

Содержание

Предисловие	10
Об авторе	12
О рецензентах	13
Введение	14
Для кого предназначена эта книга	15
Что охватывает эта книга	16
Как получить максимальную отдачу от этой книги	18
Когда читать данную книгу	19
Скачивание файлов с примерами кода	21
Условные обозначения	21
Комментарии переводчика	22
Глава 1. Начало работы с платформой TensorFlow Mobile	24
Настройка платформы TensorFlow	24
Настройка TensorFlow в MacOS	26
Настройка TensorFlow в Ubuntu с поддержкой GPU	28
Настройка среды разработки Xcode	32
Настройка среды разработки Android Studio	33
TensorFlow Mobile против TensorFlow Lite	35
Выполнение примеров приложений TensorFlow для iOS	36
Выполнение примеров приложений TensorFlow для Android	37
Резюме	38
Глава 2. Классифицирование изображений с помощью трансферного обучения	39
Трансферное обучение – что это такое и почему	40
Вторичная тренировка с использованием модели Inception v3	41
Вторичная тренировка с использованием моделей MobileNet	49
Использование вторично натренированных моделей в примере приложения для iOS	52
Использование вторично натренированных моделей в примере приложения для Android	54

Добавление платформы TensorFlow в свое собственное приложение для iOS	56
Добавление платформы TensorFlow в свое собственное приложение для iOS на языке Objective-C.....	56
Добавление платформы TensorFlow в свое собственное приложение для iOS на языке Swift	62
Добавление платформы TensorFlow в свое собственное приложение для Android.....	67
Резюме.....	71
Глава 3. Обнаружение и локализация объектов.....	72
Обнаружение объектов – краткий обзор	73
Настройка API TensorFlow обнаружения объектов.....	75
Быстрая установка и пример	76
Использование предварительно натренированных моделей.....	77
Вторичная тренировка моделей на основе SSD-MobileNet и более быстрого RCNN-детектора	81
Использование моделей обнаружения объектов в iOS	86
Ручная сборка библиотек TensorFlow для iOS	87
Использование библиотек TensorFlow для iOS в приложении	87
Добавление функционала обнаружения объектов в приложение для iOS.....	90
Использование YOLO2 – еще одной модели обнаружения объектов	96
Резюме.....	99
Глава 4. Трансформирование рисунков с помощью художественных стилей	101
Нейронный перенос стиля – краткий обзор.....	102
Тренировка моделей быстрого нейронного переноса стиля	103
Использование моделей быстрого нейронного переноса стиля в iOS	106
Добавление и тестирование с моделями быстрого нейронного переноса.....	107
Анализ программного кода iOS с использованием моделей быстрого нейронного переноса	109
Использование моделей быстрого нейронного переноса стиля в Android....	111
Использование многостилевой модели TensorFlow Magenta в iOS.....	117
Использование многостилевой модели TensorFlow Magenta в Android	123
Резюме.....	127
Глава 5. Понимание простых речевых команд.....	129
Распознавание речи – краткий обзор	130
Тренировка простой модели распознавания команд.....	132
Использование простой модели распознавания речи в Android	136
Создание нового приложения с использованием модели.....	137

Вывод результатов распознавания.....	142
Использование простой модели распознавания речи в iOS на языке Objective-C.....	145
Создание нового приложения с использованием модели.....	145
Исправление ошибок загрузки модели с помощью файла <code>tf_op_files.txt</code>	150
Использование простой модели распознавания речи в iOS на языке Swift.....	152
Резюме.....	156
Глава 6. Описание изображений на естественном языке.....	157
Аннотирование изображений – как оно работает	158
Тренировка и замораживание модели аннотирования изображений	160
Тренировка и тестирование генерирования аннотаций	160
Замораживание модели аннотирования изображений.....	163
Трансформация и оптимизация модели аннотирования изображений	170
Исправление ошибок в трансформированных моделях.....	170
Оптимизация трансформированной модели	174
Использование модели аннотирования изображений в iOS.....	175
Использование модели аннотирования изображений в Android	185
Резюме.....	191
Глава 7. Распознавание рисунков с помощью CNN- и LSTM-сетей.....	193
Классификация рисунков – как это работает	194
Тренировка, предсказание и подготовка модели классификации рисунков.....	196
Тренировка модели классификации рисунков.....	197
Предсказание с помощью модели классификации рисунков	198
Подготовка модели классификации рисунков	200
Использование модели классификации рисунков в iOS	205
Сборка пользовательской библиотеки TensorFlow для iOS	206
Разработка приложения для iOS с использованием модели	207
Использование модели классификации рисунков в Android	214
Сборка пользовательской библиотеки TensorFlow для Android	215
Разработка приложения для Android с целью применения модели	217
Резюме.....	225
Глава 8. Предсказание биржевой цены с помощью RNN-сети.....	227
RNN-сеть и предсказание биржевой цены – что это такое и как это делается	228
Использование API TensorFlow RNN для предсказания биржевой цены.....	230

Тренировка RNN-модели в TensorFlow	231
Тестирование модели TensorFlow RNN	235
Использование API RNN LSTM библиотеки Keras для предсказания биржевой цены	237
Тренировка LSTM-модели в библиотеке Keras	238
Тестирование модели Keras RNN	241
Выполнение моделей TensorFlow и Keras в iOS	243
Выполнение моделей TensorFlow и Keras в Android	250
Резюме	255

Глава 9. Генерирование и улучшение изображений

с помощью GAN-сети	257
GAN-сеть – что это такое и почему	258
Построение и тренировка GAN-моделей с помощью TensorFlow	260
Базовая GAN-модель генерирования рукописных цифр	260
Продвинутая GAN-модель улучшения разрешающей способности изображения	263
Использование GAN-моделей в iOS	267
Использование базовой GAN-модели	270
Использование продвинутой GAN-модели	272
Использование GAN-моделей в Android	275
Использование базовой GAN-модели	277
Использование продвинутой GAN-модели	279
Резюме	281

Глава 10. Создание мобильного игрового

AlphaZero-подобного приложения	283
Алгоритм AlphaZero – как он работает?	284
Тренировка и тестирование AlphaZero-подобной модели для игры «Четыре в ряд»	286
Тренировка модели	287
Тестирование модели	291
Анализ программного кода построения модели	294
Заморозка модели	295
Использование модели игры «Четыре в ряд» в iOS	296
Использование модели игры «Четыре в ряд» в Android	309
Резюме	320

Глава 11. Применение платформ TensorFlow Lite и Core ML

на мобильных устройствах	322
Платформа TensorFlow Lite – краткий обзор	323
Использование платформы TensorFlow Lite в iOS	324

Выполнение примеров приложений TensorFlow Lite для iOS	324
Использование готовой модели TensorFlow Lite в iOS.....	326
Использование вторично натренированной модели TensorFlow для платформы TensorFlow Lite в iOS.....	331
Использование пользовательской модели TensorFlow Lite в iOS.....	332
Использование платформы TensorFlow Lite в Android	334
Платформа Core ML для iOS – краткий обзор.....	338
Использование платформы Core ML с машинным обучением на основе библиотеки Scikit-Learn	339
Построение и конвертирование моделей Scikit-Learn.....	339
Использование конвертированных в формат Core ML моделей в iOS.....	341
Использование платформы Core ML с Keras и TensorFlow.....	343
Резюме.....	348

Глава 12. Разработка приложений TensorFlow

на компьютере Raspberry Pi	350
Настройка компьютера Raspberry Pi и приведение его в движение.....	351
Настройка материнской платы Raspberry Pi	353
Приведение компьютера Raspberry Pi в движение	355
Настройка платформы TensorFlow на компьютере Raspberry Pi.....	357
Распознавание изображений и речевое воспроизведение текста.....	360
Распознавание звука и движение робота	362
Самообучение с подкреплением на компьютере Raspberry Pi.....	366
Описание симулируемой среды CartPole.....	367
Начало с простой интуитивно понятной линии поведения	371
Использование нейронных сетей для построения более оптимальной линии поведения	373
Резюме.....	381
Послесловие.....	382

*Лизе и Вози, которые показали мне, что безусловная
любовь и поддержка могут уживаться в ладу
со случайной потребностью во внимании*

Предисловие

В последнее десятилетие происходит бурное развитие как машинного обучения, так и смартфонов; сегодня эти технологии наконец сливаются воедино, в результате чего образуется невероятное разнообразие приложений, от которых всего несколько лет назад вы бы отмахнулись, как от научной фантастики далекого будущего. Только подумайте: вы уже привыкли разговаривать с телефоном, спрашивать у него дорогу или поручать ему назначать время встречи в вашем графике. Фотокамера телефона отслеживает лица и распознает объекты. Игры становятся все интереснее и сложнее, поскольку боты становятся умнее и умнее. И бесчисленные программные приложения используют под капотом ту или иную форму искусственного интеллекта все менее очевидными способами, например рекомендуя контент, который вам понравится, предугадывая ваш следующий маршрут, чтобы сообщить, когда следует уйти, предлагая, что печатать дальше, и т.д.

До недавнего времени весь интеллект располагался на стороне сервера, а это означало, что пользователь должен был оставаться подключенным к интернету, в идеале, быстрым и стабильным соединением. Неизбежные задержки и сбои в обслуживании были тормозом для многих приложений. Но сегодня интеллект находится прямо у вас на ладони благодаря огромным аппаратным улучшениям и усовершенствованным библиотекам машинного обучения.

Самое главное, что эти технологии теперь полностью демократизированы: практически любой инженер-программист может научиться программировать интеллектуальное мобильное приложение на основе глубоких нейронных сетей, используя TensorFlow, мощную библиотеку глубокого обучения компании Google с открытым исходным кодом. Замечательная и уникальная книга Джеффа Танга покажет вам, как разрабатывать локальные приложения на основе TensorFlow для iOS, Android и Raspberry Pi, проведя вас через множество конкретных примеров с пошаговыми инструкциями и с большим трудом добытыми советами по устранению неполадок: начиная с классифицирования изображений, обнаружения объектов, аннотирования изображений и распознавания рисунков и заканчивая распознаванием речи, предсказанием временных рядов, генеративно-сопоставительными сетями, самообучением на основе максимизации вознаграждения и даже построением интеллектуальных игр с использованием AlphaZero – усовершенствованной технологии, построенной поверх программы AlphaGo, которая одержала верх над чемпионами мира по игре го Ли Седодем и Ке Цзе.

Эта книга станет очень популярной, поскольку в ней затронута чрезвычайно важная тема, о которой трудно получить хорошую достоверную информацию. Так что пора засучить рукава. Впереди вас ждет захватывающее путешествие! Ну, и какое же интеллектуальное мобильное приложение вы будете строить?

Орелиен Жерон,

экс-лидер команды классифицирования видеороликов YouTube и автор книги Hands-On Machine Learning with Scikit-Learn and TensorFlow («Практическое машинное обучение с помощью Scikit-Learn и TensorFlow», O'Reilly, 2017)

Париж, 11 мая 2018 г.

Об авторе

Джефф Танг влюбился в классический ИИ более двух десятилетий назад. После получения степени магистра наук в области информатики он в течение 2 лет работал над машинным переводом и затем, чтобы пережить долгую зиму ИИ¹, трудился над корпоративными, голосовыми, веб- и мобильными приложениями в стартапах, компаниях AOL, Baidu и Qualcomm. Он разработал пользующееся исключительным спросом приложение iOS с миллионами скачиваний и был удостоен звания ведущего разработчика Android Market компанией Google. Он вновь присоединился к современному ИИ в 2015 году, зная наперед, что ИИ будет его страстью и приверженностью в течение следующих двух десятилетий. Одна из его любимых тем – сделать ИИ доступным в любое время в любом месте, в результате чего и появилась настоящая книга.

Хотел бы поблагодарить Лариссу Пинто за то, что подсказала мне идею книги, Флавиана Ваза и Ахила Наира за отзывы во время редактирования черновика. Большое спасибо Питу Уордену, руководителю направления мобильных приложений TensorFlow в Google, за его помощь до и после того, как он стал техническим рецензентом книги, и Амите Капур, еще одному рецензенту книги, которая предоставила мне ценные отзывы. Особая благодарность Орелиену Жерону, автору бестселлера по практическому МО, за любезные ответы на все мои электронные письма, свои соображения и предисловие к книге, для которого он нашел время, несмотря на свое плотное расписание, – Мерси боку, Орелиен.

Я также искренне ценю понимание и поддержку всей моей семьи, помимо Лизы и Вози, во время курортного сезона и все месяцы, пока я как сумасшедший день и ночь работал над книгой, – спасибо Эми, Анне, Дженни, Софии, Марку, Сэнди и Бену.

¹ Зима ИИ – это период сокращения фондирования и падения интереса к исследованиям в области искусственного интеллекта. См. https://en.wikipedia.org/wiki/AI_winter. – Прим. перев.

О рецензентах

Пит Уорден является техническим руководителем направления мобильных и встроенных приложений TensorFlow в команде Google Brain¹.

Амита Капур является доцентом кафедры электроники Университета Дели SRCASW с 1996 года. Получатель престижной стипендии DAAD для продолжения части своей исследовательской работы в Технологическом институте Карлсруэ, Германия. Награждена лучшей презентационной наградой на международной конференции Photonics 2008. Член различных профессиональных организаций и имеет более 40 публикаций в международных журналах и конференциях. В настоящее время ее научные направления включают МО, ИИ, нейронные сети, робототехнику, буддизм и этику в ИИ.

¹ Google Brain – это исследовательский проект Google по изучению искусственного интеллекта на основе глубокого обучения. См. https://ru.wikipedia.org/wiki/Google_Brain. – *Прим. перев.*

Введение

Искусственный интеллект (ИИ), симулирование человеческого интеллекта в компьютерах, имеет долгую историю. С момента своего официального рождения в 1956 году ИИ пережил несколько периодов подъема и спада. Нынешнее возрождение интереса к ИИ, или новая революция ИИ, началось в 2012 году вместе с инновационным прорывом в области глубокого обучения, отрасли машинного обучения, которая сегодня стала самым популярным направлением ИИ, после того как **глубокие сверточные нейронные сети** (Deep Convolutional Neural Network, DCNN) одержали верх в конкурсной инициативе ImageNet по распознаванию изображений (Large Scale Visual Recognition Challenge, ILSVRC) с частотой ошибок всего 16,4% по сравнению с идущей на втором месте не DCNN-моделью с частотой ошибок 26,2%. Начиная с 2012 года улучшенные модели на основе глубокой сверточной нейронной сети выигрывают конкурс ImageNet ежегодно, а технология глубокого обучения применяется ко многим сложным задачам ИИ, находящимся за пределами компьютерного зрения, таким как распознавание речи, машинный перевод и игра в го, в результате чего происходил один инновационный прорыв за другим. В марте 2016 года программа AlphaGo компании Google DeepMind, построенная с применением глубокого самообучения на основе максимизации вознаграждения, победила 18-кратного чемпиона мира Ли Седоля со счетом 4:1. На ежегодном фестивале разработчиков Google I/O в 2017 году компания Google объявила, что они переходят от мобильного мира (mobile-first world), то есть технологий, предназначенных в первую очередь для мобильных устройств, к миру ИИ. Другие ведущие компании, такие как Amazon, Apple, Facebook и Microsoft, продолжали вкладывать значительные средства в ИИ и запускать множество продуктов на базе ИИ.

TensorFlow – это платформа Google с открытым исходным кодом для создания приложений машинного обучения. С момента своего первого выпуска в ноябре 2015 года, когда уже существовало несколько популярных платформ глубокого обучения с открытым исходным кодом, платформа TensorFlow быстро, менее чем за 2 года, стала самым популярным каркасом глубокого обучения с открытым исходным кодом. С тех пор еженедельно строились новые модели TensorFlow для решения всевозможных задач, которые требуют человеческого или даже сверхчеловеческого интеллекта, были опубликованы десятки книг по TensorFlow, и еще больше появилось сетевых журналов, учебных пособий, курсов и видеороликов, посвященных TensorFlow. Популярность ИИ и TensorFlow совершенно очевидна, но для чего нужна еще одна книга с термином «TensorFlow» в названии?

Эта книга уникальна, и уникальность ее в том, что она объединяет ИИ, приводимый в действие платформой TensorFlow, с мобильным устройством, соединяя мир яркого будущего с миром самого процветающего настоящего.

За последнее десятилетие мы все были свидетелями революции в области смартфонов iOS и Android и испытали ее на себе. И при этом мы находимся лишь в начале революции в области ИИ, которая, вероятно, окажет еще более глубокое влияние на мир вокруг нас. Что может быть лучше, чем тема, которая объединяет лучшее из двух миров, книга, которая показывает, как создавать приложения ИИ с поддержкой TensorFlow на мобильном устройстве, в любое время и в любом месте?

Разумеется, вы можете создавать приложения с элементами ИИ, используя множество существующих облачных API искусственного интеллекта, и иногда это имеет смысл. Вместе с тем запуск приложений с элементами ИИ полностью на мобильных устройствах имеет свои преимущества, которые заключаются в том, что вы можете запускать приложения, даже если отсутствует подключение к сети, когда вы не можете обмениваться данными с облачным сервером либо когда пользователи не хотят отправлять данные со своих мобильных устройств кому-либо еще.

Надо признать также и то, что среди ресурсов проекта TensorFlow с открытым исходным кодом уже есть несколько примеров приложений для iOS и Android, которые могут помочь вам начать работу с мобильной платформой TensorFlow. Однако если вы когда-либо пытались запускать захватывающую модель с поддержкой TensorFlow, которая будет поражать вас своими способностями на своем устройстве iOS или Android, то вы, скорее всего, знаете, что это сопряжено со множеством преград, которые следует преодолеть, прежде чем вы сможете увидеть модель, которая будет успешно работать на вашем мобильном устройстве.

Эта книга поможет сэкономить вам уйму времени и усилий и покажет, как решать все самые распространенные проблемы, с которыми вы можете столкнуться при запуске моделей TensorFlow на мобильном устройстве. В настоящей книге вы увидите, как будут построены с нуля более 10 полноценных программных приложений TensorFlow для iOS и Android, выполняющих множество захватывающих моделей с поддержкой TensorFlow, включая новейшие и самые крутые – модели на основе генеративно-сопоставительной сети (Generative Adversarial Network, GAN) и алгоритма AlphaZero.

Для кого предназначена эта книга

Настоящая книга предназначена для тех читателей, которые являются разработчиками приложений для iOS и/или Android и интересуются построением и вторичной тренировкой других моделей TensorFlow и запуском их в своих мобильных приложениях, а также тех, кто является разработчиком приложений с использованием платформы TensorFlow и хочет запускать новые удивительные модели TensorFlow на мобильных устройствах. Если вы заинтересованы в платформах TensorFlow Lite, Core ML или TensorFlow на компьютере Raspberry Pi, то также извлечете пользу из данной книги.

ЧТО ОХВАТЫВАЕТ ЭТА КНИГА

В главе 1 «Начало работы с платформой TensorFlow Mobile» рассматривается настройка платформы TensorFlow в Mac и Ubuntu, а также графического процессора NVIDIA GPU в Ubuntu. Помимо этого, также рассматривается установка сред разработки Xcode и Android Studio. Мы обсудим разницу между платформами TensorFlow Mobile и TensorFlow Lite и рассмотрим случаи, когда вы должны их использовать. Наконец, мы покажем вам, как запускать примеры приложений для iOS и Android с поддержкой платформы TensorFlow.

В главе 2 «Классифицирование изображений с помощью трансферного обучения» рассматривается понятие трансферного обучения и дается объяснение, почему вы должны его использовать, демонстрируется, как вторично тренировать модели Inception v3 и MobileNet для более точного и быстрого распознавания пород собак и как использовать вторично натренированные модели в примерах приложений для iOS и Android. Затем мы покажем вам, как добавлять поддержку платформы TensorFlow в свое собственное приложение для iOS на языках программирования Objective-C и Swift и в приложение для Android для распознавания пород собак.

В главе 3 «Обнаружение и локализация объектов» дается краткий обзор технологии обнаружения объектов, а затем показывается настройка API Tensorflow обнаружения объектов и его применение для вторичной тренировки моделей SSD-MobileNet и моделей Faster RCNN. Мы также покажем вам, как использовать модели, которые применялись в примере приложения TensorFlow для Android, в приложении для iOS, выполнив сборку библиотеки TensorFlow для iOS вручную с целью поддержки операций TensorFlow, отличных от стандартных. Наконец, мы покажем, как тренировать YOLO2, еще одну популярную модель обнаружения объектов, которая также используется в примере приложения TensorFlow для Android, и как ее использовать в приложении для iOS.

В главе 4 «Трансформирование рисунков с помощью художественных стилей» сначала дается обзор технологии нейронного переноса художественного стиля, получившей ускоренное развитие в последние несколько лет. Затем в ней демонстрируется процедура тренировки модели быстрого нейронного переноса стиля и их применения в приложениях для iOS и Android. После этого мы расскажем, как использовать многостилевую модель TensorFlow Magenta в своих собственных приложениях для iOS и Android, позволяющую легко создавать удивительные художественные стили.

В главе 5 «Понимание простых речевых команд» дается описание технологии распознавания речи и демонстрируется тренировка простой модели распознавания речевых команд. Затем мы покажем вам, как использовать такую модель в Android, а также в iOS с применением языков программирования Objective-C и Swift. Мы также дадим ряд советов в отношении того, как исправлять возможные ошибки загрузки и запуска модели на мобильном устройстве.

В главе 6 «*Описание изображений на естественном языке*» дается описание принципа работы технологии аннотирования изображений. Затем в ней демонстрируется процедура тренировки и заморозки модели аннотирования изображений в TensorFlow. Далее мы обсудим вопрос конвертации и оптимизации многосложной модели с целью ее подготовки к работе на мобильных устройствах. Наконец, мы предложим полноценные приложения для iOS и Android, в которых данная модель используется для генерирования описания изображений на естественном языке.

В главе 7 «*Распознавание рисунков с помощью CNN- и LSTM-сетей*» объясняется принцип работы технологии классифицирования рисунков. Данная глава посвящена подготовке модели, ее тренировке и выведению предсказаний. Затем мы покажем, как создать еще одну пользовательскую библиотеку TensorFlow в iOS для использования данной модели в забавном приложении для iOS, выполняющем распознавание набросков. Наконец, мы покажем вам, как создавать пользовательскую библиотеку TensorFlow для Android, которая позволит исправлять ошибку загрузки новой модели, а затем продемонстрируем использование модели в своем собственном приложении для Android.

В главе 8 «*Предсказание биржевой цены с помощью RNN-сети*» дан пошаговый анализ технологии RNN-сети и продемонстрировано ее применение для предсказания биржевой цены. Затем мы покажем, как строить RNN-модель предсказания биржевой цены с помощью API TensorFlow и как строить LSTM RNN-модель с более простым в использовании API библиотеки Keras для достижения той же самой цели. Мы проведем тестирование этих моделей и посмотрим, смогут ли они превзойти случайную стратегию покупки или продажи. Наконец, мы покажем вам, как запускать модели TensorFlow и Keras в приложениях для iOS и Android.

В главе 9 «*Генерирование и улучшение изображений с помощью GAN-сети*» дается обзор технологии генеративно-сопоставительной сети (GAN-сети) и объясняется, почему она имеет такой большой потенциал. Затем в данной главе будет дано краткое описание процедуры построения и тренировки базовой GAN-модели, которая может использоваться для создания человекоподобных рукописных цифр, а также более продвинутой модели, которая способна улучшать качество изображения с низким разрешением. Наконец, мы расскажем, как использовать обе эти модели на основе GAN-сети в приложениях для iOS и Android.

В главе 10 «*Создание мобильного игрового AlphaZero-подобного приложения*» сначала дается описание работы новейшей и популярной программы AlphaZero, затем демонстрируется процедура тренировки и тестирования AlphaZero-подобной модели для игры в простую, но увлекательную игру под названием «Четыре в ряд» с использованием библиотеки Keras и платформы TensorFlow в качестве бэкенда. Далее мы покажем вам полноценные приложения для iOS и Android, которые позволяют использовать эту модель и играть в игру «Четыре в ряд» на ваших мобильных устройствах.

В главе 11 «*Применение платформ TensorFlow Lite и Core ML на мобильном устройстве*» демонстрируется работа платформы TensorFlow Lite, а затем показывается применение в iOS готовой модели TensorFlow, вторично натренированной модели TensorFlow для платформы TensorFlow Lite и пользовательской модели TensorFlow Lite. Мы также покажем вам, как использовать платформу TensorFlow Lite в Android. После этого мы дадим краткий обзор платформы Core ML компании Apple и покажем, как использовать платформу Core ML со стандартным машинным обучением на основе библиотеки Scikit-Learn. Наконец, мы рассмотрим пример использования платформы Core ML с платформой TensorFlow и библиотекой Keras.

В главе 12 «*Разработка приложений TensorFlow на компьютере Raspberry Pi*» сначала рассматривается настройка компьютера Raspberry Pi и приведение его в движение, а также настройка платформы TensorFlow на компьютере Raspberry Pi. Затем мы расскажем, как использовать модели TensorFlow распознавания изображений и звука, а также API речевого воспроизведения текста (TTS) и API движения робота для создания робота Raspberry Pi, который может двигаться, видеть, слушать и говорить. Наконец, мы подробно обсудим вопрос использования библиотеки OpenAI Gym и платформы TensorFlow для создания и тренировки с нуля мощной нейросетевой модели линии поведения на основе самообучения с подкреплением в симулируемой среде, позволяющей роботу учиться сохранять равновесие.

КАК ПОЛУЧИТЬ МАКСИМАЛЬНУЮ ОТДАЧУ ОТ ЭТОЙ КНИГИ

Мы рекомендуем начать с чтения первых четырех глав по порядку, а также с запуска сопутствующих приложений для iOS и Android, доступных из репозитория исходного кода книги по адресу <http://github.com/jeffxtang/mobiletfbook>. Они позволят вам обеспечить наличие сред разработки, настроенных для разработки мобильных приложений с поддержкой TensorFlow, и сведений о том, как интегрировать TensorFlow в свои собственные приложения для iOS и/или Android. Если вы являетесь разработчиком iOS, то также узнаете, как использовать языки программирования Objective-C или Swift с платформой TensorFlow, когда и как использовать модуль TensorFlow либо пользовательскую библиотеку TensorFlow для iOS ручной сборки.

Затем, если вам необходимо выполнить ручную сборку пользовательской библиотеки TensorFlow для Android, перейдите к главе 7 «*Распознавание рисунков с помощью CNN- и LSTM-сетей*», и если вы хотите узнать, как использовать в своем мобильном приложении модель Keras, то обратитесь к главе 8 «*Предсказание биржевой цены с помощью RNN-сети*» и главе 10 «*Создание мобильного игрового AlphaZero-подобного приложения*».

Если вы больше заинтересованы в платформе TensorFlow Lite или Core ML, то прочтите главу 11 «*Применение платформ TensorFlow Lite и Core ML на мобильном устройстве*», а если же вы больше всего заинтересованы в платформе

TensorFlow на компьютере Raspberry Pi или реализацией самообучения с подкреплением в платформе TensorFlow, то перейдите к главе 12 «*Разработка приложений TensorFlow на компьютере Raspberry Pi*».

Кроме этого, прежде чем сразу переходить к подробному изучению реализаций моделей, вы можете просмотреть главы 5–10, чтобы узнать, как выполнять тренировку различных видов моделей – на основе CNN, RNN, LSTM, GAN и AlphaZero – и как их использовать на мобильных устройствах, возможно, запуская приложения для iOS и/или Android, приводимые в каждой главе. Кроме того, вы можете перейти непосредственно к любой главе с интересующей вас моделью; просто имейте в виду, что более поздняя глава может ссылаться на более раннюю главу относительно повторяющихся подробностей, таких как шаги добавления в свое приложение пользовательской библиотеки TensorFlow для iOS или исправление некоторых ошибок загрузки либо запуска модели путем ручной сборки пользовательской библиотеки TensorFlow. Так или иначе, будьте уверены, что вы не потеряетесь. По крайней мере, мы сделали все возможное, чтобы обеспечить вам удобное пошаговое руководство с периодическими ссылками на некоторые шаги из предыдущих инструкций, с тем чтобы помочь вам избежать всевозможных ловушек, с которыми вы можете столкнуться при создании мобильных приложений с поддержкой платформы TensorFlow, а также просто чтобы избежать повторов.

Когда читать данную книгу

В последние годы ИИ, точнее его самая популярная область – машинное обучение, а если еще точнее, его самая популярная подобласть – глубокое обучение, получили ускоренное развитие. Новые выпуски платформы TensorFlow, поддерживаемые компанией Google и самым популярным сообществом разработчиков на данной платформе из всех платформ машинного обучения с открытым исходным кодом, также выходили с более высокой скоростью. Когда мы приступили к написанию данной книги в декабре 2017 года, последний релиз платформы TensorFlow был 1.4.0, выпущенный 2 ноября 2017 года, и после этого версия 1.5.0 была выпущена 26 января 2018 года, версия 1.6.0 – 28 февраля 2018 года, версия 1.7.0 – 29 марта 2018 года, и версия 1.8.0 – 27 апреля 2018 года. Весь приводимый в книге программный код для iOS и Android и на языке Python был протестирован со всеми этими версиями платформы TensorFlow. Добавим, что к тому времени, когда вы будете читать эту книгу, последней версией TensorFlow, вероятно, будет уже версия выше 1.8.0.

Как оказалось, вам не нужно особо переживать по поводу частого выхода новых версий платформы TensorFlow; приводимый в книге программный код, скорее всего, будет легко работать в последних версиях данной платформы. Во время тестирования нами программных приложений на платформах TensorFlow 1.4, 1.5, 1.6, 1.7 и 1.8 мы не вносили никаких изменений в про-

граммный код. Вполне вероятно, что в более поздней версии по умолчанию будет поддерживаться еще больше операций платформы TensorFlow, и поэтому вам не придется выполнять ручную сборку пользовательской библиотеки TensorFlow, либо вы сможете выполнить ручную сборку пользовательской библиотеки TensorFlow более простым способом.

Конечно, нет никакой гарантии, что в будущих версиях платформы TensorFlow весь программный код будет работать без каких-либо изменений, но с учетом всех подробных инструкций и советов по устранению неполадок, изложенных в данной книге, независимо от того, когда вы читаете эту книгу, сейчас или по прошествии нескольких месяцев, у вас не должно возникнуть шероховатостей при ее чтении и запуске приводимых в ней приложений с помощью платформы TensorFlow 1.4–1.8 или более поздней версии.



Несмотря на то что в целях публикации данной книги мы в определенный момент вынуждены были остановиться на конкретной версии платформы TensorFlow, мы продолжим выполнять тестовый прогон всего программного кода книги после выхода каждой новой крупной версии платформы TensorFlow, а также соответствующим образом обновлять программный код и результаты тестов в репозитории исходного кода книги по адресу <http://github.com/jeffxtang/mobiletfbook>. Если у вас возникли вопросы по поводу программного кода или данной книги, то вы также можете отправить вопрос непосредственно в репозиторий.

Еще одной проблемой является выбор между платформами TensorFlow Mobile и TensorFlow Lite. В большинстве глав данной книги обсуждается использование платформы TensorFlow Mobile (главы с 1 по 10). Может быть, в будущем платформа TensorFlow Lite станет мобильной альтернативой настольной платформе TensorFlow, но пока, согласно конференции разработчиков Google I/O 2018, эта платформа находится в предварительной версии – именно поэтому компания Google ожидает, что вы будете «применять платформу TensorFlow Mobile для большинства производственных задач». Даже после того, как платформа TensorFlow Lite будет выпущена официально, по данным компании Google, «в обозримом будущем платформа TensorFlow Mobile не собирается уходить со сцены» – на самом деле с учетом последней версии TensorFlow 1.8.0, которую мы протестировали до публикации книги, мы обнаружили, что применение платформы TensorFlow Mobile стало еще проще.

Если же, наконец, наступит тот день, когда платформа TensorFlow Lite полностью заменит собой платформу TensorFlow Mobile во всех случаях использования, располагая большей производительностью и меньшим размером, то навыки, которые вы приобретете из этой книги, только лучше подготовят вас к этому дню. В то же время, прежде чем это непредвиденное будущее наступит, вы имеете возможность прочитать книгу и узнать, как применять ее более солидный аналог, платформу TensorFlow Mobile, для запуска всех этих удивительных и мощных моделей с поддержкой TensorFlow в мобильных приложениях.

СКАЧИВАНИЕ ФАЙЛОВ С ПРИМЕРАМИ КОДА

Файлы с примерами можно скачать с вашего аккаунта по адресу <http://www.packtpub.com/> для всех книг издательства Packt Publishing, которые вы приобрели. Если вы купили эту книгу в другом месте, то можно посетить <http://www.packtpub.com/support> и зарегистрироваться там, чтобы получить файлы прямо по электронной почте.

Скачать файлы с примерами можно, выполнив следующие шаги.

1. Войдите на наш веб-сайт или зарегистрируйтесь там, используя ваш адрес электронной почты и пароль.
2. Наведите указатель мыши на вкладку SUPPORT вверху страницы.
3. Щелкните по разделу Code Downloads & Errata, посвященному примерам программного кода и опечаткам.
4. Введите название книги в поле поиска.

Скачав файл, пожалуйста, убедитесь, что вы распаковали или извлекли папку, воспользовавшись последней версией указанных ниже архиваторов:

- WinRAR / 7-Zip для Windows;
- Zipeg / iZip / UnRarX для Mac OS;
- 7-Zip / PeaZip для Linux.

Помимо этого, комплект примеров программного кода, прилагаемый к данной книге, размещен на GitHub в разделе Packt по адресу <https://github.com/PacktPublishing/Intelligent-Mobile-Projects-with-TensorFlow>. В случае обновления программного кода он будет обновлен в существующем репозитории GitHub. Мы также располагаем другими комплектами примеров программного кода, которые можно выбрать из нашего богатого каталога книг и видеороликов, предлагаемого на странице <https://github.com/PacktPublishing/>. Можете убедиться сами!

УСЛОВНЫЕ ОБОЗНАЧЕНИЯ

В этой книге используется ряд условных обозначений.

КодВТексте: указывает кодовые слова в тексте, имена таблиц базы данных, имена папок, имена файлов, расширения файлов, имена путей, фиктивные URL-адреса, ввод данных пользователем и дескрипторы Twitter. Вот пример: «Установить библиотеки `matplotlib`, `pillow`, `lxml` и `ipython` в Ubuntu или Mac, которые вы можете запустить».

Блок кода выглядит следующим образом:


```
syntax = "proto2";
package object_detection.protos;
message StringIntLabelMapItem {
  optional string name = 1;
  optional int32 id = 2;
  optional string display_name = 3;
};
```


```
message StringIntLabelMap {  
  repeated StringIntLabelMapItem item = 1;  
};
```

Любой ввод или вывод из командной строки записывается следующим образом:

```
sudo pip install pillow  
sudo pip install lxml  
sudo pip install jupyter  
sudo pip install matplotlib
```

Полужирный шрифт: обозначает новый термин, важное слово или слова, отображаемые на экране. Например, слова в меню или диалоговых окнах отображаются в тексте следующим образом. Вот пример: «А теперь давайте выберем опцию **«Улучшить изображение»**, и вы увидите результат».

 Данный элемент обозначает предупреждение или предостережение.

 Данный элемент обозначает подсказку или совет.

КОММЕНТАРИИ ПЕРЕВОДЧИКА

В центре внимания машинного обучения и его подобласти, глубокого обучения, находится обучающаяся система, то есть система, способная с течением времени приобретать новые знания и улучшать свою работу, используя поступающую информацию¹. В зарубежной специализированной литературе для *передачи* знаний и *приобретения* знаний существуют отдельные термины – *train* (*натренировать*, обучить) и *learn* (*выучить*, обучиться). Приведенное ниже предложение из главы 2 настоящей книги четко это демонстрирует:

It'd take us many weeks of training of a modern CNN from scratch to learn all the weights.

Нам пришлось бы тренировать современную CNN-сеть на протяжении многих недель с нуля, пока она не заучит все веса,

где *training* (тренировка) – это работа, которую выполняет исследователь-проектировщик для получения обучившейся модели, в основе которой лежит обучающийся алгоритм, по сути искатель минимумов (или максимумов) для надлежащим образом сформулированных функций, а *learning* (самообучение, заучивание) – это работа, которую выполняет алгоритм-ученик по приобретению новых знаний или изменению и закреплению существующих знаний и по-

¹ См. https://ru.wikipedia.org/wiki/Обучающаяся_система, а также <https://bigenc.ru/mathematics/text/1810335>. – *Прим. перев.*

ведения¹. Когда же в русской спецлитературе используется термин «обучение», то он несет в себе двусмысленность, потому что под ним может подразумеваться и передача знаний, и получение знаний одновременно, как, например, в случае с термином «машинное обучение», который может означать и тренировку алгоритмических машин, и способность таких машин обучаться, что нередко вносит путаницу и терминологический разброд в переводной литературе при решении дилеммы «training-learning», в то время как появление в зарубежной технической литературе термина learning в любом виде подразумевает исключительно второе – *самообучение, заучивание* алгоритмом весов и других параметров. Отсюда вытекает одно важное следствие: английский термин machine learning обозначает *приобретение знаний алгоритмической машиной*, а следовательно, более соответствовать оригиналу будет термин *машинное самообучение или автоматическое обучение*. Весомым аргументом в пользу этого термина является и то, что с начала 60-х и до середины 80-х годов XX столетия в ходу был похожий термин – «обучающиеся машины». Проблематика обучающихся и самопроизводящихся машин изучалась в работах А. Тьюринга «Может ли машина мыслить?», 1960 (обучающиеся машины), К. Шеннона «Работы по теории информации и кибернетике» (самовоспроизводящиеся машины), Н. Винера «Кибернетика, или Управление и связь в животном и машине», 1961, Н. Нильсона «Обучающиеся машины», 1974, и Цыпкина Я.З. «Основы теории обучающихся систем», 1970.

В настоящем переводе далее за основу принят зарубежный подход, который неизбежно привел к некоторой корректировке терминологии. Применяемые в машинном обучении и глубоком обучении алгоритмы, модели и системы переведены как *обучающиеся, машинно-обучающиеся и глубоко обучающиеся*. То есть акцент делается не на классификации алгоритма в соответствующей иерархии, а на его характерном свойстве. Далее, методы, которые реализуются в обучающихся алгоритмах, переведены как *методы самообучения* (ср. методы обучения). Как известно, эти методы делятся на три широкие категории. Следуя принципу бритвы Оккама, они переведены как методы контролируемого самообучения (ср. обучение с учителем), методы неконтролируемого самообучения (ср. обучение без учителя) и методы самообучения на основе максимизации вознаграждения (ср. обучение с подкреплением). Последний термин обусловлен тем, что в его основе лежит алгоритм, который «учится максимизировать некое понятие вознаграждения», получаемого за правильно выполненное действие². Среди многих гиперпараметров, которые позволяют настроить работу обучающегося алгоритма, имеется rate of learning, который переведен как *скорость заучивания* (ср. темп обучения).

¹ См. <http://www.basicknowledge101.com/subjects/learningstyles.html#diy>. – *Прим. перев.*

² См. https://en.wikipedia.org/wiki/Reinforcement_learning. – *Прим. перев.*

Глава 1

Начало работы с платформой TensorFlow Mobile

Данная глава посвящена настройке среды разработки для создания всех тех приложений для iOS или Android с поддержкой TensorFlow, которые рассматриваются в остальной части книги. Мы не будем подробно обсуждать все поддерживаемые версии платформы TensorFlow, версии ОС, версии сред Xcode и Android Studio, которые могут использоваться для разработки, поскольку такую информацию можно легко найти на веб-сайте TensorFlow (<http://www.tensorflow.org>) или в Google. Вместо этого в данной главе мы кратко расскажем о примерах рабочих сред, которые позволят нам быстро углубиться в изучение и разработку всех этих удивительных приложений.

Если у вас уже установлена платформа TensorFlow, а также среды разработки Xcode и Android Studio, и вы можете запускать и тестировать примеры приложений TensorFlow для iOS и Android, и если у вас уже установлен графический процессор NVIDIA GPU, предназначенный для более быстрой тренировки глубоко обучающихся моделей, то вы можете пропустить эту главу либо перейти непосредственно к незнакомому разделу.

В данной главе мы рассмотрим следующие темы (настройка среды разработки в компьютере Raspberry Pi будет обсуждаться в главе 12 «*Разработка приложений TensorFlow на компьютере Raspberry Pi*»):

- настройка платформы TensorFlow;
- настройка среды разработки Xcode;
- настройка среды разработки Android Studio;
- TensorFlow Mobile против TensorFlow Lite;
- запуск примеров приложений TensorFlow для iOS;
- запуск примеров приложений TensorFlow для Android.

НАСТРОЙКА ПЛАТФОРМЫ TENSORFLOW

Платформа TensorFlow является ведущей платформой с открытым исходным кодом для машинного интеллекта. Когда компания Google выпустила

Конец ознакомительного фрагмента.

Приобрести книгу можно

в интернет-магазине

«Электронный универс»

e-Univers.ru