Оглавление

Содержание

Вводное слово

Считается, что система глубокого обучения является эффективной, если она способна наводить мосты между двумя разными мирами – научными изысканиями / прототипированием и производственными операциями. Разрабатывающие такие системы коллективы должны уметь общаться с практиками из этих двух миров и работать с разными наборами технических требований и ограничений, которые исходят от каждого из них. В силу этого требуется безупречное понимание конструкционных особенностей компонентов в системах глубокого обучения и принципов их работы в тандеме. Этому аспекту инженерии глубокого обучения посвящено очень мало существующей литературы. Данный информационный пробел становится проблемой, когда технология глубокого обучения внедряется среди младших инженеров программного обеспечения и от них ожидается, что они станут эффективными инженерами в данной сфере.

На протяжении многих лет инженерно-конструкторские коллективы заполняли этот пробел, используя свой приобретенный опыт и извлекая то, что им нужно знать, из литературы. Их работа помогала традиционным инженерам программного обеспечения разрабатывать, конструировать и расширять системы глубокого обучения за относительно короткий промежуток времени. Поэтому я с большим волнением узнал, что Кай и Дональд, оба из которых возглавляли коллективы инженеров глубокого обучения, выступили с очень важной инициативой консолидировать эти знания и поделиться ими в форме книги.

Нам давно пора выпустить всеобъемлющую книгу о разработке систем, которые поддерживают наведение мостов в области глубокого обучения, от научных изысканий и прототипирования до производства. Книга «Разработка систем глубокого обучения», наконец, охватывает эту потребность.

Данная книга начинается с высокоуровневого введения, описывающего суть системы глубокого обучения и ее функциональности.

В последующих главах каждый компонент системы обсуждается подробно, приводится мотивация и дается представление о плюсах и минусах различных конструкционных вариантов.

Каждая глава заканчивается анализом, который помогает читателям оценить варианты, наиболее подходящие для их собственных вариантов использования. В заключение авторы, опираясь на все предыдущие главы, подробно рассказывают о сложном пути перехода от научных изысканий и прототипирования к производству. И для того чтобы помочь инженерам воплотить все эти идеи на практике, они создали образец системы глубокого обучения с полностью работоспособным исходным кодом, дабы проиллюстрировать ключевые концепции и предложить попробовать ее тем, кто только начинает работать в этой области.

В целом читатели найдут, что эту книгу легко читать и по ней легко перемещаться, а их понимание способов организации, конструирования и реализации систем глубокого обучения поднимется на совершенно новый уровень. Практики всех уровней знаний, заинтересованные в разработке эффективных систем глубокого обучения, по достоинству оценят эту книгу как бесценный ресурс и справочную информацию. Они прочтут ее один раз, чтобы получить общую картину, а затем будут возвращаться к ней снова и снова при разработке своих собственных систем, конструировании компонентов и принятии важных конструкционных решений, удовлетворяющих все коллективы, которые используют эти системы.

– *Сильвио Саварезе*, вице-президент,
главный научный сотрудник Salesforce

– *Каймин Сюн*, вице-президент Salesforce

Предисловие

Чуть более десяти лет назад нам посчастливилось разработать несколько первых продуктовых функциональностей, ориентированных на конечного пользователя, которые были основаны на искусственном интеллекте. Это было грандиозное предприятие. Сбор и организация данных, которые были пригодны для тренировки моделей, в то время не были обычной практикой. Несколько алгоритмов машинного обучения были упакованы в виде готовых к использованию библиотек. Проведение экспериментов требовало ручного управления и разработки конкретно-прикладных рабочих процессов и визуализаций. Для раздачи каждого типа моделей были созданы конкретно-прикладные серверы. За исключением ресурсоемких технологических компаний, почти каждая новая продуктовая функциональность на базе искусственного интеллекта создавалась с нуля. Это была далеко идущая мечта о том, что интеллектуальные приложения однажды станут товаром.

Поработав с несколькими приложениями ИИ, мы поняли, что всякий раз повторяли один и тот же ритуал, и нам показалось, что имеет смысл разработать системный подход с прототипированием для внедрения продуктовых функциональностей на базе ИИ в производство. Результатом этих усилий стал комплект фреймворкового программного обеспечения с открытым исходным кодом под названием PredictionIO, который соединяет в себе самые передовые программные компоненты для сбора и извлечения данных, тренировки и раздачи моделей. Полностью адаптируемый под конкретно-прикладные задачи с помощью API-интерфейсов и пригодный для развертывания в виде служб всего несколькими командами, он помог сократить время, необходимое на каждом этапе – от проведения экспериментов по обработке данных до тренировки и развертывания готовых к производству моделей. Мы были очень рады узнать, что разработчики по всему миру смогли использовать продукт PredictionIO для создания своих собственных приложений на базе искусственного интеллекта,

что привело к поразительному росту их бизнеса. Позже стартап PredictionIO был приобретен компанией Salesforce для решения аналогичной задачи в еще большем масштабе.

К тому времени, когда мы решили написать эту книгу, индустрия уже процветала благодаря здоровой экосистеме программного обеспечения для искусственного интеллекта. Стало доступно множество алгоритмов и инструментов для решения разных задач. Некоторые поставщики облачных технологий, такие как Amazon, Google и Microsoft, даже предоставляют законченные развернутые системы, которые позволяют коллективам сотрудничать в проведении экспериментов, прототипировании и развертывании производственных приложений в одном централизованном месте. Независимо от вашей цели теперь у вас есть множество вариантов и множество способов их достижения.

Тем не менее, поскольку мы работаем с коллективами над внедрением продуктовых функциональностей глубокого обучения, возникают повторяющиеся вопросы. Почему наша система глубокого обучения сконструирована именно так, а не иначе? Подойдет ли ее конструкция и для других конкретных случаев использования? Мы заметили, что эти вопросы чаще всего задают младшие инженеры программного обеспечения, и мы проинтервьюировали некоторых из них, чтобы выяснить причину. В результате обнаружилось, что их традиционная ученая программа в области разработки программного обеспечения не подготовила их к эффективной работе с системами глубокого обучения. И когда они искали учебные ресурсы, они находили лишь скудную и разрозненную информацию о конкретных системных компонентах, и почти ни в одном ресурсе не обсуждались основы программных компонентов, причины, по которым они были собраны таким образом, и характер их работы вместе, образуя целостную систему.

В целях решения этой проблемы мы начали разрабатывать базу знаний, которая в конечном итоге превратилась в похожий на руководство учебный материал, объясняющий принципы конструирования каждого компонента системы, плюсы и минусы конструкционных решений, а также обоснование как с технической, так и с продуктовой точки зрения. Мы узнали, что наш материал помогал быстро набирать новых сотрудников в коллективы разработчиков и позволял традиционным инженерам программного обеспечения, не имеющим предшествующего опыта в разработке систем глубокого обучения, входить в курс дела. И решили поделиться этим учебным материалом с гораздо большей аудиторией в формате книги. Мы связались с издательством Manning, и остальное стало историей.

Отзывы и пожелания

Мы всегда рады отзывам наших читателей. Расскажите нам, что вы думаете об этой книге – что понравилось или, может быть, не по-

нравилось. Отзывы важны для нас, чтобы выпускать книги, которые будут для вас максимально полезны.

Вы можете написать отзыв на нашем сайте www.dmkpress.com, зайдя на страницу книги и оставив комментарий в разделе «Отзывы и рецензии». Также можно послать письмо главному редактору по адресу dmkpress@gmail.com; при этом укажите название книги в теме письма.

Если вы являетесь экспертом в какой-либо области и заинтересованы в написании новой книги, заполните форму на нашем сайте по адресу http://dmkpress.com/authors/publish_book/ или напишите в издательство по адресу dmkpress@gmail.com.

Список опечаток

Хотя мы приняли все возможные меры для того, чтобы обеспечить высокое качество наших текстов, ошибки все равно случаются. Если вы найдете ошибку в одной из наших книг, мы будем очень благодарны, если вы сообщите о ней главному редактору по адресу dmkpress@gmail.com. Сделав это, вы избавите других читателей от недопонимания и поможете нам улучшить последующие издания этой книги.

Нарушение авторских прав

Пиратство в интернете по-прежнему остается насущной проблемой. Издательства «ДМК Пресс» и Manning Publications очень серьезно относятся к вопросам защиты авторских прав и лицензирования. Если вы столкнетесь в интернете с незаконной публикацией какой-либо из наших книг, пожалуйста, пришлите нам ссылку на интернет-ресурс, чтобы мы могли применить санкции.

Ссылку на подозрительные материалы можно прислать по адресу электронной почты dmkpress@gmail.com.

Мы высоко ценим любую помощь по защите наших авторов, благодаря которой мы можем предоставлять вам качественные материалы.

Благодарности

Написание книги требует по-настоящему больших персональных усилий, но эта книга была бы невозможна без помощи следующих ниже людей.

В основу значительной части данной книги легла работа с разными коллективами из Salesforce Einstein groups (платформа Einstein, E.ai, Hawking). К этим блестящим и влиятельным коллегам относятся (в алфавитном порядке) Сара Ашер, Джимми Ау, Джон Болл, Аня Бида, Джин Беккер, Ятиш Бхагаван, Джексон Чанг, Химай Десаи, Мехмет Эзбидерли, Виталий Гордон, Индира Айер, Арпит Кейл, Шрирам Кришнан, Энни Лэнг, Чан Ли, Эли Левин, Дафна Лю, Лия Макгвайр, Ивайло Михов, Ричард Пак, Генри Сапутра, Рагу Сетти, Шон Сенекал, Карл Скуча, Магнус Торн, Тед Таттл, Иэн Варли, Ян Ян, Марцин Земински и Лео Чжу.

Мы также хотим воспользоваться этой возможностью, чтобы поблагодарить нашего редактора по разработке Фрэнсис Лефковиц. Она не только отличный редактор, который дает потрясающие рекомендации по написанию и прямому редактированию, но и является замечательной наставницей, которая руководила нами на протяжении всего процесса написания книги. Без нее эта книга не была бы такого качества, как сейчас, и не была бы завершена так, как планировалось.

Мы выражаем благодарность коллективу издательства Manning за их руководство на протяжении всего процесса написания книги. Мы по-настоящему ценим возможность узнать мнение читателей на ранних стадиях написания книги благодаря используемой в Manning программе раннего доступа (MEAP).

Всем рецензентам: Алексу Бланку, Амиту Кумару, Аюшу Томару, Бхагвану Коммади, Динкару Джуялу, Эсрефу Дурне, Гаураву Суду, Гийому Аллеону, Хаммаду Аршаду, Джейми Шафферу, Джапнит Сингх, Джереми Чену, Жуану Динису Феррейре, Кате Паткин, Киту Киму, Ларри Каю, Марии Ане, Микаэлю Дотри, Ник Декруз, Николь

Кенигштейн, Ной Флинн, Оливеру Кортен, Омару Эль Малаку, Пранджалу Ранджану, Рави Суреш Машру, Саиду Эч-Чади, Сандипу Д., Санкету Шарма, Сатеджу Кумару Саху, Саяку Полу, Швете Джоши, Симоне Сгуацца, Шрираму Мачарлу, Сумиту Бхаттачарья, Урсину Стауссу, Видхья Вина и Вэ Луо – ваши предложения помогли сделать эту книгу лучше.

Я хотел бы поблагодарить мою жену Пей Ву за ее безграничную любовь и огромную поддержку на протяжении всего процесса написания этой книги. В трудные времена пандемии Covid Пей оставалась мирным уголком, что позволило писать книгу в окружении оживленной семьи с двумя очаровательными малышами – Кэтрин и Тяньченгом.

Я также хотел бы выразить свою благодарность Ян Сюэ, талантливому разработчику 10X, который написал почти всю лабораторию исходного кода. Его помощь делает ее не только высококачественной, но и простой в освоении. Жена Яна, Донг, всем сердцем его поддерживала, чтобы Ян смог сосредоточиться на лаборатории книги.

Еще один человек, которого я хочу поблагодарить, – Диана Сиболд, талантливый и опытный технический специалист Salesforce. Диана вдохновила меня своим собственным писательским опытом и побудила меня начать писать.

– Кай Ван

Один из основателей стартапа PredictionIO (позже приобретенного компанией Salesforce) научил меня бесценным навыкам разработки продуктов с открытым исходным кодом для разработчиков машинного обучения. Это авантюрное и полезное путешествие было бы невозможным без мужественных душ, которые безмерно доверяли друг другу. Это (в алфавитном порядке) Кеннет Чан, Том Чан, Пэт Феррел, Изабель Ли, Пол Ли, Алекс Мерритт, Томас Стоун, Марко Виверо и Джастин Ип.

Саймон Чан заслуживает особого упоминания. Чан был соучредителем стартапа PredictionIO, и я также имел честь работать с ним и учиться у него в его предыдущих предпринимательских начинаниях. Он был первым человеком, который официально познакомил меня с программированием, когда мы оба учились в одной школе (колледж Ва Янь, Коу-Лун) в Гонконге. Среди других вдохновляющих деятелей школы (в алфавитном порядке) были Дональд Чан, Джейсон Чан, Гамлет Чу, Ка Куен Фу, Джеффри Хау, Фрэнсис Конг, Эрик Лау, Кам Лау, Рэйлекс Ли, Кевин Лей, Дэнни Шинг, Норман Со, Стивен Танг и Лобо Вонг.

Я чрезвычайно благодарен своим родителям и моему брату Рональду. Они помогли мне познакомиться с компьютерами в раннем возрасте. Их постоянная поддержка сыграла жизненно важную роль в годы моего становления, когда я стремился стать инженером-компьютерщиком.

Мой сын Спенсер является ходячим доказательством того, почему биологические глубокие нейронные сети – самые удивительные вещи в мире. Он – чудесный дар, который каждый день показывает мне, что я всегда могу расти и становиться лучше.

Словами не передать, как много значит для меня моя жена Вики. Она всегда способна пробудить во мне все самое лучшее, чтобы в трудные моменты я мог продолжать двигаться вперед. Она – лучшая компаньонка, о которой я когда-либо мог мечтать.

– Дональд Сзето

О книге

Данная книга призвана оснастить инженеров знаниями о том, как конструировать, строить или организовывать эффективные системы машинного обучения и адаптировать эти системы под любые потребности и ситуации, с которыми они могут столкнуться. Разрабатываемые ими системы будут облегчать, автоматизировать и ускорять разработку проектов машинного обучения (в частности, проектов глубокого обучения) в различных областях.

В области глубокого обучения именно модели привлекают все внимание. Возможно, это справедливо, если учесть, что на рынок регулярно поступают новые приложения, разработанные на основе этих моделей, – приложения, приводящие потребителей в восторг, такие как камеры слежения, распознающие человека, виртуальные персонажи в интернет-видеоиграх, которые ведут себя как настоящие люди, программа, которая может писать исходный код решения поставленных перед ней произвольных задач, а также передовые системы помощи водителю, которые в один прекрасный день приведут к полностью автономным и самоуправляемым автомобилям. За очень короткий промежуток времени область глубокого обучения наполнилась огромным волнением и многообещающим потенциалом, ожидающим полного воплощения.

Но модель действует не в одиночку. Материализация продукта или службы предусматривает, что модель будет располагаться в системе или на платформе (мы используем эти термины взаимозаменяемо), которая поддерживает модель различными службами и хранилищами. Для этого, среди прочего, нужны, например, API, менеджер наборов данных и склад артифактов[1] и метаданных. Таким образом, за каждым коллективом разработчиков моделей глубокого обучения

[1] Арт*и*факт – это продукт деятельности вообще, в данном случае результат работы системы или модели, тогда как арт*е*факт – это произведение искусства или изделие художественного творчества. – *Прим. перев.*

стоит коллектив разработчиков, в чьи обязанности входит не само глубокое обучение, а создание инфраструктуры, которая содержит модель и все остальные компоненты.

Мы заметили в отрасли проблему, которая заключается в том, что нередко разработчики, которым поручено разрабатывать систему и компоненты глубокого обучения, обладают лишь поверхностными знаниями о глубоком обучении. Они не понимают конкретных требований, предъявляемых к инженерно-конструкторской работе над системой в глубоком обучении, поэтому при разработке системы они склонны следовать обобщенным подходам. Например, они могут принять решение передать всю работу, связанную с разработкой модели глубокого обучения, исследователю данных и сосредоточиться только на автоматизации. И поэтому разрабатываемая ими система будет опираться на традиционную систему планирования списков заданий или бизнес-аналитическую систему анализа данных, которая не оптимизирована ни под характер выполнения заданий глубокого обучения, ни под специфичные для глубокого обучения шаблоны доступа к данным. В результате систему будет сложно использовать для разработки моделей, а скорость доставки моделей останется низкой. По сути, инженеров, которым не хватает глубокого понимания глубокого обучения, просят разрабатывать системы с поддержкой моделей глубокого обучения. Как следствие эти инженерные системы неэффективны и не подходят для систем глубокого обучения.

О разработке моделей глубокого обучения было написано немало с точки зрения исследователя данных, охватывая сбор данных и насыщение наборов данных, написание алгоритмов тренировки и т. п. Но очень немногие книги или даже блоги были посвящены системе и службам, которые поддерживают все эти мероприятия по глубокому обучению.

В данной книге мы обсуждаем строительство и конструирование систем глубокого обучения с точки зрения разработчика программного обеспечения. Наш подход заключается в том, чтобы сначала описать типичную систему глубокого обучения в целом, включая ее главнейшие компоненты и их взаимосвязь; затем в отдельной главе мы углубляемся в каждый главнейший компонент. Мы начинаем каждую главу о компонентах с изложения технических требований. Затем знакомим с принципами конструирования, а также образцами служб / исходного кода и, наконец, оцениваем технологические решения с открытым исходным кодом.

Поскольку мы не можем охватить в книге все существующие системы глубокого обучения (от производителя или с открытым исходным кодом), то сосредоточиваемся на изложении технических требований и принципов конструирования (с примерами). Изучив эти принципы, попробовав примеры служб из книги и прочитав наше обсуждение вариантов с открытым исходным кодом, мы надеемся, что читатели смогут провести собственное исследование и найти то, которое подходит им лучше всего.

Кому следует прочитать эту книгу?

Первостепенная аудитория этой книги – инженеры программного обеспечения (включая недавно окончивших учебу студентов со специализацией в области вычислительных наук), которые хотят быстро перейти к разработке систем глубокого обучения, например те, кто хочет работать на платформах глубокого обучения или интегрировать какую-либо функциональность искусственного интеллекта – например, раздачу моделей – в свои продукты.

Исследователи данных, инженеры-изыскатели, менеджеры и все остальные, кто использует машинное обучение для решения реальных задач, также найдут эту книгу полезной. Разобравшись в опорной инфраструктуре (или системе), они будут оснащены всем необходимым для предоставления точной обратной связи коллективу инженеров в том, что касается повышения эффективности процесса разработки моделей.

Это инженерная книга, и вам не нужны знания в области машинного обучения, но вы должны быть знакомы с базовыми концепциями вычислительных наук и инструментами программирования, такими как микросервисы, gRPC и Docker, чтобы работать с лабораторией исходного кода и понимать технический материал. Независимо от вашего образования, вы все равно сможете извлечь пользу из нетехнических материалов книги, которые помогут вам лучше понять принципы работы машинного обучения и систем глубокого обучения, для того чтобы переносить продукты и службы из области идей в производство.

Прочитав эту книгу, вы сможете понять механизмы работы систем глубокого обучения и способы разработки каждого компонента. Вы также поймете ситуации, когда следует собирать технические требования пользователей, транслировать их в конструкционные варианты системных компонентов и интегрировать компоненты с целью создания целостной системы, которая поможет вашим пользователям быстро разрабатывать и предоставлять функциональные возможности глубокого обучения.

Как эта книга организована: дорожная карта

В этой книге 10 глав и три дополнения (включая одно лабораторное дополнение). В первой главе объясняется, что такое цикл освоения глубокого обучения и как выглядит базовая система глубокого обучения. В следующих главах мы подробно рассмотрим каждый функциональный компонент эталонной системы глубокого обучения. Наконец, в последней главе обсуждается вопрос о том, как модели отправляются в производство. Дополнение к книге содержит лабораторию исходного кода, позволяющую читателям опробовать образец системы глубокого обучения.

В главе 1 описывается, что такое система глубокого обучения, разные участвующие в разработке системы интересанты и как они

взаимодействуют с ней в целях ее оснащения функциональными возможностями глубокого обучения. Это взаимодействие называется циклом освоения глубокого обучения. Вдобавок вы сформируете для себя концепцию системы глубокого обучения, именуемую эталонной системной архитектурой, которая содержит все необходимые элементы и может быть адаптирована в соответствии с вашими требованиями.

Главы со 2 по 9 охватывают каждый стержневой компонент эталонной архитектуры системы глубокого обучения, такой как служба управления наборами данных, служба тренировки моделей, служба автоматической гиперпараметрической оптимизации и служба оркестровки рабочих процессов.

В главе 10 обсуждается вопрос о том, как подготовить конечный продукт в фазе научного изыскания или прототипирования к выпуску в публичное пространство. В дополнении A представлен образец системы глубокого обучения и демонстрируется лабораторное упражнение, в дополнении B приводится обзор существующих технологических решений, а в дополнении C обсуждается система Kubeflow Katib.

Об исходном коде

Мы считаем, что самый лучший способ учиться – это делать, практиковаться и экспериментировать. В целях демонстрации описанных в этой книге принципов конструирования и получения практического опыта мы создали образец системы глубокого обучения и лабораторию исходного кода. Весь исходный код, инструкции по настройке и лабораторные скрипты образца системы глубокого обучения доступны на GitHub (https://github.com/orca3/MiniAutoML). Вы также можете получить исполняемые фрагменты исходного кода из онлайновой версии этой книги в liveBook по адресу https://livebook.manning.com/book/software-engineers-guide-to-deep-learning-system-design и с веб-сайта издательства Manning (www.manning.com).

Лаборатория исходного кода «hello world» (в дополнении A) содержит полную, хотя и упрощенную, мини-систему глубокого обучения с наиболее важными компонентами (управлением наборами данных, тренировкой и раздачей моделей). Мы предлагаем вам опробовать указанную там систему после прочтения первой главы книги либо сделать это до того, как вы испытаете наши образцы служб, описанные в этой книге. Данная лаборатория также предоставляет скрипты командной оболочки и ссылки на все ресурсы, необходимые для того, чтобы приступить к работе.

Помимо лаборатории исходного кода, эта книга содержит множество примеров исходного кода в виде отдельных нумерованных листингов и внутри обычного текста. В обоих случаях исходный код отформатирован `шрифтом фиксированной ширины, подобным этому`, чтобы отделять его от обычного текста. Иногда исходный код также выделяется **жирным шрифтом**, чтобы выделять тот исходный код, который

изменился по сравнению с предыдущими шагами в данной главе, например когда новая функциональная возможность добавляется к существующей строке исходного кода.

Во многих случаях изначальный исходный код был переформатирован; мы добавили переносы строк и переработали отступы, чтобы уместиться в доступное пространство страницы книги. В редких случаях даже этого было недостаточно, и листинги включали маркеры продолжения строки (➥). Вдобавок нередко комментарии в исходном коде из листингов удалялись, когда исходный код описывался в тексте. Многие листинги сопровождаются аннотациями к исходному коду, выделяющими важные концепции.