент, хорошо справлявшийся с домашними заданиями, всё равно может растеряться, и как это переносится на модель.

Подсказка: Спросите себя, на чём студент тренировался и на чём его проверяли.

Ответ: Студент тренировался на вопросах в стиле ассистентов, а экзамен составлен в стиле лектора, так что вопросы отличаются от тех, на которых он учился. С моделью так же: она учится на обучающих данных, и если данные в работе приходят из другого источника или из изменившегося мира, её хорошие результаты на обучении могут не перенестись.

2. Для каждой среды скажите, помогает ли она обучающейся системе, борется с ней или ни то ни другое: человек, медленно и отчётливо говорящий в приложение для диктовки; спамеры, пишущие письма; погода.

Подсказка: Спросите себя, есть ли у среды цель, связанная с моделью.

Ответ: Внимательный говорящий помогает, потому что хочет, чтобы его поняли. Спамеры борются, подстраивая письма, чтобы обойти фильтр. Погода ни то ни другое: она может меняться со временем, но не в ответ на желания модели.

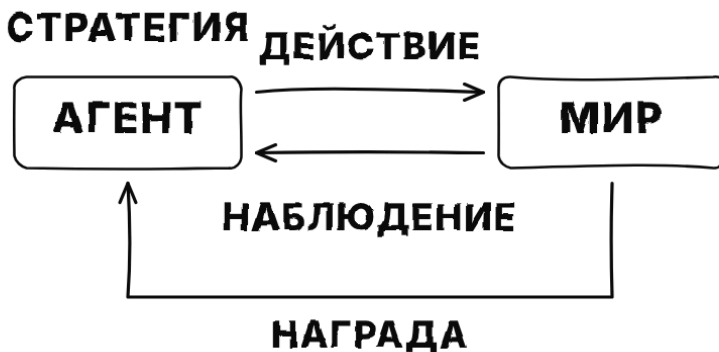
3. Предложите, как превратить большую коллекцию фотографий, ни одна из которых не размечена, в обучающие упражнения для самообучающейся модели.

Подсказка: Спрячьте то, что может выдать сама фотография.

Ответ: Закрывайте кусочек каждой фотографии и просите модель дорисовать его по остальному; правильный ответ даёт исходная фотография, так что никому не нужно ничего размечать. Модель узнаёт о формах и текстурах, и потом её можно дообучить для настоящей задачи.

Обучение действием и с чего всё началось

Посмотрите, как ребёнок осваивает видеоигру, не читая инструкций. Он жмёт на кнопки, экран меняется, очки растут или теряется жизнь, и через какое-то время он играет хорошо. Никто не говорил ему, какую кнопку нажимать в каждый момент; он видел только счёт. В этом идея обучения с подкреплением — вида обучения для тех, кто действует: роботов, игроков в игры и программ, которые ведут разговор.

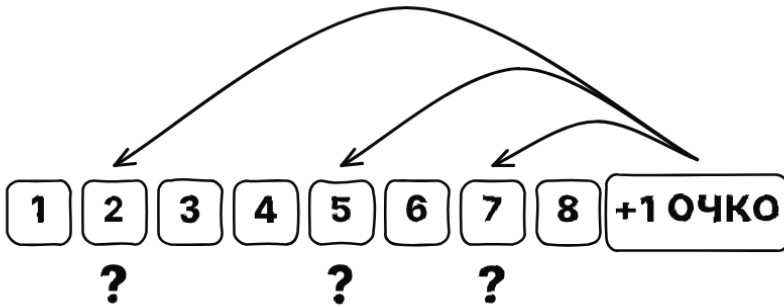


Петля агента

Её образ — петля. Агент смотрит на свой мир, и тот даёт ему наблюдение; агент выбирает действие; мир меняется и возвращает награду; и петля идёт по новому кругу. Привычка агента превращать то, что он видит, в то, что он делает, называется его стратегией, и вся цель — выучить хорошую. Идея настолько общая, что обычное обучение с учителем умещается внутри неё: одно действие на каждый класс и награда, равная тому, насколько хороша была догадка. Знаменитые успехи — программа, которая выучила старые игры Atari по одному лишь экрану и обыграла людей, и AlphaGo, обыгравшая чемпиона мира по го.

Трудность в том, чего мир не говорит. Никто не сообщает агенту лучшее действие, только награду, а награда может прийти намного позже действий, которые её заработали. В шахматах единственная настоящая награда — победа или поражение в конце, так какой из сорока ходов заслуживает заслуги? Это задача распределения заслуг, та же, что встаёт перед человеком, которого повысили после года хороших решений и который хочет знать, какие из них сыграли роль. Агент может видеть лишь часть своего мира, как робот-уборщик, застрявший в одном из нескольких одинаковых чуланов, который должен помнить, как туда попал, чтобы знать, где он. И ему всё время приходится выбирать между лучшим известным планом и попыткой чего-то нового, что может оказаться лучше: использовать или исследовать. Лента рекомендаций живёт с этим выбором каждый день: показать вам ещё того, что вам уже нравится, или попробовать новое и рискнуть, что вы пролистаете. Поскольку всё это сразу слишком сложно, исследователи изучают более простые случаи, которые книга называет, не вдаваясь в них.

ЗАСЛУГА?

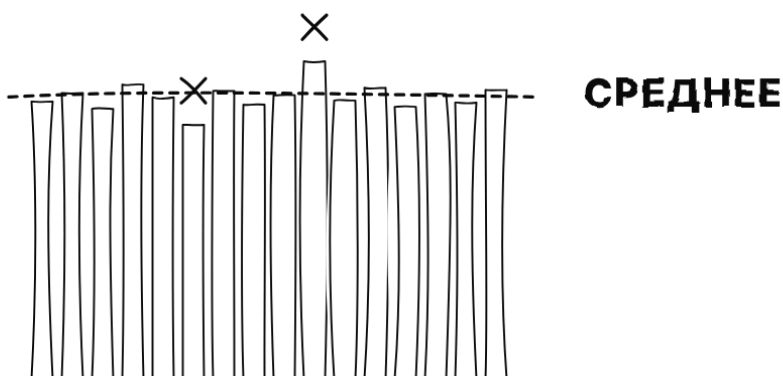


Какой ход принёс победу?

Теперь книга возвращается в историю, потому что желанию учиться на данных уже несколько веков. Гаусс дал нам метод наименьших квадратов, которым до сих пор пользуются в страховании и медицине, а простые линейные модели описывают такие законы, как связь тока и напряжения в проводе. В шестнадцатом веке геометр Якоб Кёбель описал, как установить длину фута: остановить шестнадцать мужчин, выходящих из церкви, выстроить их ступни в ряд и взять среднее; позднее улучшение — сначала убрать самую короткую и самую длинную ступню — одно из самых ранних усечённых средних. Рональд Фишер создал значительную часть современной статистики, и его небольшой набор данных об ирисах 1936 года до сих пор используют для демонстраций; он

был также сторонником евгеники — напоминание о том, что данные давно служат не только добру, но и злу. Свой вклад внесли и теория информации Шеннона, и вопрос Тьюринга: «Могут ли машины мыслить?»

16 СТУПНЕЙ



Фут Кёбеля

Другим источником вдохновения был мозг. Дональд Хебб предположил, что связи между нейронами усиливаются, когда помогают, и эта мысль — поощрять хорошее поведение и ослаблять плохое — проходит через то, как сегодня обучают нейронные сети. Нейронные сети взяли имя у мозга, хотя сходство стало слабым; остались слои простой обработки, поставленные друг на друга, и способ настраивать все слои сразу.

И всё же примерно с 1995 по 2005 год нейронные сети вышли из моды. Вычислительной мощности было мало, а наборы данных были крошечными: собрание из шестидесяти тысяч рукописных цифр считалось огромным. При таком малом количестве данных и вычислений другие методы, например деревья решений и ядерные методы, попросту работали лучше, обучались за часы, а не за недели, и давали надёжные гарантии. Многие спам-фильтры того времени строились на таких классических методах, а не на нейронных сетях. Как этот баланс сместился, — в следующей части.

Запомним

Обучение с подкреплением — для агентов, которые действуют: они наблюдают, действуют, получают награды и выучивают стратегию. Оно трудно, потому что лучшее действие никогда не показывают, награды приходят поздно, мир может быть виден лишь отчасти, а агент должен уравнивать пробы нового и использование того, что работает. Обучению на данных несколько веков: от усреднения ступней и наименьших квадратов Гаусса до статистики Фишера и вопроса Тьюринга. Нейронные сети позаимствовали у мозга имя и некоторые идеи. Примерно в 1995–2005 годах они уступили другим методам, потому что данных и вычислительной мощности было слишком мало.

Вопросы

1. Что делает обучение с подкреплением труднее обучения с учителем?
 - a. Агенту никогда не показывают лучшее действие, только награду, которая может прийти намного позже
 - b. У агента вообще нет наблюдений
 - c. Награды даются сразу за все возможные действия
 - d. Его нельзя использовать для игр
2. Шахматная программа выигрывает партию после сорока ходов. С какой задачей она сталкивается, обучаясь на этой победе?
 - a. Сдвиг распределения
 - b. Частичная наблюдаемость
 - c. Распределение заслуг: выяснить, какие из ходов заслуживают заслуги
 - d. Переобучение на тестовой выборке
3. Музыкальное приложение всё время играет только ваш любимый жанр и никогда не пробует ничего другого. Что оно делает в терминах обучения с подкреплением?
 - a. Исследует
 - b. Использует лучшую известную стратегию и никогда не исследует
 - c. Распределяет заслуги
 - d. Обучается офлайн
4. Почему примерно с 1995 по 2005 год другие методы обходили нейронные сети?
 - a. Нейронные сети ещё не были изобретены

- b. Другим методам нужно было больше данных
- c. Нейронные сети нельзя было обучать с наградами
- d. Данных и вычислительной мощности было мало, а другие методы обучались быстрее и с надёжными гарантиями

Ответы

1. **a)** В обучении с учителем каждый пример несёт правильный ответ; здесь мир даёт только награды, часто намного позже действий, которые их заработали.
2. **c)** Единственная награда приходит в конце, поэтому обучающийся должен решить, какие прежние ходы её заработали.
3. **b)** Применять лучший известный план — это использование; никогда не исследуя, приложение лишается шанса найти что-то лучше.
4. **d)** При маленьких наборах данных и малой вычислительной мощности деревья решений и ядерные методы работали лучше и обучались за часы, а не за недели.

Разобранные примеры

Усечённое среднее ступней В духе метода Кёбеля измерили ступни шести мужчин: 26, 27, 28, 27, 31 и 21 сантиметр. Найдите простое среднее, а затем усечённое среднее, убрав самую короткую и самую длинную ступню.

Простое среднее: шесть длин в сумме дают 160 сантиметров, а 160, делённое на 6, — около 26,7 сантиметра.

Чтобы усечь, уберите самую короткую ступню, 21, и самую длинную, 31, потому что необычные ступни тянули бы среднее в сторону.

Оставшиеся четыре в сумме — 26 плюс 27 плюс 28 плюс 27, то есть 108 сантиметров.

108, делённое на 4, — это 27 сантиметров.

Ответ: Простое среднее около 26,7 см; усечённое среднее 27 см

Задачи

1. Пять измерений ступни: 25, 29, 28, 27 и 36 сантиметров. Найдите усечённое среднее, убрав самое короткое и самое длинное, и объясните, почему здесь оно лучше простого среднего.

Подсказка: Сначала уберите 25 и 36, потом усредните остальное.

Ответ: 28 см: 29 плюс 28 плюс 27 — это 84, а 84, делённое на 3, — 28. Простое среднее, 29 см, тянет вверх одна необычно длинная ступня, которую усечённое среднее не учитывает.

2. Книга описывает робота-уборщика, запертого в одном из нескольких одинаковых чуланов. Объясните, почему роботу недостаточно осмотреться в чулане, чтобы понять, где он, и что ему нужно вместо этого.

Подсказка: Что робот увидел бы в каждом из чуланов?

Ответ: Изнутри все чуланы выглядят одинаково, поэтому текущее наблюдение не раскрывает состояния робота; это частичная наблюдаемость. Робот должен использовать прежние наблюдения — как он двигался, прежде чем войти, — чтобы понять, в каком он чулане.

Что изменилось: данные, чипы и инструменты

Вспомните фотографии в своём телефоне: их тысячи, каждая — несколько миллионов пикселей, они хранятся почти бесплатно и не задумываясь копируются в облако. Тридцать лет назад память целого компьютера не вместила бы ни одной из них. Именно эта переменная — в том, сколько существует данных и как дёшево их хранить и обрабатывать, — вернула нейронные сети из забвения.

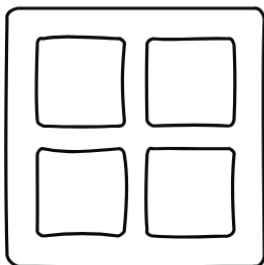


Рост с 1970 по 2020 год

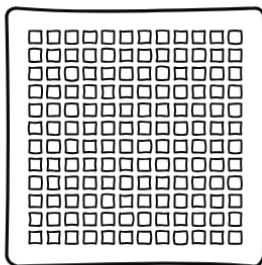
Книга показывает это таблицей, по строке на десятилетие. В семидесятых годах типичный набор данных составлял около сотни примеров, как ирисы Фишера; к двадцатым годам этого века данные социальной сети доходили примерно до триллиона. Память тоже росла, но гораздо медленнее: примерно от килобайта до сотни гигабайт. Быстрее всего росла вычислительная мощность: от ста тысяч операций в секунду примерно до миллиона миллиардов. Самое интересное — вывод: раз вычисления обогнали память, лучшие модели те, что хранят мало, а вычисляют много, и именно это делают глубокие нейронные сети. Поэтому многие старые идеи, годами лежавшие без дела, были заново открыты в последнее десятилетие.

Решающей машиной стала видеокарта. Графические процессоры создавали, чтобы рисовать видеоигры, выполняя огромное число простых вычислений параллельно, и оказалось, что именно это и нужно для обучения нейронной сети. Вдруг модели, казавшиеся немыслимо дорогими, стали доступны. Распределение работы между многими видеокартами сократило время обучения стандартной модели для изображений с нескольких дней до менее чем семи минут.

CPU



GPU

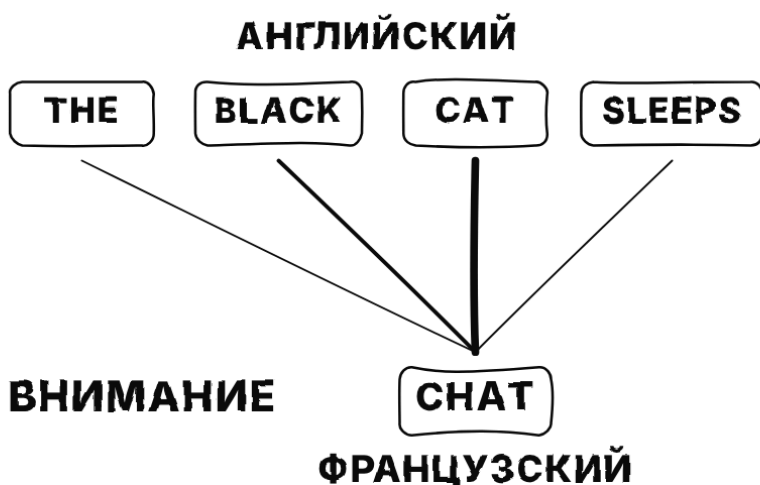


ДНИ → МИНУТЫ

Много маленьких работников

Но дело было не только в том, что на старые рецепты вылили больше мощности; пришёл и взрыв новых идей, который книга сравнивает с внезапным расцветом жизни в кембрийском периоде. Дропаут добавляет случайный шум внутрь сети во время обучения, и это помогает против переобучения. Механизм внимания позволяет модели указывать назад, на те части входа, которые важны прямо сейчас, вместо того чтобы втискивать в память всё предложение перед переводом. Трансформер, построенный только на внимании, продолжает улучшаться с ростом данных, размера модели и вычислений, и единственный трансформер научился и болтать, и подписывать картинки, и играть в Atari, и управлять роботом. Языковые модели, которые предсказывают текст по тексту, при

масштабировании обрели удивительные способности. ChatGPT, которым вы, возможно, пользуетесь сами, — такая языковая модель, настроенная выполнять то, о чём люди её действительно просят. Порождающие состязательные сети, а позднее диффузионные модели научились создавать реалистичные изображения; диффузионные модели строят картинку, шаг за шагом убирая шум. Приложения, которые рисуют картинку по набранной вами фразе, используют диффузионные модели именно такого рода.



Внимание указывает назад

Последний ингредиент — программы. Бесплатные фреймворки глубокого обучения, от ранних вроде Theano и Torch до нынешних PyTorch, TensorFlow и JAX, берут на себя трудную

математику и работу с железом. Книга даёт меру того, насколько это всё изменило: в 2014 году обучение простого классификатора давали как настоящее домашнее задание новым аспирантам Университета Карнеги — Меллона; сегодня на это уходит меньше десяти строк кода. В этой части мы опустили подробности каждого метода — о них пойдёт речь в следующих главах.

Запомним

Нейронные сети вернулись, потому что данных стало много, а хранение и вычисления подешевели. Вычислительная мощность росла быстрее памяти, а это благоприятствует моделям, которые хранят мало и вычисляют много, как глубокие сети. Видеокарты, созданные для игр, оказались идеальны для их обучения. Новые идеи — дропаут, внимание, трансформеры, языковые модели и диффузионные модели — двигали прогресс не меньше, чем голая мощность. Бесплатные фреймворки сделали всё это достаточно простым для любого программиста.

Вопросы

1. Вычислительная мощность росла гораздо быстрее памяти. Каким моделям это благоприятствует?
 - а. Моделям, которые хранят очень много, а вычисляют мало

- b. Моделям, которые хранят мало, а вычисляют много, как глубокие нейронные сети
 - c. Моделям, которым не нужны данные
 - d. Моделям, которые работают без компьютера
2. Почему видеокарты оказались так полезны для глубокого обучения?
- a. У них больше памяти, чем у любого другого чипа
 - b. Их с самого начала проектировали для нейронных сетей
 - c. Они выполняют огромное число простых вычислений параллельно, а это и нужно для обучения
 - d. Они дешевле хранят наборы данных
3. Что механизм внимания позволяет делать модели перевода?
- a. Указывать назад, на части входа, важные сейчас, вместо того чтобы сначала держать в памяти всё предложение
 - b. Добавлять случайный шум, чтобы избежать переобучения
 - c. Создавать изображения из шума
 - d. Обучаться сразу на многих видеокартах
4. Что, по словам книги, сделало глубокое обучение доступным обычным программистам?
- a. Меньшие наборы данных
 - b. Конец переобучения
 - c. Обучение с подкреплением
 - d. Бесплатные фреймворки, которые берут на себя математику и железо

Ответы

1. **b)** Когда вычисления дешёвы, а память относительно скудна, выгоднее всего модели, тратящие много вычислений на немногие хранимые числа, — именно так работают глубокие сети.
2. **c)** Графические процессоры создавали, чтобы рисовать игры, выполняя много простых вычислений одновременно, и обучение нейронной сети — работа именно такого рода.
3. **a)** С вниманием модели не нужно втискивать всё предложение в память фиксированного размера; по ходу дела она оглядывается на нужные части.
4. **d)** Задача, которая в 2014 году была домашним заданием аспирантов, теперь занимает меньше десяти строк кода благодаря фреймворкам.

Разобранные примеры

Много карт, один большой пакет Обучение распределено по 1024 видеокартам, и каждая карта обрабатывает за раз небольшой пакет из 32 изображений. Сколько изображений обрабатывается вместе за один шаг на всех картах?

Каждая карта берёт свой пакет из 32 изображений.

Все 1024 карты работают одновременно, поэтому пакеты складываются.

1024 умножить на 32 — это 32 768.

Ответ: 32 768 изображений, около 32 000

Задачи

1. В таблице из книги типичный набор данных вырос со 100 примеров в 1970 году до 1 триллиона в 2020-м, а память — с 1

килобайта до 100 гигабайт. Считая гигабайт за миллион килобайт, во сколько раз выросло каждое и что выросло больше?

Подсказка: Триллион — это единица с двенадцатью нулями; 100 гигабайт — это 100 миллионов килобайт.

Ответ: Данные выросли в десять миллиардов раз (со 100 до триллиона); память — в сто миллионов раз (с 1 до 100 миллионов килобайт). Данные выросли примерно в сто раз больше, поэтому память не поспевает, и модели должны бережно расходовать память.

2. Если бы 512 карт брали каждая пакет из 64 изображений, сколько изображений обрабатывал бы один шаг и как это соотносится с 1024 картами по 32?

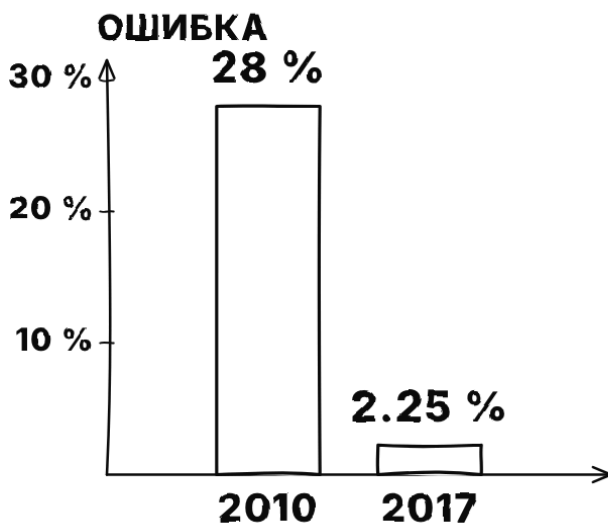
Подсказка: Умножьте число карт на пакет каждой карты.

Ответ: 512 умножить на 64 — это 32 768 изображений, ровно столько же, сколько 1024 на 32: если вдвое уменьшить число карт и вдвое увеличить пакет каждой, итог не изменится.

Где это уже работает и что делает его глубоким

Оплатите сегодня что-нибудь в интернете — и в ту секунду, пока платёж не прошёл, модель проверит его на мошенничество. Отправьте письмо — и во многих странах машина прочтёт рукописный адрес; именно эта задача дала нам знаменитое собрание рукописных цифр. Машинное обучение делает такую тихую работу с девяностых годов: читает чеки, оценивает заявки на кредит, ранжирует результаты поиска, выбирает, что вам показать. Оно повсюду, но в основном незаметно.

Новое то, что теперь оно делает вещи, которые люди видят и которые считались невозможными. Голосовые помощники довольно хорошо выполняют устные просьбы, от включения света до записи в парикмахерскую, а распознавание речи в некоторых задачах сравнялось с людьми. Когда вы спрашиваете голосового помощника о завтрашней погоде и он понимает вас в шумной комнате, вы слышите этот прогресс. Распознавание предметов на фотографиях из трудного стало обыденным: на стандартном тесте лучшие системы 2010 года ошибались примерно в двадцати восьми процентах случаев, даже когда им давали пять попыток; к 2017 году — примерно в двух процентах. Те же методы теперь узнают птиц по пению и помогают диагностировать рак кожи.

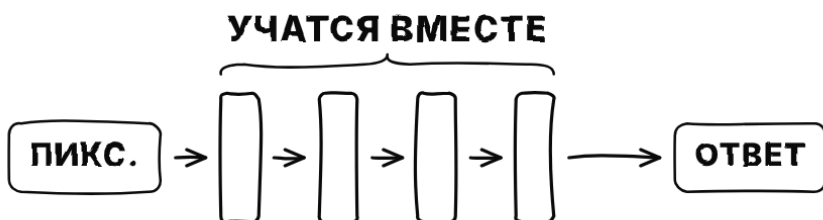
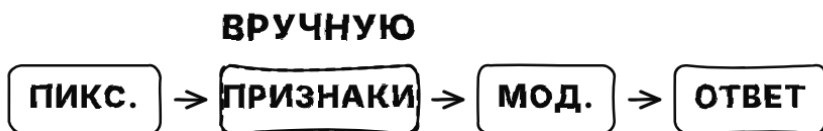


Ошибка на ImageNet

Игры давно служат мерилom. Программа для нард научилась с подкреплением ещё десятилетия назад; Deep Blue обыграл Каспарова в шахматы благодаря особому железу и грубому перебору; го, где возможных позиций гораздо больше, покорила AlphaGo, соединившей глубокое обучение с выборочным обходом дерева ходов; а программа Libratus обыграла сильнейших игроков в покер — игру, где не видно карт соперников. Беспилотные автомобили добились настоящего прогресса, хотя полная автономия всё ещё недостижима; там глубокое обучение в основном отвечает за зрение, а остальное инженеры настраивают вручную. Робототехника, логистика, биология, физика частиц и астрономия обязаны этим инструментам недавними успехами.

А как же страх перед машинами, которые обретут сознание и обратятся против нас? Авторы относятся к этому спокойно. Нынешние системы строят для конкретных целей, и за любой видимостью общего интеллекта стоят правила, эмпирические приёмы и статистические модели; инструментов для машины, которая сама себя улучшает и перестраивает, чтобы решить любую задачу, нет. Заботы, которые они считают более насущными, ближе: автоматизация рутинной работы — как сельскохозяйственные роботы автоматизируют сбор урожая, — с большими последствиями для людей, живущих такой работой; и небрежные модели, принимающие предвзятые решения о кредитах, работе или залоге. Та же осторожность нужна и с лентой рекомендаций в вашем телефоне, которая каждый день принимает мелкие автоматические решения о том, что видят миллионы людей.

Так что же в итоге такое глубокое обучение? Это машинное обучение с моделями из многих слоёв преобразований, которые все вместе выучиваются по данным. До него эксперты годами вручную придумывали шаги, превращающие сырые пиксели или звук в полезные признаки, и только потом отдавали эти признаки простой модели. Глубокое обучение обучает всю цепочку сразу, от начала до конца, от сырого входа до ответа, и фильтры, которые оно находит само, оказываются лучше придуманных людьми. Вот почему оно блистает на сыром звуке, сырых пикселях и предложениях любой длины и почему оно разрушило стены между зрением, речью, языком и медициной, дав им один общий набор инструментов.



Вручную или сквозное обучение

Оно отражает и смену стиля. Когда данных много, область меньше опирается на упрощающие допущения и больше на гибкие модели, плотно подогнанные к данным, — так физики перешли от приближений, решённых вручную, к компьютерному моделированию: точнее, хотя зачастую труднее для истолкования. Она принимает решения, которые хороши, а не доказуемо лучшие, и пробует, прежде чем доказывать. И она делится: инструменты, модели и обученные сети выпускаются открыто, и поэтому эта книга тоже бесплатна.

На этом введение заканчивается. На протяжении этих частей рассказ шёл от программ, написанных как правила, к программам, которые учатся на примерах, через множество

форм, которые может принимать обучение, к данным, чипам и инструментам, благодаря которым глубокие сети наконец заработали.

Запомним

Машинное обучение давно работает незаметно — в проверках на мошенничество, сортировке почты и поиске, — а глубокое обучение принесло зримые успехи в речи, изображениях и играх. Авторы видят настоящие риски не в машинах с сознанием, а в потерянных рабочих местах и предвзятых автоматических решениях. Глубокое обучение — это много слоёв преобразований, выученных вместе, от начала до конца, от сырых данных до ответа. Оно заменило признаки, придуманные вручную, и дало таким областям, как зрение, речь и язык, один общий набор инструментов. Его прогресс держится на обилии данных, дешёвых вычислениях и видеокάρтах и на открытых фреймворках, которыми может пользоваться любая.

Вопросы

1. Что, по словам этой части, главным образом отличает глубокое обучение от прежнего машинного обучения?
 - а. Оно вообще не использует данных
 - б. Его многочисленные слои выучиваются вместе, от начала до конца, вместо признаков, придуманных

- вручную и подаваемых в простую модель
- с. Оно работает только с играми
 - д. Оно всегда находит доказуемо лучшее решение
2. Какой риск ИИ авторы считают сегодня более насущным?
- а. Что машины обретут сознание и обратятся против своих создателей
 - б. Что у компьютеров кончится память
 - с. Автоматизацию рутинной работы и небрежные модели, принимающие предвзятые решения
 - д. Что игры станут для компьютеров слишком лёгкими
3. Почему покер был для компьютера задачей другого рода, чем шахматы?
- а. В покере меньше возможных позиций, чем в нардах
 - б. В покере нет правил
 - с. Шахматам нельзя научиться с подкреплением
 - д. В покере не видно карт соперников, так что игра наблюдаема лишь отчасти
4. Почему глубокое обучение разрушило стены между такими областями, как зрение, речь и язык?
- а. Потому что сквозное обучение заменяет значительную часть подготовки, специфичной для каждой области, и одни и те же инструменты работают везде
 - б. Потому что все эти области теперь используют одни и те же наборы данных
 - с. Потому что зрение и речь — одна и та же задача

d. Потому что фреймворки запрещают специализированные методы

Ответы

1. **b)** Раньше эксперты придумывали шаги получения признаков вручную; глубокое обучение обучает всю цепочку от сырого входа до ответа сразу.
2. **c)** Они не видят инструментов для самоулучшающегося общего интеллекта, но видят реальные последствия автоматизации и предвзятых автоматических решений о кредитах, работе или залоге.
3. **d)** В шахматах видна вся доска, а в покере часть ситуации скрыта, и рассуждать гораздо труднее.
4. **a)** Когда признаки выучиваются по сырым данным, а не придумываются экспертами каждой области, всем им служит один общий набор инструментов.

Разобранные примеры

Насколько лучше стало распознавание предметов На тесте ImageNet лучшие системы 2010 года имели долю ошибок «топ-5» 28 процентов; к 2017 году — 2,25 процента. Сколько из 10 000 тестовых изображений распознала неверно каждая и во сколько раз это меньше ошибок?

Доля ошибок — это доля примеров, на которых модель ошибается.

В 2010 году: 28 процентов от 10 000 — это 2800 ошибочных изображений.

В 2017 году: 2,25 процента от 10 000 — это 225 ошибочных изображений.

2800, делённое на 225, — около 12,4.

Ответ: 2800 ошибок против 225: примерно в двенадцать раз меньше

Задачи

1. Систему с долей ошибок «топ-5» 2,25 процента запускают на 40 000 фотографий. Примерно на скольких она ошибётся и на скольких ошиблась бы система 2010 года с долей ошибок 28 процентов?

Подсказка: Умножьте число фотографий на каждую долю ошибок, записанную дробью.

Ответ: 900 у системы 2017 года (0,0225 умножить на 40 000) и 11 200 у системы 2010 года (0,28 умножить на 40 000).

2. Объясните своими словами разницу между старым способом распознавать предметы на изображениях — с выделением признаков, придуманным вручную, — и способом глубокого обучения. Почему победил новый способ?

Подсказка: Спросите себя, кто в каждом случае решает, как сырые пиксели превращаются в признаки.

Ответ: При старом способе эксперты вручную придумывали шаги, превращающие пиксели в признаки, например детекторы границ, а затем простая модель использовала эти признаки. В глубоком обучении вся цепочка, включая признаки, выучивается по данным от начала до конца. Он победил, потому что алгоритм, автоматически перебирающий миллионы вариантов, находит фильтры лучше, чем одна лишь человеческая изобретательность.