Лекция 1. История создания искусственного интеллекта

> Искусственный интеллект станет окончательной версией Google. Окончательной поисковой системой, которая будет понимать во Всемирной паутине всё. Она будет точно понимать, чего вы хотите, и давать вам то, что нужно. Сейчас мы и близко к этому не подходим. Тем не менее мы можем постепенно к этому приближаться, и это в основном то, над чем мы работаем.
> - Ларри Пейдж, соучредитель компании Google Inc. и генеральный директор компании Alphabet

Научная фантастика всегда была для нас способом понять потенциальные последствия новых технологий, и искусственный интеллект (ИИ) был главной темой. Самые запоминающиеся научно-фантастические персонажи включают андроидов, или компьютеры, которые начинают сознавать себя, например, в кинофильмах «Терминатор», «Бегущий по лезвию», «2001 год: Космическая одиссея» и даже «Франкенштейн».

Но с неумолимым темпом развития новых технологий и инноваций сегодня научная фантастика начинает становиться реальностью. Теперь мы можем разговаривать с нашими смартфонами и получать ответы; наши аккаунты в социальных сетях предоставляют нам контент, который нас интересует; наши банковские приложения предоставляют нам напоминания и т. д. Такое персонализированное создание контента выглядит почти волшебным, но быстро становится нормальным в нашей повседневной жизни.

Для того чтобы разобраться в ИИ, важно знать азы его богатой истории. Мы увидим, как развитие этой индустрии было полно инновационных прорывов и спадов. Кроме того, в этой области существует группа блестящих исследователей и ученых, таких как Алан Тьюринг, Джон Маккарти, Марвин Мински и Джеффри Хинтон, которые раздвинули границы этой технологии. Но через все это шел постоянный прогресс.

Алан Тьюринг и тест Тьюринга

Алан Тьюринг (Alan Turing) – выдающаяся фигура в области информатики и искусственного интеллекта. Его часто называют родоначальником ИИ.

В 1936 году он написал работу под названием "О вычислимых числах" (On Computable Numbers). В ней он изложил ключевые концепции компьютера, который стал известен как машина Тьюринга. Имейте в виду, что настоящие компьютеры будут разработаны только через десять лет.

И все же исторической для ИИ станет его статья под названием "Вычислительная техника и интеллект" (Computing Machineiy and Intelligence). В ней он сосредоточился на идее машины с интеллектом. Но для того чтобы создать такую машину, должен быть способ измерить ее интеллект. Что такое интеллект, по крайней мере, для машины?

Именно в этой статье он придумал свой знаменитый "тест Тьюринга". По сути, он представляет собой игру с тремя игроками: двумя людьми и одним

компьютером. Оценщик, человек, задает открытые вопросы двум другим (человеку и компьютеру) с целью установить, кто из них является человеком. Если оценщик не может это установить, то предполагается, что компьютер имеет интеллект.

Гениальность этой идеи заключается в том, что нет необходимости видеть, действительно ли машина что-то знает, осознает себя или даже права ли она вообще. Вместо этого тест Тьюринга показывает, что машина может обрабатывать большие объемы информации, интерпретировать речь и общаться с людьми.

В 2014 году отмечен случай, когда создалось впечатление, будто бы тест Тьюринга был пройден. Он был связан с компьютером, который утверждал, что ему 13 лет. Интересно, что человеческие судьи, скорее всего, были одурачены, потому что некоторые ответы содержали ошибки.

Затем в мае 2018 года на конференции, проводимой компанией Google по вводу/выводу, генеральный директор Сундар Пичаи (Sundar Pichai) проделал выдающуюся демонстрацию Google Ассистента. Перед живой аудиторией он показал устройство, которое позвонило местному парикмахеру договориться о встрече. Женщина на другом конце провода вела себя так, словно разговаривала с человеком!

Тем не менее это устройство все-таки, вероятно, не прошло тест Тьюринга. Причина заключается в том, что разговор был сосредоточен на одной теме – тема не была открытой.

Неудивительно, что по поводу теста Тьюринга постоянно ведутся споры, поскольку некоторые люди считают, что им можно манипулировать. В 1980 году философ Джон Сёрл (John Searle) написал знаменитую работу под названием "Умы, мозги и программы" (Minds, Brains, and Programs), в которой он поставил собственный мысленный эксперимент, названный "аргументом китайской комнаты", чтобы подчеркнуть недостатки теста.

Цель эксперимента состоит в опровержении утверждения о том, что цифровая машина, наделённая «искусственным интеллектом» путём её программирования определённым образом, способна обладать сознанием в том же смысле, в котором им обладает человек. Иными словами, целью является опровержение гипотезы так называемого «сильного» искусственного интеллекта и критика теста Тьюринга.

Сильный и слабый искусственные интеллекты – гипотеза в философии искусственного интеллекта, согласно которой некоторые формы искусственного интеллекта могут действительно обосновывать и решать проблемы.

Теория сильного искусственного интеллекта предполагает, что компьютеры могут приобрести способность мыслить и осознавать себя как отдельную личность (в частности, понимать собственные мысли), хотя и не обязательно, что их мыслительный процесс будет подобен человеческому.

Теория слабого искусственного интеллекта отвергает такую возможность.

Сёрл считал, что существуют две формы ИИ:

– Сильный ИИ. В этом случае машина действительно понимает то, что происходит. Она даже может иметь эмоции и проявлять творчество. По большей части этот ИИ экранизирован в научно-фантастических фильмах. Этот тип ИИ также называется развитым искусственным интеллектом (artificial general

intelligence, AGI). Обратите внимание, что на этой категории сосредоточены лишь несколько компаний, такие как подразделение DeepMind компании Google.

– Слабый ИИ. С ним машина основывается на процедуре сопоставления с шаблоном и обычно сосредоточена на узких задачах. Его примеры включают помощников Siri от компании Apple и Alexa от компании Amazon.

Реальность такова, что ИИ находится на ранних стадиях развития слабого ИИ. На достижение точки сильного ИИ могут легко уйти десятилетия. Некоторые исследователи считают, что этого вообще никогда не произойдет.

Учитывая ограничения теста Тьюринга, появились альтернативы, такие как:

– тест Курцвейла – Капора – этот тест был предложен футурологом Рэем Курцвейлом (Ray Kurzweil) и технологическим антрепренером Митчем Капором (Mitch Карог). Их тест требует, чтобы компьютер вел беседу в течение двух часов, и при этом двое из трех судей считали, что говорит человек. Капор, правда, не верит, что это будет реализовано до 2029 года;

– тест на кофе – этот тест был предложен сооснователем компании Apple Стивом Возняком (Steve Wozniak). Согласно тесту на кофе, робот должен быть в состоянии войти в дом незнакомца, найти кухню и заварить чашку кофе.

Кибернетика

В 1948 году Винер опубликовал книгу "Кибернетика, или Контроль и общение у животного и машины" (Cybernetics: Or Control and Communication in the Animal and the Machine). Несмотря на то что эта работа была научной и наполнена сложными уравнениями, она все же стала очень популярной, попав в список бестселлеров газеты New York Times.

Данная книга предвосхитила развитие теории хаоса, цифровых коммуникаций и даже компьютерной памяти. Эта книга оказала влияние и на ИИ. Подобно Маккалоку и Питтсу, Винер сравнивал человеческий мозг с компьютером. Более того, он сделал предположение, что компьютер сможет играть в шахматы и в конечном счете обыграет гроссмейстеров. Главная причина, по его убеждениям, будет заключаться в том, что машина сможет учиться по ходу игр. Он даже думал, что компьютеры смогут сами себя копировать.

Первая программа искусственного интеллекта

Разработка первой программы искусственного интеллекта связана Джоном Маккарти.

Интерес Джона Маккарти (John McCarthy) к компьютерам возрос в 1948 году, когда он посетил семинар под названием "Церебральные механизмы в поведении", на котором обсуждался вопрос о том, как машины в конечном итоге станут способными думать. Среди его участников были ведущие первопроходцы в этой области, такие как Джон фон Нейман (John von Neumann), Алан Тьюринг (Alan Turing) и Клод Шеннон (Claude Shannon).

Маккарти продолжал погружаться в развивающуюся компьютерную индустрию, в том числе работал в американской корпорации Bell Labs

(Лаборатории Белла), и в 1956 году организовал десятинедельный исследовательский проект в Дартмутском университете. Он назвал его "Исследованием искусственного интеллекта". И это был первый раз, когда данный термин получил применение.

Летом 1956 года Джон Маккарти, Марвин Мински, Клод Шеннон и Натан Рочестер организовали конференцию по поводу того, что они назвали «искусственный интеллект» (термин, придуманный Маккарти для этого случая). На этой конференции Аллен Ньюэлл (Allen Newell), Клифф Шоу (Cliff Shaw) и Герберт Саймон (Herbert Simon) продемонстрировали компьютерную программу под названием "Теоретик логики" (The Logic Theorist), которую они разработали в компании RAND (Research and Development Corporation). Эта программа была сосредоточена на решении различных математических теорем из книги "Начала математики" (Principia Mathematica).

Создание программы "Теоретик логики" стало задачей не из легких. Ньюэлл, Шоу и Саймон использовали компьютер IBM 701, в котором применялся машинный язык. Поэтому им пришлось создать высокоуровневый язык IPL (Information Processing Language – язык для обработки информации), который ускорил процесс программирования. В течение нескольких лет этот язык стал приоритетным для ИИ.

Компьютеру IBM 701 также не хватало памяти для "Теоретика логики". Это привело к еще одному нововведению – обработке списков. Она позволяла динамически выделять и высвобождать память по мере исполнения программы.

В итоге программа "Теоретик логики" считается первой когда-либо разработанной программой ИИ.

Несмотря на это, она не вызвала особого интереса! Дартмутская конференция стала в основном разочарованием. Подвергся критике даже сам термин "искусственный интеллект".

Что касается Маккарти, то он продолжил свою миссию по продвижению инноваций. В конце 1950-х годов он разработал язык программирования Lisp, который часто применялся для проектов на основе ИИ из-за простоты использования нечисловых данных. Он также создал такие понятия программирования, как рекурсия, динамическая типизация и сборка мусора. Язык Lisp продолжает использоваться и сегодня, например, в робототехнике и бизнес-приложениях. Работая над языком, Маккарти также стал одним из основателей Лаборатории искусственного интеллекта МТИ (Массачусетского технологического института).

В 1961 году он сформулировал концепцию совместного использования времени компьютеров, которая оказала трансформирующее влияние на эту индустрию. Эта концепция также привела к развитию Интернета и облачных вычислений.

Несколько лет спустя он основал Лабораторию искусственного интеллекта в Стэнфорде.

В 1969 году он написал статью под названием "Автомобили под управлением компьютеров" (Computer-Controlled Cars), в которой объяснил, как человек может

вводить в компьютер направления движения с помощью клавиатуры, а телевизионная камера будет управлять автомобилем.

В 1971 году он получил премию Тьюринга. Эта премия считается Нобелевской премией по компьютерным наукам.

Золотой век искусственного интеллекта

С 1956 по 1974 год область искусственного интеллекта была одной из самых горячих в технологическом мире. Главным катализатором стало быстрое развитие компьютерных технологий. Они превратились из массивных систем, основанных на вакуумных лампах, в более миниатюрные системы, работающие на интегральных схемах, которые были намного быстрее и имели больший объем памяти.

Главным источником финансирования проектов ИИ стало Агентство перспективных научно-исследовательских проектов (Advanced Research Projects Agency, ARPA) Министерства обороны США, которое было создано в конце 1950-х годов после шока, полученного от запуска советского космического спутника.

Помимо компании IBM, частный сектор участвовал в разработке ИИ мало. Следует учитывать, что к середине 1950-х годов компания IBM отойдет от этой темы и сосредоточится на коммерциализации своих компьютеров. Клиенты действительно опасались, что эта технология приведет к значительным потерям рабочих мест. И поэтому компания IBM не хотела, чтобы ее в этом обвиняли.

Другими словами, большая часть инноваций в ИИ исходила из академических кругов. Например, в 1959 году Ньюэлл, Шоу и Саймон продолжили расширять границы в области искусственного интеллекта, разработав программу под названием "Универсальный решатель задач" (General Problem Solver). Как следовало из ее названия, речь шла о решении математических задач, таких как ханойская башня.

Но появилась масса других программ, которые пытались приблизиться к уровню сильного ИИ:

– SAINT или Symbolic Automatic INTegrator (Символический автоматический интегратор, 1961). Эта программа, созданная исследователем МТИ Джеймсом Слэглом (James Slagle), помогала решать задачи исчисления уровня младших курсов университетов. В дальнейшем она была обновлена, превратившись в другие программы, SIN и MACSYMA, которые делали гораздо более продвинутые математические расчеты. Программа SAINT фактически стала первым примером экспертной системы.

– ANALOGY (Аналогия, 1963). Эта программа была создана профессором МТИ Томасом Эвансом (Thomas Evans). Она продемонстрировала, что компьютер может решать задачи на рассуждение по аналогии с тестом IQ.

– STUDENT (Студент, 1964). Даниэль Бобров (Daniel Bobrow) создал это приложение ИИ для своей кандидатской диссертации под руководством Мински в МТИ. Система использовала обработку естественного языка для решения алгебраических задач уровня старших классов средней школы.

– ELIZA (Элиза, 1965). Эту программу разработал профессор МТИ Джозеф Вейценбаум (Joseph Weizenbaum), и она мгновенно стала большим хитом. Она даже вызвала шумиху в магистральной прессе. Данная программа была названа в честь Элизы (героини пьесы Джорджа Бернарда Шоу "Пигмалион") и служила психоаналитиком. Пользователь мог вводить вопросы, а Элиза давала советы (это был первый пример разговорного робота, или чат-бота). Некоторые люди, использовавшие ее, думали, будто программа была реальным человеком, что глубоко обеспокоило Вейценбаума, поскольку лежащая в основе технология была довольно простой.

– Computer Vision (Компьютерное зрение, 1966). В легендарной истории Марвин Мински из МТИ предложил студенту Джеральду Джею Сассману (Gerald Jay Sussman) провести лето, подключив камеру к компьютеру и заставив компьютер описывать то, что он видел. Джеральд так и поступил, и выстроил систему, которая обнаруживала базовые регулярности. Эта программа стала первым применением компьютерного зрения.

– Mac Hack (1968). Профессор МТИ Ричард Д. Гринблатт (Richard D. Greenblatt) создал эту программу для игры в шахматы. Данная программа была первой, которая выступала в реальных турнирах и получила рейтинг С, т. е. второй по сложности уровень, достигаемый средним клубным или турнирным шахматистом-любителем.

– Hearsay I (конец 1960-х). Профессор Радж Редди (Raj Reddy) разработал систему непрерывного распознавания речи. Некоторые из его учеников затем продолжат начатую работу и создадут Dragon Systems, ставшую крупной технологической компанией.

В этот период появилось много научных работ и книг по искусственному интеллекту. Некоторые темы включали байесовы методы, машинное обучение и компьютерное зрение.

Но в целом существовало две главенствующие теории ИИ. Одну из них возглавлял Минский, который говорил, что должны существовать символические системы. Иными словами, ИИ должен основываться на традиционной компьютерной логике или предварительном программировании, т. е. на использовании таких подходов, как инструкции если-то-иначе.

И был Фрэнк Розенблатт (Frank Rosenblatt), который считал, что ИИ должен использовать системы, подобные головному мозгу, такие как нейронные сети (эта область также носила название коннекционизма). Но вместо того чтобы называть внутренние механизмы нейронами, он называл их персептронами. Система могла учиться с течением времени по мере поступления данных.

В 1957 году Розенблатт создал для этого первую компьютерную программу-персептрон под названием "Mark 1". Она также поддерживала камеры, помогавшие проводить различие между двумя изображениями (они имели формат 20x20 пикселов).

Эта программа определенно стала для ИИ прорывной. В газете "Нью-Йорк тайме" даже вышла хвалебная для Розенблатта статья, провозгласившая, что "Navy сегодня показал эмбрион электронного компьютера, который, как ожидает автор,

сможет ходить, говорить, видеть, писать, воспроизводить себя и осознавать свое существование".

Но с персептроном все еще оставались проблемы. Во-первых, нейронная сеть имела только один слой (главным образом из-за отсутствия в то время вычислительной мощности). Во-вторых, исследования мозга все еще находились на начальной стадии развития и мало что могли дать в плане понимания когнитивных способностей.

Мински вместе с Сеймуром Пейпертом (Seymour Papert) написал книгу "Персептроны" (1969). Авторы подвергли безжалостной атаке подход Розенблатта, и этот подход быстро сошел на нет. Отметим, что в начале 1950-х годов Мински разработал грубую нейросетевую машину, в частности используя сотни вакуумных трубок и запасных частей от бомбардировщика B-24. И он убедился на своем опыте, что эта технология была слишком далека от того, чтобы стать работоспособной.

Розенблатт попытался сопротивляться, но было уже поздно. Сообщество ИИ быстро разочаровалось в нейронных сетях.

Однако к 1980-м годам его идеи возродятся, что приведет к революции в ИИ, в первую очередь с развитием технологии глубокого обучения.

По большей части золотой век ИИ был в состоянии свободного плавания и будоражил умы. Некоторые самые яркие ученые в мире пытались создавать машины, которые могли бы действительно думать. Но оптимизм часто доходил до крайности.

В 1965 году Саймон заявил, что в течение 20 лет машина сможет делать все, что может человек. Затем в 1970 году в интервью журналу Life он сказал, что это произойдет всего через 3-8 лет.

К сожалению, следующая фаза ИИ будет гораздо темнее. Все больше ученых становились скептиками. Пожалуй, самым яростным был философ Хьюберт Дрейфус (Hubert Dreyfus). В таких книгах, как "Что компьютеры все-таки не могут делать: критика искусственного разума" (What Computers Still Can’t Do: A Critique of Artificial Reason), он высказывал свои идеи о том, что компьютеры не похожи на человеческий мозг и что искусственный интеллект, к сожалению, не оправдает высоких ожиданий.

Зима искусственного интеллекта

В начале 1970-х годов энтузиазм в отношении ИИ начал угасать. Этот период станет называться "зимой ИИ" и продлится до 1980 года.

Несмотря на то что в области ИИ было проделано много значительных шагов вперед, они все-таки были главным образом академическими и предусматривали контролируемые среды. В то время компьютерные системы были еще ограниченными. Например, компьютер DEC PDP-11/45, который очень часто использовался для исследований ИИ, имел возможность расширения оперативной памяти только до 128 Кбайт.

Язык Lisp также не был для компьютерных систем идеальным. И даже наоборот, в корпоративном мире в центре внимания находился язык FORTRAN.

Далее, имелся целый ряд сложных аспектов понимания интеллекта и логического рассуждения. И двусмысленность является лишь одним из примеров. Это ситуация, когда слово имеет более одного смысла, что увеличивает трудность в работе программы ИИ, т. к. ей тоже нужно было понимать контекст.

Наконец, экономическая ситуация в 1970-е годы была далеко не стабильной. Наблюдались устойчивая инфляция, медленный рост и перебои в поставках, как, например, во время нефтяного кризиса.

Самый большой удар по ИИ был нанесен в 1973 году докладом профессора сэра Джеймса Лайтхилла (James Lighthill). Профинансированный парламентом Соединенного Королевства, этот доклад стал полным развенчанием "грандиозных целей" сильного ИИ. Главной трудностью, как отмечалось в докладе, был "комбинаторный взрыв", который становился непреодолимой проблемой, когда модели становились слишком сложными и трудно настраиваемыми.

В заключении доклада говорилось, что "ни в одной части этой области сделанные до сих пор открытия не оказали того значительного воздействия, которое было тогда обещано". Он был настолько пессимистичен, что не верил, что компьютеры способны распознавать образы или обыгрывать гроссмейстера в шахматы.

Дела пошли настолько плохо, что многие исследователи изменили свою карьеру. А те, кто по-прежнему изучал ИИ, часто называли свою работу другими терминами – например, машинное обучение, распознавание образов и информатика.

Взлет и падение экспертных систем

Даже во время зимы искусственного интеллекта продолжались крупные инновации. Одной из них явился алгоритм обратного распространения, который необходим для назначения весов в нейронных сетях. Затем была разработана рекуррентная нейронная сеть (RNN-сеть, recurrent neural network). Она позволяет соединениям перемещаться через входной и выходной слои.

Но в 1980-х и 1990-х годах также появились экспертные системы. Ключевым фактором для них стал взрывной рост числа персональных компьютеров и мини-компьютеров.

Экспертные системы были основаны на концепции символической логики Минского, включающей сложные пути. Они часто разрабатывались экспертами в определенных областях, таких как медицина, финансы и автомобилестроение.

Хотя существуют экспертные системы, которые восходят к середине 1960-х годов, они не получили коммерческого использования вплоть до 1980-х. Примером может служить экспертная система XCON (eXpert CONfigurer – эксперт-конфигуратор), разработанная Джоном Макдермоттом (John McDermott) в Университете Карнеги – Меллона. Данная система позволяла оптимизировать подбор компонентов компьютера и изначально имела около 2500 правил.

Когда компании убедились в успехе экспертной системы XCON, произошел бум экспертных систем, превратившийся в миллиардную индустрию.

Следует учесть, что компания IBM использовала экспертную систему для своего компьютера Deep Blue. В 1996 году он обыграл гроссмейстера Гарри Каспарова в одном из шести матчей. Компьютер Deep Blue, который компания IBM разрабатывала с 1985 года, обрабатывал 200 млн позиций в секунду.

Но с экспертными системами имелись и проблемы. Они часто были узкоспециализированными и трудноприменимыми в других сферах. Более того, по мере расширения экспертных систем управлять ими и подавать в них данные становилось все труднее. В итоге результаты содержали больше ошибок. Далее процесс тестирования таких систем часто оказывался сложным. Следует признать тот факт, что временами эксперты расходились во мнениях по фундаментальным вопросам. Наконец, экспертные системы не учились с течением времени. Вместо этого необходимо было постоянно обновлять базовые логические модели, что значительно увеличивало затраты и сложности.

К концу 1980-х годов экспертные системы начали терять популярность в деловом мире, и многие стартапы претерпели слияние или обанкротились. На самом деле, это обусловило еще одну зиму ИИ, которая продлилась примерно до 1993 года. Персональные компьютеры быстро откусывали куски рынков аппаратного обеспечения более высокого класса, что означало резкое сокращение числа машин на основе языка Lisp.

Нейронные сети и глубокое обучение

Первые работы по нейронным сетям связывают с именем Джеффри Хинтона.

Будучи подростком в 1950-х годах, Джеффри Хинтон (Geoffrey Hinton) хотел стать профессором и изучать ИИ. Он происходил из семьи известных ученых (его прапрадедом был английский математик и логик Джордж Буль). Его мама часто говорила: "Стань академиком либо неудачником".

Даже во время первой зимы искусственного интеллекта Хинтон продолжал увлекаться искусственным интеллектом и был убежден, что подход Розенблатта к нейронным сетям был правильным. Так, в 1972 году он получил докторскую степень по данной теме в Эдинбургском университете.

Но в этот период многие считали, что Хинтон попусту тратит свое время и таланты. ИИ, по существу, считался пограничной областью. Научные исследования в этой области даже не воспринимались как наука.

Но это только еще больше воодушевляло Хинтона. Он наслаждался своим положением аутсайдера и знал, что его идеи в конце концов победят.

Хинтон понял, что самым большим препятствием для ИИ является компьютерная мощь. Он также видел, что время было на его стороне. Закон Мура предсказывал, что число компонентов на чипе будет удваиваться примерно каждые 18 месяцев.

Тем временем Хинтон неустанно работал над разработкой стержневых теорий нейронных сетей – того, что со временем стало известно как глубокое обучение. В 1986 году он написал – вместе с Дэвидом Румелхартом (David Rumelhart) и Рональдом Дж. Уильямсом (Ronald J. Williams) – новаторскую работу под названием "Усвоение представлений путем обратного распространения ошибок"

(Learning Representations by Back-propagating Errors). В ней изложены ключевые процессы для использования обратного распространения в нейронных сетях. В результате было достигнуто значительное улучшение точности, в частности, предсказаний и визуального распознавания.

Технологические движущие силы современного искусственного интеллекта

Помимо достижений в новых концептуальных подходах, теориях и моделях, у ИИ были и другие важные движущие силы.

Взрывной рост числа наборов данных. Интернет был главенствующим фактором для ИИ, поскольку он позволил создавать массивные совокупности данных.

Инфраструктура. Пожалуй, самой влиятельной компанией для ИИ за последние 15 лет или около того была компания Google. Для того чтобы идти в ногу с индексацией Всемирной паутины, которая росла с ошеломляющей скоростью, указанная компания должна была придумать творческие подходы к строительству масштабируемых систем. Результатом стали инновации в кластерах из серийных серверов, виртуализации и программно-информационном обеспечении с открытым исходным кодом. А с запуском проекта Google Brain в 2011 году компания Google также стала одной из первых приверженицц глубокого обучения. А еще через несколько лет эта компания наняла Хинтона.

Графические процессоры (Graphics Processing Units, GPU – графическое процессорное устройство, ГПУ). Эта микросхемная технология, первопроходцем в которой стала компания NVIDIA, изначально предназначалась для высокоскоростной графики в играх. Но архитектура графических процессоров в конечном счете станет идеальной и для ИИ. Обратите внимание, что большинство исследований в области глубокого обучения проводится с помощью этих чипов. Причина в том, что при параллельной обработке скорость повышается многократно, в отличие от традиционных центральных процессоров. Это значит, что вычисление модели может занять день-два по сравнению с неделями или даже месяцами в обычных условиях.

Ключевые моменты

Эволюционирование технологий часто занимает больше времени, чем первоначально понималось.

Искусственный интеллект – это не только компьютерные науки и математика. В ИИ был сделан ключевой вклад из таких областей, как экономика, нейробиология, психология, лингвистика, электротехника, математика и философия.

Существует два основных типа ИИ: слабый и сильный. При сильном ИИ машины осознают себя, тогда как слабый предназначен для систем, которые сконцентрированы на решении конкретных задач. В настоящее время ИИ находится на слабой стадии развития.

Тест Тьюринга – это часто встречающийся способ проверить, может ли машина думать. Он основан на том, действительно ли кто-то из людей считает систему разумной.

Ключевые факторы развития ИИ включают в себя, в частности, новые теории таких исследователей, как Хинтон, взрывной рост объема данных, новую технологическую инфраструктуру и графические процессоры.

Выводы

Нет ничего нового в том, что термин "искусственный интеллект" сегодня является модным. Этот термин видел различные турбулентные циклы подъема и спада.

Может быть, он снова впадет в немилость? Возможно. Но на этот раз в ИИ появились настоящие инновации, которые подвергают бизнес трансформации. Мегатехнологические компании, такие как Google, Microsoft и Facebook, считают эту категорию одним из главных приоритетов. В целом похоже, что можно смело делать ставку на то, что ИИ будет продолжать расти и изменять наш мир.

Лекция 2. Данные – топливо для искусственного интеллекта

Введение

Компания Pinterest является одним из самых популярных стартапов в Кремниевой долине, позволяющим пользователям прикреплять свои любимые предметы для создания привлекательных виртуальных досок.

Популярным занятием в Pinterest является планирование свадеб. У будущей невесты будут "доски" (или тематические коллекции) для платьев, мест проведения свадьбы и медового месяца, тортов, приглашений и т. д.

Это также означает, что Pinterest имеет преимущество в сборе огромного объема ценных данных. Отчасти это помогает обеспечивать целенаправленную рекламу. Но есть и возможности для проведения кампаний по электронной почте. Однажды компания Pinterest прислала сообщение, в котором говорилось:

"Ты выходишь замуж! И поскольку мы любим планировать свадьбы – в особенности подбирать прекрасные писчебумажные и галантерейные принадлежности, – мы приглашаем вас просмотреть наши лучшие доски, кураторами которых являются графические дизайнеры, фотографы и другие будущие невесты, и все они являются пиннерами с острым глазом и брачным союзом на уме".

Проблема, однако, оказалась в том, что многие получатели письма уже были замужем либо не собирались выходить замуж в ближайшее время.

Это был важный урок для крупной IT-компании. Бывает, что портачат даже самые технологически подкованные компании.

Основы данных

Хорошо иметь представление о жаргоне, связанном с данными.

Прежде всего, бит (bit – от англ, binary digit, т. е. двоичная цифра) – это наименьшая форма данных в компьютере. Думайте о бите как об атоме. Бит может быть либо 0, либо 1, т. е. двоичным. Он также обычно используется для измерения объема передаваемых данных (например, в сети или Интернете).

Байт главным образом предназначен для измерения хранимых данных. Конечно, количество байтов может очень быстро увеличиваться. Ниже в таблице посмотрим, насколько быстро.

Пример	Значение	Вариант использования
Мегабайт	1024 килобайт	Небольшая книга
Гигабайт	1024 мегабайт	Около 230 песен
Терабайт	1024 гигабайт	500 часов фильмов
Петабайт	1024 терабайт	Пять лет работы системы наблюдения Земли (Earth Observing System, EOS)
Экзабайт	1024 петабайт	Вся библиотека конгресса США, увеличенная в 3000 раз
Зеттабайт	1024 экзабайт	36 000 лет видео в формате HD-TV

Йоттабайт	1024 зеттабайт	Для этого потребуется центр обработки данных размером со штат Делавэр и штат Род-Айленд, вместе взятые

Данные также могут поступать из разных источников:

– Всемирная паутина/социальные сети (Facebook, Twitter, Instagram, YouTube);

– биометрические данные (фитнес-трекеры, генетические тесты);

– системы кассовых терминалов в точках продаж (от традиционных магазинов до веб-сайтов электронной коммерции);

– Интернет вещей, или IoT (ID-теги и умные устройства);

– облачные системы (бизнес-приложения, такие как Salesforce.com);

– корпоративные базы данных и электронные таблицы;

– и многие другие.

Типы данных

Существует четыре способа организации данных. Во-первых, это структурированные данные, которые обычно хранятся в реляционной базе данных или электронной таблице. Вот несколько примеров:

– финансовая информация;

– номера социального страхования;

– адреса;

– продуктовая информация;

– данные точек продаж;

– телефонные номера.

По большей части со структурированными данными работать проще. Эти данные часто поступают из систем управления взаимоотношениями с клиентами (Customer Relationship Management, CRM) и систем управления бизнес-процессами (Enterprise Resource Planning, ERP), и они обычно имеют меньшие объемы.

Существуют различные бизнес-информационные и аналитические программы из категории BI (business intelligence), которые помогают выводить из структурированных данных сущностные сведения. Тем не менее этот тип данных составляет около 20 % проекта на основе ИИ.

Большинство из них будет поступать из неструктурированных данных, т. е. информации, которая не имеет предопределенного форматирования. Специалисту, работающему с данными, придется их самостоятельно форматировать, что может отнимать много времени. Но существуют инструменты, такие как базы данных следующего поколения, например, основанные на технологии NoSQL, которые помогают в этом процессе. Системы ИИ также являются эффективными с точки зрения управления и структурирования данных, поскольку алгоритмы могут распознавать регулярности.

Вот примеры неструктурированных данных:

– файлы изображений;

– видеофайлы;

– аудиофайлы;
– текстовые файлы;
– информация социальных сетей, такая как твиты и посты;
– спутниковые снимки.

Дальше, имеются данные, которые представляют собой гибрид структурированных и неструктурированных источников – так называемые полуструктурированные данные. Информация имеет некоторые внутренние теги, которые помогают с категоризацией.

Примеры полуструктурированных данных включают XML (Extensible Markup Language – язык расширенной разметки), который основан на различных правилах идентификации элементов документа, и JSON (JavaScript Object Notation – объектная нотация языка JavaScript), который является способом передачи информации по Всемирной паутине через интерфейсы прикладного программирования (application programming interface, API).

Но полуструктурированные данные составляют лишь от 5 до 10 % всех данных.

Наконец, существуют данные временных рядов, которые могут быть как структурированными, неструктурированными, так и полуструктурированными. Этот тип информации предназначен для целей взаимодействия, в частности для отслеживания "путешествия клиента". Они являются результатом сбора информации о том, когда пользователь заходит на веб-сайт, использует приложение или даже заходит в магазин.

Большие данные

Благодаря повсеместному доступу в Интернет, мобильным и носимым устройствам произошел выброс стремительного потока данных. Каждую секунду Google обрабатывает более 40 тыс. запросов или 3,5 млрд. в сутки. Поминутно пользователи Snapchat делятся 527 760 фотографиями, а пользователи YouTube смотрят более 4,1 млн. видео. Кроме того, электронная почта продолжает демонстрировать значительный рост. Каждую минуту пользователи получают 156 млн. сообщений.

Но нужно учитывать и другое: компании и машины тоже генерируют огромные объемы данных. Согласно исследованиям немецкого портала Statista, к 2020 году число датчиков достигнет 12,86 млрд. единиц.

В свете всего этого, похоже, можно смело ставить на то, что объемы данных будут продолжать расти быстрыми темпами. В докладе исследовательской и консалтинговой компании IDC (International Data Corporation) под названием "Data Age 2025" (Век данных 2025) объем создаваемых данных, как ожидается, к 2025 году достигнет ошеломляющих 163 зеттабайт. Это в 10 раз больше того, что было в 2017 году.

Для того чтобы со всем этим справиться, появилась категория технологий под названием "большие данные" (Big Data). Вот как компания Oracle объясняет важность этого тренда: "Сегодня большие данные стали капиталом. Подумайте о некоторых крупнейших технологических компаниях в мире. Большая часть

ценности, которую они предлагают, исходит из их данных, которые они постоянно анализируют, работая более эффективно и разрабатывая новые продукты".

Так что да, большие данные останутся важной частью многих проектов на основе ИИ.

Тогда что же такое большие данные? Каким будет хорошее определение этого понятия? На самом деле, нет ни одного, хотя есть много компаний, которые сосредоточены на этом рынке! Но у больших данных есть следующие характерные особенности, которые называются тремя V. Дуг Лейни (Doug Laney), аналитик глобальной аналитической компании Gartner, придумал этот перечень еще в 2008 году): огромный объем (volume), разнообразие структур и источников (variety) и высокая скорость обработки (velocity).

Объем

Это масштаб данных, которые часто не структурированы. Не существует какого-то незыблемого правила относительно их порога, но обычно это десятки терабайт.

Когда речь заходит о больших данных, то их объем часто является серьезной проблемой. Но облачные вычисления и базы данных следующего поколения оказали большую помощь – с точки зрения емкости и более низких затрат.

Разнообразие

Эта характерная особенность описывает многообразие данных, скажем сочетание структурированных, полуструктурированных и неструктурированных данных (объясненных выше). Она также показывает различные источники данных и их использование. Несомненно, интенсивный рост неструктурированных данных сыграл ключевую роль в разнообразии больших данных.

Управление разнообразием данных может быстро стать серьезной преградой. Тем не менее машинное обучение часто является именно тем, что помогает оптимизировать его обработку.

Скорость

Эта характерная особенность показывает темп, с которым данные создаются. Как было показано ранее в этой главе, такие веб-службы, как YouTube и Snapchat, имеют экстремальные уровни скорости (их часто называют "пожарной частью" данных). Обеспечение скорости требует значительных инвестиций в технологии нового поколения и центры обработки данных. Данные также часто обрабатываются прямо в памяти, а не на дисковых системах.

Из-за этих трудностей скорость часто считается самой сложной из всех трех V. Следует честно признать, в современном цифровом мире люди хотят получить интересующие их данные как можно быстрее. Если это происходит слишком медленно, то люди расстраиваются и уходят на другие площадки.

Однако с годами, по мере развития больших данных, добавлялось все больше V. В настоящее время их насчитывается свыше десяти.

Но вот несколько самых распространенных.

Достоверность (Veracity). Речь идет о данных, которые считаются точными. В этой главе мы рассмотрим некоторые технические приемы оценивания достоверности.

Ценность (Value). Эта характерная особенность показывает полезность данных. Часто речь идет о наличии надежного источника.

Изменчивость (Variability). Эта характерная особенность означает, что данные с течением времени обычно изменяются. Например, это относится к контенту социальных сетей, который может претерпеть метаморфозу, основываясь на совокупном мнении относительно новых событий и последних новостей.

Визуализация (Visualization). Эта особенность предусматривает использование визуальных элементов, таких как графики, в целях более точного понимания данных.

Как видите, управление большими данными имеет много движущихся частей, что приводит к сложности. Это помогает объяснить, почему многие компании до сих пор используют только крошечную часть своих данных.

Базы данных и другие инструменты

Существует масса инструментов, которые помогают работать с данными. В их сердцевине лежит база данных. И ничего удивительного в том, что эта критически важная технология претерпевала эволюцию на протяжении десятилетий. Но даже более старые технологии, такие как реляционные базы данных, все еще широко используются сегодня. Когда речь заходит о критически важных данных, компании неохотно вносят изменения – даже если есть явные преимущества от нововведений.

Для того чтобы разобраться в этом рынке, давайте вернемся в 1970 год, когда исследователь в области компьютерных наук из компании IBM Эдгар Кодд (Edgar Codd) опубликовал свою работу "Реляционная модель данных для крупных совместных банков данных" (A Relational Model of Data for Large Shared Data Banks).

Она стала прорывной, поскольку ввела структуру реляционных баз данных. До этого момента базы данных были довольно сложными и жестко структурированными в виде иерархий. Это отнимало много времени на поиск и отыскание связей в данных.

Что касается подхода Кодда к реляционным базам данных, то он был построен для более современных машин. Язык сценариев SQL (Structured Query Language – язык структурированных запросов) был прост в использовании, позволяя выполнять операции CRUD (Create, Read, Update, and Delete – создание, чтение, обновление и удаление). Таблицы тоже имели соединения с первичными и внешними ключами, что позволяло создавать важные соединения, такие как:

– один-к-одному – одна строка в таблице связана только с одной строкой в другой таблице. Пример: уникальный номер водительского удостоверения связан с одним сотрудником;

– один-ко-многим – это место, где одна строка в таблице связана с другими таблицами. Пример: клиент имеет несколько заказов на покупку;

– многие-ко-многим – строки из одной таблицы ассоциированы со строками других таблиц. Пример: разные отчеты имеют различных авторов.

С помощью этих типов структур реляционная база данных могла оптимизировать процесс создания сложных отчетов. Она действительно была революционной.

Но, несмотря на все преимущества, компания IBM не заинтересовалась указанной технологией и продолжала сосредотачиваться на собственных системах. Компания считала, что реляционные базы данных были слишком медленными и хрупкими для корпоративных клиентов.

Но появился еще один человек, у которого на этот счет было другое мнение: это был Ларри Эллисон (Larry Ellison.). Он прочитал статью Кодда и понял, что изложенные в ней идеи изменят правила игры. Ради того, чтобы это доказать, в 1977 году он стал одним из основателей компании Oracle, сосредоточившись на строительстве реляционных баз данных, которые быстро стали массовым рынком. Статья Кодда была, по сути, дорожной картой его предпринимательской деятельности.

И только в 1993 году компания IBM выпустила собственную реляционную базу данных DB2. Но было уже слишком поздно. К этому времени компания Oracle стала лидером на рынке баз данных.

На протяжении 1980-х и 1990-х годов технология реляционных баз данных была стандартом для мейнфреймовых компьютеров и клиент-серверных систем. Правда, когда большие данные стали важным фактором, в указанной выше технологии проявились серьезные недостатки, такие как:

– расползание данных – со временем разные базы данных распространялись по всей организации. В результате сложнее стало централизовывать данные;

– новые среды – технология реляционных баз данных не была создана для облачных вычислений, высокоскоростных данных или неструктурированных данных;

– высокая стоимость – реляционные базы данных бывают дорогостоящими. Это означает, что использование указанной технологии для проектов на основе ИИ может оказаться запретительным;

– сложности разработки – современная разработка программно-информационного обеспечения в значительной степени зависит от итеративной обработки. Но у реляционных баз данных возникли трудности с этим процессом.

В конце 1990-х годов появились проекты с открытым исходным кодом, призванные помочь в создании систем баз данных следующего поколения. Возможно, самый критически важный из них поступил от Дуга Каттинга (Doug Cutting), который разработал библиотеку Lucene, предназначенную для текстового поиска.

Его технология базировалась на сложной индексной системе, обеспечивающей производительность с низкой задержкой. Поисковая технология Lucene мгновенно стала хитом и начала эволюционировать, в частности, в

модульный поисковый каркас Apache Nutch, который эффективно ползал по Всемирной паутине и сохранял данные в индексе.

Но была и большая проблема: для того чтобы ползать по паутине, нужна была инфраструктура, способная к гипермасштабированию. Именно поэтому в конце 2003 года Каттинг приступил к разработке нового вида инфраструктурной платформы, которая могла бы решить эту проблему. Он взял эту идею из статьи, опубликованной в Google, в которой описывалась их массивная файловая система. Год спустя Каттинг построил свою новую платформу, которая обеспечивала изощренное хранение без особых сложностей. В ее основе был алгоритм MapReduce, который позволял обрабатывать данные на многочисленных серверах. Затем результаты сливались воедино, позволяя получать содержательные отчеты.

В конце концов система Каттинга трансформировалась в платформу под названием Hadoop – и она станет играть существенную роль в управлении большими данными, например, с целью создания сложных хранилищ данных. Изначально ею пользовалась компания Yahoo!, а затем она быстро распространилась по мере того, как такие компании, как Facebook и Twitter, принимали эту технологию на вооружение. Теперь эти компании могли иметь панорамный вид своих данных, а не только их подмножеств. А это означало возможность проведения более эффективных экспериментов с данными.

Но, будучи проектом с открытым исходным кодом, платформе Hadoop все еще не хватало изощренных систем для корпоративных клиентов. Для решения этой задачи стартап под названием Hortonworks настроил новые технологии, такие как YARN, поверх платформы Hadoop. Платформа имела такие способности, как аналитическая обработка прямо в памяти, онлайновая обработка данных и интерактивная обработка на основе SQL. Эти функциональные способности обеспечивали внедрение платформы Hadoop во многих корпорациях.

Но, конечно же, появились и другие проекты с открытым исходным кодом по хранению данных. Хорошо известными из них являются такие, как Storm и Spark, которые сосредоточены на потоковой передаче данных. Платформа Hadoop, с другой стороны, была оптимизирована для пакетной обработки.

Помимо хранилищ данных, были также инновации в традиционном бизнесе баз данных. Часто они известны как системы NoSQL. Возьмем, к примеру, MongoDB. Она началась как проект с открытым исходным кодом и превратилась в очень успешную компанию, которая стала публичной в октябре 2017 года. База данных MongoDB, имеющая более 40 млн скачиваний, предназначена для работы в облачных, локальных и гибридных средах.

В ней также обеспечивается большая гибкость по структурированию данных, которая основана на документной модели. В MongoDB можно даже управлять структурированными и неструктурированными данными в больших петабайтовых масштабах.

Несмотря на то что стартапы были источником инноваций в системах управления базами данных и хранения данных, важно отметить, что операторы мегатехнологий также сыграли свою критически важную роль. С другой стороны, таким компаниям, как Amazon.com и Google, пришлось отыскивать способы

справиться с огромным объемом данных из-за необходимости управления своими массивными платформами.

Одним из нововведений стало создание озера данных, которое позволяет беспрепятственно хранить структурированные и неструктурированные данные. Обратите внимание, что нет необходимости переформатировать данные. Озеро данных справится с этим и позволит быстро выполнять функции ИИ. Согласно исследованию аналитической фирмы Aberdeen, компании, использующие эту технологию, имеют в среднем 9 %-й органический рост по сравнению с теми, кто ее не использует.

Процесс обработки данных

Объем денег, потраченных на данные, просто огромен. Согласно прогнозам исследовательской и консалтинговой компании IDC, расходы на технологические решения по обработке больших данных и аналитике вырастут с 166 млрд. долларов в 2018 году до 260 млрд. долларов к 2022 году. Это составляет 11,9 %-й совокупный годовой прирост.

Вот что сообщает Джессика Гепферт (Jessica Goepfert), вице-президент программы по исследованию и анализу клиентской базы компании IDC:

"На высоком уровне организации обращаются к технологическим решениям по обработке и анализу больших данных, чтобы провести конвергенцию их физического и цифрового миров. Эта трансформация принимает разную форму в зависимости от индустрии. Например, в банковской и розничной торговле – двух самых быстрорастущих областях больших данных и аналитики – инвестиции направлены на управление и оживление клиентского опыта. В то время как в производстве промышленной продукции фирмы преображают себя, чтобы, по существу, стать высокотехнологичными компаниями, используя свои продукты в качестве платформы для обеспечения возможности и предоставления цифровых услуг".

Но высокий уровень расходов не обязательно приводит к хорошим результатам. По оценкам исследования, проведенного аналитической компанией Gartner, примерно 85% проектов на основе больших данных прекращаются до того, как они попадают в пилотную стадию.

Среди причин тому приводятся следующие:

– отсутствие четкой ориентации;
– грязные данные;
– инвестиции в неправильные ИТ-инструменты;
– проблемы со сбором данных;
– отсутствие поддержки со стороны ключевых заинтересованных сторон и компаньонов в организации.

В свете этого крайне важно иметь процесс обработки данных.

Несмотря на то что существует много подходов – часто превозносимых разработчиками / поставщиками программно-информационного обеспечения, – есть один, который имеет широкое признание. Группа экспертов, разработчиков программно-информационного обеспечения, консультантов и ученых в конце

1990-х годов создала процесс CRISP-DM (Cross-Industry Standard Process for Data Mining – Межотраслевой стандартный процесс для добычи регулярностей из данных).

Обратите внимание, что шаги 1-3 могут составлять 80% времени обработки данных.

Это связано с такими факторами, как:

– данные плохо организованы и поступают из разных источников (например, от разных поставщиков или из разных подразделений организации);

– недостаточно внимания уделяется средствам автоматизации;

– первоначальное планирование было недостаточным для масштаба проекта.

Также стоит иметь в виду, что процесс CRISP-DM не является строго линейным. Во время работы с данными может быть много итераций. Например, может иметься несколько попыток найти правильные данные и их протестировать.

Шаг 1. Понимание бизнеса

Необходимо иметь четкое представление о бизнес-задаче, которая должна быть решена.

На шаге 1 необходимо собрать подходящую команду для проекта. Только у крупных IT-компаний есть возможность набирать группу из кандидатов наук в области машинного обучения и науки о данных. Такая возможность является довольно редкой и дорогой.

Но для проекта на основе ИИ не нужна и армия первоклассных инженеров. На самом деле становится все проще применять модели машинного и глубокого обучения благодаря системам с открытым исходным кодом, таким как TensorFlow, и облачным платформам от компаний Google, Amazon.com и Microsoft. Другими

словами, может понадобиться всего пара человек с опытом работы в области науки о данных.

Далее, нужно найти людей, которые обладают необходимыми предметными знаниями по проекту в области ИИ. Им нужно будет продумать рабочие процессы, модели и тренировочные данные с особым пониманием индустрии и требований со стороны клиентов.

Наконец, необходимо оценить технические потребности. Какие инфраструктурные и программные средства будут использоваться? Возникнет ли необходимость в увеличении мощностей или приобретении новых программно-информационных решений?

Шаг 2. Понимание данных

На этом шаге нужно обратиться к источникам данных для проекта.

Следует учесть, что существует три главных источника.

1. Внутренние данные.

Эти данные могут поступать из веб-сайта, радиомаяков в магазине, датчиков Интернета вещей, мобильных приложений и т. д. Основное преимущество этих данных в том, что они являются бесплатными и адаптированы для бизнеса. Но, с другой стороны, есть и некоторые риски. Могут возникнуть проблемы, если не было уделено достаточного внимания форматированию данных или тому, какие данные следует отбирать.

2. Открытые исходные данные.

Обычно они находятся в свободном доступе, что, безусловно, является хорошим преимуществом. Некоторые примеры данных из открытых источников включают государственную и научную информацию. Доступ к данным часто осуществляется через API, что делает процесс довольно простым. Открытые исходные данные также обычно хорошо отформатированы. Однако некоторые из этих переменных могут быть неясными и могут допускать систематическое смещение, например, в сторону определенной демографической группы.

3. Сторонние данные.

Это данные от коммерческого поставщика. Комиссионные за предоставление этих данных бывают высокими. На самом деле, в некоторых случаях может возникать нехватка качества таких данных.

По информации компании Teradata, основанной на собственных разработках в области ИИ, около 70 % источников данных являются внутренними, 20 % – из открытых источников, и остальные – от коммерческих поставщиков. Но, несмотря на источник, все данные должны быть достоверными. Если они таковыми не являются, то, скорее всего, возникнет проблема "мусор на входе, мусор на выходе".

Для того чтобы оценить полученные данные, необходимо ответить на следующие вопросы:

– Являются ли данные полными? Чего может не хватать?
– Откуда взялись эти данные?
– Каковы были пункты их сбора?
– Кто прикасался к данным и обрабатывал их?

– Какие изменения произошли в данных?
– Какие проблемы имеются с их качеством?

Если мы работаем со структурированными данными, то этот этап должен быть проще. Правда, когда речь заходит о неструктурированных и полуструктурированных данных, придется выполнить разметку данных – и этот процесс может быть длительным. Однако на рынке появляются инструменты, которые помогут автоматизировать этот процесс.

Шаг 3. Подготовка данных

Первый шаг в процессе подготовки данных – решить, какие наборы данных использовать.

Давайте взглянем на сценарий. Предположим, мы работаем в издательской компании и хотите разработать стратегию по улучшению способов удержания клиентов. Некоторые данные, которые должны помочь, включают демографическую информацию о клиентуре, такую как возраст, пол, доход и образование. В целях придания большей цветности мы также можем посмотреть браузерную информацию. Какой тип контента интересует клиентов? Какова частота и продолжительность посещения веб-сайта?

Имеются ли еще какие-то интересные закономерности, скажем обращение к информации в выходные дни? Объединив источники информации, мы можем собрать мощную модель. Например, если в некоторых областях наблюдается снижение активности, то это может привести к риску ухода клиента. Это будет тревожным сигналом для специалистов отдела продаж о том, чтобы они обратились к клиентам лично.

Хотя этот процесс выглядит разумным, все же тут есть подводные камни. Включение или исключение даже одной переменной может оказать значительное негативное влияние на модель ИИ.

Для того чтобы понять почему, оглянитесь на финансовый кризис. Модели андеррайтинга ипотечных кредитов были сложными и основывались на огромном объеме данных. В нормальные экономические времена они работали довольно хорошо, т. к. крупные финансовые институты, такие как Goldman Sachs, JP Morgan и AIG, во многом опирались на них.

Но была одна проблема: модели не учитывали падение цен на жилье! Главная причина заключалась в том, что на протяжении десятилетий не было ни одного падения общенационального масштаба. И в связи с этим все исходили из допущения, что жилье – это главным образом локальное явление.

Как известно, цены на жилье не просто упали, а упали резко. Модели тогда оказались далеко не на высоте, и миллиарды долларов убытков едва не обрушили финансовую систему США. У федерального правительства не было иного выбора, кроме как предоставить в долг 700 млрд. долларов для спасения Уолл-стрит.

Разумеется, этот случай является крайним. Но он подчеркивает важность отбора данных. Именно здесь бывает необходима сплоченная команда экспертов в предметной области и исследователей данных.

Далее, на этапе подготовки данных, необходимо будет провести очистку данных. Дело в том, что все данные имеют проблемы. Даже такие компании, как Facebook, имеют в своих наборах данных пробелы, неоднозначности и выбросы. Это неизбежно.

Поэтому ниже перечислено несколько действий, которые нужно предпринять с целью очистки данных.

– Устранение повторов. Задать тесты для выявления любых дубликатов и удаления посторонних данных.

– Выбросы – это данные, которые находятся далеко за пределами диапазона большинства остальных данных. Это может означать, что информация не является полезной. Но, конечно же, бывают ситуации, когда верно обратное. Это действие необходимо для выявления мошенничества.

– Согласованность. Убедитесь, что есть четкие определения своих переменных. Даже такие термины, как "доход" или "клиент", могут иметь несколько значений.

– Правила проверки достоверности. Когда мы смотрим на данные, мы постараемся найти присущие им ограничения. Например, можно установить флажок для столбца "возраст". Если во многих случаях он превышает 120, то данные имеют серьезные проблемы.

– Разбиение на корзины. Некоторые данные не нуждаются в конкретизации. Имеет ли какое-то значение, если кому-то 35 или 37 лет? Вероятно, нет. Но сравнение возрастной категории от 30 до 40 с возрастной категорией от 41 до 50, скорее всего, будет иметь значение.

– Новизна. Являются ли данные своевременными и актуальными?

– Слияние. В некоторых случаях столбцы данных могут содержать очень похожую информацию. Возможно, один из них имеет рост в дюймах, а другой – в футах. Если модель не требует более детального числа, то можно оставить только столбец в футах.

– Кодирование с одним активным состоянием. Это способ заменить категориальные данные на числа. Например, допустим, у нас есть база данных со столбцом, в котором имеется три возможных значения: яблоко, ананас и апельсин. Мы можем представить яблоко как 1, ананас как 2 и апельсин как 3. Звучит разумно, не так ли? Это как посмотреть; проблема в том, что алгоритм ИИ может подумать, что апельсин больше яблока. Но с кодированием с одним активным состоянием можно избежать этой проблемы. Предстоит создать три новых столбца: is Apple, is Pineapple и is Orange. В каждой строке данных там, где плод существует, поставим 1, и для остальных плодов – 0.

– Конверсионные таблицы. Мы можем использовать их при переводе данных из одного стандарта в другой. Это будет иметь место там, где у нас есть данные в десятичной системе, и мы хотим перейти к метрической системе.

Эти шаги позволят значительно улучшить качество данных.

В этом деле также помогут инструменты автоматизации, например, от таких компаний, как SAS, Oracle, IBM, Lavastorm Analytics и Talend. Кроме того, существуют проекты с открытым исходным кодом, такие как OpenRefine, plyr и reshape.

И даже несмотря на это, данные не будут идеальными. Ни один источник данных не бывает таким. Скорее всего, все еще останутся пробелы и неточности.

Вот почему нужно подходить к делу творчески. К примеру, вот что сделал Эял Лифшиц (Eyal Lifshitz), генеральный директор компании BlueVine. Его компания использует ИИ для финансирования малого бизнеса. По его словам, "одним из наших источников данных является кредитная информация наших клиентов. Но мы обнаружили, что владельцы малого бизнеса неверно определяют свой тип бизнеса. А это может означать плохие результаты для нашего андеррайтинга. Для решения этой проблемы мы соскребаем данные с веб-сайта клиента с помощью алгоритмов искусственного интеллекта, которые помогают идентифицировать индустрию клиента".

Подходы к очистке данных также будут зависеть от вариантов использования проекта, разрабатываемого на основе ИИ. Например, если мы строим систему для предсказательного технического сопровождения на производстве, то главная трудность будет в обработке широких вариаций от разных датчиков. Как возможный результат, большой объем данных будет иметь малую ценность и будет в основном представлять шум.

Сколько данных нужно для искусственного интеллекта?

Чем больше данных, тем лучше, верно? Так обычно и бывает.

Возьмем, к примеру, так называемый феномен Хьюза (Hughes). Он постулирует, что при добавлении признаков в модель результативность модели, как правило, увеличивается.

Но количество не есть последняя инстанция. Может наступить момент, когда данные начнут деградировать. Имейте в виду, что мы можем столкнуться с так называемым проклятием размерности. По словам Чарльза Исбелла (Charles Isbell), профессора и старшего адъюнкт-декана Школы интерактивных вычислений в Технологическом институте Джорджии, "по мере роста числа признаков или размерностей объем данных, которые нам нужно точно обобщить, растет экспоненциально".

Каковы практические последствия этого? Получить хорошую модель станет просто невозможным, т. к. может не хватить данных.

Вот почему, когда речь заходит о таких приложениях, как видеораспознавание, проклятие размерности бывает весьма проблематичным. Даже во время анализа изображений RGB число размерностей составляет примерно 7500. Только представьте, насколько интенсивным будет этот процесс с использованием реальновременного видео высокой четкости.

Дополнительные термины и понятия относительно данных

Занимаясь анализом данных, нужно знать основные термины. Вот наиболее важные из них.

Категориальные данные – это данные, которые не имеют числового смысла. Вместо этого они имеют текстовый смысл, как описание группы (расы и пола). Хотя можно назначить каждому элементу числа.

Тип данных – это вид информации, которую переменная представляет, например, булево число, целое число, строка или число с плавающей запятой.

Описательная аналитика – это анализ данных с целью более глубокого понимания текущего состояния бизнеса. Некоторые его примеры включают измерение того, какие продукты продаются лучше, или определение рисков в службе поддержки клиентов. Существует ряд традиционных программных средств для описательной аналитики, таких как BI-приложения (бизнес-информации и аналитики).

Диагностическая аналитика – это запрос данных, позволяющих увидеть, почему что-то произошло. Этот тип аналитики использует такие методы, как глубинный анализ данных, деревья решений и корреляции.

ETL (extraction, transformation, and load – извлечение, трансформация и загрузка) – это форма интеграции данных, которая обычно используется в хранилище данных.

Признак – это столбец данных.

Экземпляр – это строка данных.

Метаданные – это данные о данных, т. е. описания. Например, музыкальный файл может иметь метаданные, такие как размер, длина, дата закачивания на сервер, комментарии, жанр, исполнитель и т. д. Этот тип данных может оказаться весьма полезным для проекта на основе ИИ.

Числовые данные – это любые данные, которые могут быть представлены числом. Но числовые данные могут иметь две формы. Существуют дискретные данные, представляющие собой целое число, т. е. число без десятичной запятой. Затем, есть непрерывные данные, которые имеют поток, скажем температуру или время.

OLAP (Online Analytical Processing – онлайновая аналитическая обработка) – это технология, позволяющая анализировать информацию из различных баз данных.

Порядковые данные – это смесь числовых и категориальных данных. Распространенным их примером является пятизвездочный рейтинг в Amazon.com. С ним ассоциированы и звезда, и число.

Предсказательная аналитика включает в себя использование данных для составления прогнозов. Ее модели обычно бывают очень сложными и опираются на подходы из области ИИ, такие как машинное обучение. В целях эффективности важно обновлять опорную модель новыми данными. Некоторые инструменты для предсказательной аналитики включают в себя методы машинного обучения, такие как регрессии.

Прескриптивная аналитика. Речь идет об использовании больших данных для принятия более эффективных решений. Это связано не только с предсказанием результатов, но и с пониманием рациональных оснований. И именно здесь ИИ играет большую роль.

Скалярные переменные – это переменные, которые содержат единичные значения, такие как имя или номер кредитной карты.

Транзакционные данные – это данные о финансовых, деловых и логистических действиях. Примеры включают платежи, счета-фактуры и страховые требования.

Ключевые моменты

Структурированные данные помечаются и форматируются, а затем часто хранятся в реляционной базе данных или электронной таблице.

Неструктурированные данные – это информация, которая не имеет предопределенного форматирования.

Полуструктурированные данные имеют некоторые внутренние теги, которые помогают в их классификации.

Большие данные описывают способ обработки огромных объемов информации.

Реляционная база данных основана на связях между данными. Но эта структура может оказаться трудной в работе для современных приложений, таких как ИИ.

База данных NoSQL имеет более свободную форму, основанную на документной модели. Это позволило данной технологии лучше справляться с неструктурированными и полуструктурированными данными.

Процесс CRISP-DM обеспечивает способ пошагового управления данными по проекту, в котором шаги включают понимание бизнеса, понимание данных, подготовку данных, моделирование, оценивание и развертывание.

Объем данных, безусловно, имеет важность, но также необходимо много работать над качеством. Даже незначительные ошибки могут оказать огромное влияние на результаты работы модели ИИ.

Вывод

Быть успешным с ИИ означает иметь культуру, основанную на данных. Это то, что было в центре внимания таких компаний, как Amazon.com, Google и Facebook. Принимая решения, они в первую очередь смотрят на данные. Кроме того, в организации должна быть обеспечена широкая доступность данных.

Без этого подхода успех с ИИ будет мимолетным, независимо от планирования. Возможно, это помогает объяснить, почему – согласно исследованию консалтинговой компании NewVantage Partners – около 77 % респондентов говорят, что "принятие бизнесом на вооружение" больших данных и ИИ остается серьезным вызовом.

Лекция 3. Машинное обучение. Добыча регулярностей из данных

Что такое машинное обучение?

После недолгой работы в Массачусетском технологическом институте (МТИ) и Bell Telephone Laboratories Артур Л. Сэмюэл (Arthur L. Samuel) в 1949 году присоединился к компании IBM в лаборатории Poughkeepsie Laboratory. Он разрабатывал приложения и среди них было одно, которое войдет в историю, — это его компьютерная игра в шашки. Данное приложение было первым примером автоматически обучающейся системы (в 1959 году Сэмюэл опубликовал об этом влиятельную статью).

Обращаясь к игре в шашки, он показал, как работает машинное самообучение, – другими словами, компьютер может учиться и совершенствоваться, обрабатывая данные без необходимости явного программирования. Это стало возможным благодаря использованию передовых статистических концепций, в особенности в области вероятностного анализа. Благодаря им компьютер мог учиться делать точные предсказания.

Эта идея была революционной, поскольку в то время разработка программно-информационного обеспечения была в основном связана со списком команд, которые следовали логически организованному рабочему процессу.

Для того чтобы понять, как работает машинное обучение, возьмем пример из телевизионного комедийного шоу "Кремниевая долина" на телевизионном канале HBO (Home Box Office). Инженер Цзянь-Ян предположительно должен был создать сногсшибательное приложение для еды. Для того чтобы натренировать это приложение, он должен был предоставить огромный набор фотографий еды. К сожалению, времени было в обрез, и приложение научилось распознавать только... хот-доги. Другими словами, приложение могло отвечать только "хот-дог" и "не хот-дог".

Этот эпизод довольно хорошо продемонстрировал машинное обучение. По сути, это процесс взятия помеченных данных и отыскания связей. Если мы будем тренировать систему с помощью хот-догов, таких как тысячи фотографий хот-догов, то она будет все лучше и лучше их распознавать.

Ниже мы рассмотрим ключевые статистические показатели и методы, которые необходимо знать о машинном обучении. Сюда входят стандартное отклонение, нормальное распределение, теорема Байеса, корреляция и извлечение признаков.

Стандартное отклонение

Стандартное отклонение измеряет среднее расстояние от среднего значения. На самом деле, нет необходимости учиться его вычислять (процесс вычисления состоит из несколько шагов), т. к. Excel или другое программно-информационное обеспечение может это сделать легко.

Для того чтобы понять суть стандартного отклонения, возьмем пример стоимости домов в жилом районе. Предположим, что средняя стоимость равна 145 тыс. долларов, а стандартное отклонение — 12 тыс. долларов. Это означает, что

стоимость на одно стандартное отклонение ниже среднего будет равна 133 тыс. долларов (145 000 - 12 000 долларов), а стоимость на одно стандартное отклонение выше среднего составит 157 тыс. долларов (145 000 + 12 000). Это дает нам возможность количественно оценить вариацию в данных. То есть существует разброс в 24 тыс. долларов от среднего значения.

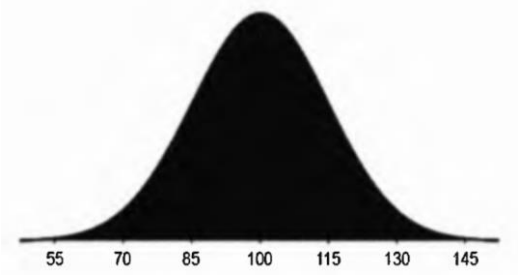
Теперь посмотрим на данные: если, скажем, Марк Цукерберг переезжает в этот район, и в результате среднее значение подскакивает до 850 тыс. долларов, а стандартное отклонение составит 175 тыс. долларов. Но отражают ли эти статистические показатели рыночные оценки? Не совсем. Покупка дома Цукербергом является выбросом. В этой ситуации лучшим подходом будет исключить его дом из рассмотрения.

Нормальное распределение

При построении нормального распределения на графике оно выглядит как колокол (вот почему другое его название — "колоколообразная кривая"). Оно представляет собой сумму вероятностей для переменной. Интересно, что нормальная кривая распространена в естественном мире, поскольку она отражает распределение таких вещей, как рост и вес.

Общий подход при интерпретации нормального распределения заключается в использовании правила 68–95–99,7. Оно означает, что 68 % элементов данных будут находиться в пределах одного стандартного отклонения, 95 % — в пределах двух стандартных отклонений и 99,7 % — в пределах трех стандартных отклонений.

Одним из способов его понять — взглянуть на баллы IQ. Предположим, что средний балл равен 100 и стандартное отклонение — 15. Мы бы получили его для трех стандартных отклонений.



Обратите внимание, что пик на этом графике является средним числом баллов. Так, если человек имеет IQ, равный 145 баллам, то только 0,15% будут иметь более высокий балл.

Указанная кривая может иметь разные формы в зависимости от вариации в данных. Например, если наши данные IQ содержат большое число гениев, то распределение будет смещаться вправо.

Теорема Байеса

Для того чтобы понять, как работает теорема Байеса, давайте возьмем пример. Исследователь придумывает тест на некоторый тип гриппа, и этот тест оказывается

точным в 80% случаев. Такой результат называется истинным утверждением (истинноположительным результатом).

Но в 9,6% случаев тест идентифицирует человека как больного гриппом, даже если его у него нет, и такой результат называется ложным утверждением (ложноположительным результатом).

Имейте в виду, что в некоторых тестах на наркотики этот процент может быть выше, чем показатель точности!

И наконец, 1% населения имеет грипп.

В свете всего этого, если врач использует указанный тест на одном человеке, и он показывает, что у него грипп, какова вероятность того, что он действительно болен гриппом? Так вот, теорема Байеса покажет путь. Это вычисление использует такие факторы, как точность, ложные утверждения и уровень населения, получая на выходе вероятность.

Шаг 1: точность 80% х вероятность наличия гриппа (1%) = 0,008.

Шаг 2: вероятность отсутствия гриппа (99%) х 9,6% ложных утверждений = 0,09504.

Шаг 3: подставим вышеуказанные числа в следующее выражение:

0,008 / (0,008 + 0,09504) = 7,8%.

Звучит как-то странно, верно? Определенно. В конце концов, как получается, что тест, который точен на 90%, имеет только 7,8% вероятности быть правильным? Но помните, что показатель точности основан на показателе тех, кто болен гриппом. И это число является небольшим, т. к. только 1% населения болеет гриппом.

Более того, тест по-прежнему выдает ложные утверждения. Таким образом, теорема Байеса – это способ обеспечить более глубокое понимание результатов, что имеет решающее значение для таких систем, как ИИ.

Корреляция

Алгоритм машинного обучения нередко включает в себя некий тип корреляции между данными. Количественный способ ее описать— использовать корреляцию Пирсона, которая показывает силу связи между двумя переменными, находящуюся в интервале от –1 до 1 (это коэффициент).

Вот как она работает:

– больше 0 – это такая ситуация, когда увеличение одной переменной приводит к увеличению другой. Предположим, что существует корреляция между доходами и расходами в размере 0,9. Если доход увеличивается на 1000 долларов, то расходы увеличиваются на 900 долларов (1000 долларов х 0,9);

– 0 – между этими двумя переменными нет корреляции;

– меньше 0 – любое увеличение переменной означает уменьшение другой, и наоборот. Корреляция описывает обратную связь.

Тогда какая корреляция считается сильной? Вообще, она считается сильной, если коэффициент равен +0,7 или около того. Если же он меньше 0,3, то корреляция является слабой.

Все это напоминает старую поговорку: «корреляция – это не обязательно каузация» (т. е. причинно-следственная зависимость). Однако в том, что касается машинного обучения, эта идея может быть легко проигнорирована и привести к вводящим в заблуждение результатам.

Например, существует много корреляций, которые являются просто случайными. На самом деле, некоторые из них могут быть прямо-таки комичными. Взгляните на следующие ниже мнимые корреляции из веб-сайта Tylervigen.com:

– уровень разводов в штате Мэн имеет корреляцию в размере 99,26% с потреблением маргарина на душу населения;

– импорт сырой нефти США из Норвегии имеет корреляции в размере 95,4% с водителями, погибшими при столкновении с железнодорожным поездом.

Для этого есть название: закономерничество, или шаблонизм (pattemicity), т. е. тенденция находить регулярности в бессмысленном шуме.

Извлечение признаков

Примером этого может служить компьютерная модель, которая идентифицирует мужчину или женщину по фотографии. Человеку это сделать довольно легко и быстро. Это нечто интуитивное.

Но если бы кто-то попросил нас описать различия, смогли бы мы это сделать? Для большинства людей это было бы трудной задачей. Однако, если мы хотим построить эффективную автоматически обучающуюся модель, нам нужно правильно выполнить процедуру извлечения признаков, и эта работа может быть субъективной.

Извлечение признаков также имеет некоторые нюансы. Во-первых, это потенциальное систематическое смешение из-за предвзятости. Например, есть ли у нас предубеждения относительно того, как выглядит мужчина или женщина? Если это так, то это может привести к моделям, которые дают неправильные результаты.

Ввиду всего этого неплохая идея иметь группу экспертов, которые могут выделить правильные признаки. И если выработать признаки из рассматриваемой задачи оказывается слишком сложно, то машинное обучение, вероятно, не является для нее хорошим вариантом.

Есть и другой подход, который стоит того, чтобы его рассмотреть: глубокое обучение. Сюда входят многосложные модели, которые отыскивают признаки в данных. На самом деле это одна из причин того, что глубокое обучение стало крупным инновационным прорывом в ИИ.

Что можно сделать с помощью машинного обучения?

Поскольку машинное обучение существует уже на протяжении десятилетий, эта мощная технология нашла много применений. Она также помогает получать очевидные преимущества с точки зрения экономии затрат, возможностей получения дохода и мониторинга рисков.

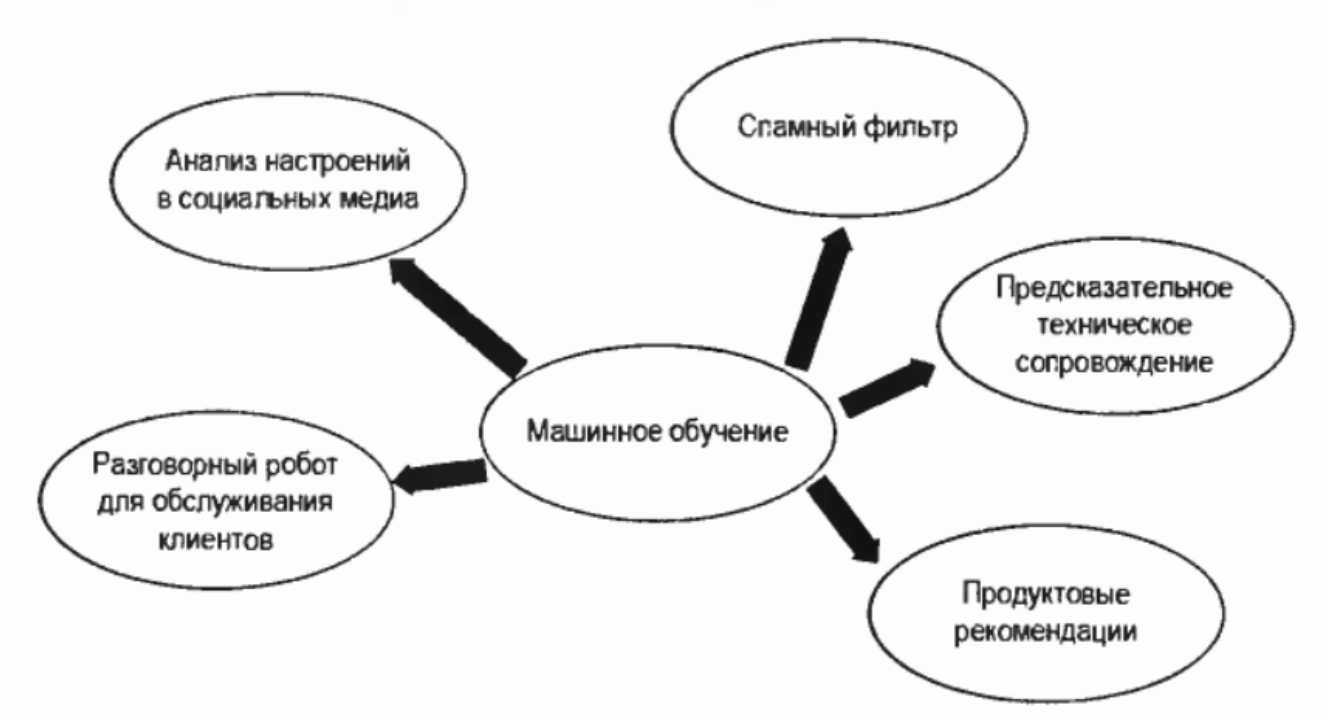
Вот несколько примеров.

– Предсказательное техническое сопровождение. Оно выполняет мониторинг датчиков с целью предсказания времени, когда оборудование может выйти из строя. Оно не только помогает снизить затраты, но и сокращает время простоя и повышает безопасность.

– Подбор персонала.

– Покупательский опыт клиентов.

– Знакомства.



Процесс машинного обучения

Для того чтобы добиться успеха в применении машинного обучения к поставленной задаче, важно использовать системный подход.

Прежде всего нужно пройти процесс обработки данных. Когда он будет закончен, неплохо выполнить визуализацию данных. Какие они: они главным образом разбросаны или же в них имеются какие-то регулярности? Если ответ является положительным, то эти данные могут быть хорошим кандидатом для машинного обучения.

Цель процесса машинного обучения — создание модели, в основе которой лежит один или несколько алгоритмов. Мы разрабатываем модель, тренируя ее. Цель состоит в том, чтобы результирующая модель обеспечивала высокую степень предсказуемости.

Теперь рассмотрим этот процесс подробнее.

Шаг 1. Рандомизировать данные

Если данные отсортированы, то это может исказить результаты. То есть автоматически обучающийся алгоритм может обнаружить это как регулярность! Вот почему неплохо выполнять рандомизацию порядка данных.

Шаг 2. Выбрать модель

Вам необходимо подобрать алгоритм. Он будет выбран на основе предположения, которое объединит в себе процесс проб и ошибок.

Шаг 3. Натренировать модель

Тренировочные данные, которые составят около 70% всего набора данных, будут использоваться для создания связей в алгоритме. Например, предположим, что мы строим автоматически обучающуюся систему для определения стоимости подержанного автомобиля. Среди признаков модели будут находиться год выпуска, марка, модель, пробег и состояние. Обработав эти тренировочные данные, алгоритм вычислит веса для каждого из этих факторов.

Шаг 4. Оценить модель

Мы собираем тестовые данные, которые составляют оставшиеся 30% набора данных. Они должны быть репрезентативными по размаху и типу информации в тренировочных данных.

С помощью тестовых данных можно увидеть, насколько алгоритм является точным. В примере с подержанными автомобилями мы получим ответ о том, согласуются ли полученные рыночные стоимости с тем, что происходит в реальном мире.

Шаг 5. Отрегулировать модель

На этом шаге мы можем откорректировать значения параметров в алгоритме. Это нужно для того, чтобы посмотреть, сможем ли мы получить оптимальные результаты.

Применение алгоритмов

Некоторые алгоритмы вычисляются довольно просто, в то время как другие требуют сложных шагов и математики. Хорошей новостью является то, что вам обычно не нужно вычислять алгоритм самостоятельно, потому что имеется целый ряд языков программирования, таких как Python и R, которые упрощают этот процесс.

Несмотря на то что имеются сотни алгоритмов машинного обучения, их можно разделить на четыре главенствующие категории: контролируемое самообучение, неконтролируемое самообучение, подкрепляемое самообучение и полуконтролируемое самообучение. Мы рассмотрим каждую из них.

Контролируемое самообучение

Контролируемое самообучение использует помеченные данные. Например, предположим, что у нас есть набор фотографий тысяч собак. Данные считаются помеченными, если каждая фотография идентифицирует породу каждой собаки. В большинстве случаев это облегчает анализ, т. к. мы можем сравнить наши результаты с правильным ответом.

Один из ключевых моментов в контролируемом самообучении — наличие крупного объема данных. Это помогает уточнять модель и получать более точные результаты.

Правда, существует одна большая трудность: реальность такова, что большая часть располагаемых данных не помечена. В дополнение к этому при наличии массивного набора данных на создание меток может уходить много времени.

Неконтролируемое самообучение

Неконтролируемое самообучение относится к ситуациям, когда мы работаем с непомеченными данными. Это означает, что мы будем использовать алгоритмы глубокого обучения для обнаружения регулярностей.

Подкрепляемое самообучение

Когда мы были еще детьми и хотели поиграть в новый командный вид спорта, скорее всего, мы не читали инструкцию. Вместо этого мы наблюдали за тем, как это делают другие люди, и пытались понять, что происходит. В некоторых ситуациях мы допускали ошибки и теряли мяч, и товарищи по команде выражали свое неудовольствие. Но в других случаях мы делали правильные ходы и забивали. Благодаря этому процессу проб и ошибок учеба совершенствовалась на основе положительного и отрицательного подкрепления.

На высоком уровне это аналогично подкрепляемому самообучению. Оно было ключевым для нескольких наиболее заметных достижений в области ИИ, таких как: игры, робототехника.

Полуконтролируемое самообучение

Это смесь контролируемого и неконтролируемого самообучения. Оно происходит, когда у нас есть небольшой объем непомеченных данных. Но мы можем использовать системы глубокого обучения для трансляции неконтролируемых данных в контролируемые данные. Этот процесс называется псевдоразметкой. После этого можно применить соответствующие алгоритмы.

Интересным примером использования полуконтролируемого самообучения является интерпретация магнитно-резонансной томограммы (МРТ). Рентгенолог может сначала пометить снимки, и после этого система глубокого обучения может найти остальные регулярности.

Ключевые моменты

Машинное обучение, корни которого уходят в 1950-е годы, — это та область, где компьютер может учиться, не будучи явно запрограммированным. Вместо этого он будет поглощать и обрабатывать данные, используя сложные статистические методы.

Выброс — это данные, которые находятся далеко за пределами остальных чисел в наборе данных.

Стандартное отклонение измеряет среднее расстояние от среднего значения.

Нормальное распределение, имеющее форму колокола, представляет собой сумму вероятностей для переменной.

Теорема Байеса — это изощренный статистический метод, обеспечивающий более глубокий взгляд на вероятности.

Истинное утверждение (истинноположительный результат) — это утверждение, при котором модель делает правильное предсказание. Ложное утверждение (ложноположительный результат), с другой стороны, — это утверждение, при котором модельное предсказание показывает, что результат является истинным, даже если это не так.

Корреляция Пирсона показывает силу связи между двумя переменными в интервале от –1 до 1.

Извлечение признаков, или выработка признаков, описывает процесс отбора переменных для модели. Оно является очень важным, поскольку даже одна неверная переменная может оказать существенное влияние на результаты.

Тренировочные данные — это данные, которые используются для создания связей в алгоритме. С другой стороны, тестовые данные применяются для оценивания модели.

Контролируемое самообучение использует помеченные данные для создания модели, в то время как неконтролируемое самообучение этого не делает. Существует также полуконтролируемое самообучение, в котором применяется сочетание обоих подходов.

Подкрепляемое самообучение — это способ натренировать модель, поощряя точные предсказания и наказывая те, которые таковыми не являются.

Вывод

Приведенные выше алгоритмы могут становиться очень сложными и требуют по-настоящему сильных технических навыков. Но важно не слишком увязнуть в технологии. В конце концов, центральным является отыскание способов использования машинного обучения для достижения четких целей.

Лекция 4. Глубокое обучение. Революция в искусственном интеллекте

Разница между глубоким и машинным обучением

Часто существует путаница между глубоким обучением и машинным обучением. И это вполне оправданно. Обе темы являются довольно сложными, и у них много общего.

Поэтому в целях понимания разницы между ними давайте сначала рассмотрим два высокоуровневых аспекта машинного обучения и их связь с глубоким обучением. Прежде всего, хотя и то и другое обычно требует больших объемов данных, оба типа, как правило, различаются.

Рассмотрим следующий пример: предположим, у нас есть фотографии тысяч животных и мы хотим создать алгоритм отыскания лошадей. Так вот, машинное обучение не может анализировать фотографии, взятые как таковые; как раз напротив, данные должны быть помечены. Затем алгоритм машинного обучения будет натренирован распознавать лошадей в ходе процесса, именуемого контролируемым самообучением.

Несмотря на то что машинное обучение, скорее всего, даст хорошие результаты, оно все равно будет иметь ограничения. Учитывая это, не лучше ли обратиться к пикселам самих фотографий и отыскать регулярности? Несомненно.

Для этого в машинном обучении необходимо использовать процесс, именуемый извлечением признаков. Это означает, что мы должны сформулировать такие характеристики лошади, как форма, копыта, окрас и рост, которые алгоритмы затем попытаются идентифицировать.

Опять же, этот подход неплох, но он далек от совершенства. Что делать, если признаки не соответствуют действительности либо не учитывают выбросы или исключения? В таких случаях точность модели, скорее всего, пострадает. В конце концов, существует много подвидов лошадей. Извлечение признаков также имеет недостаток в том, что оно игнорирует крупный объем данных. В некоторых случаях использования извлекать признаки становится чрезмерно сложно – если не невозможно. Взгляните на компьютерные вирусы. Их структуры и шаблоны, именуемые сигнатурами, постоянно изменяются, что позволяет им проникать в вычислительные системы. Но при извлечении признаков человек должен каким-то образом предвосхищать это, что в принципе не осуществимо на практике.

А вот с помощью глубокого обучения эти задачи можно решать.

Этот подход анализирует все данные – пиксел за пикселом – и затем отыскивает связи с помощью нейронной сети, которая имитирует человеческий мозг.

Так что же такое глубокое обучение?

Глубокое обучение – это подобласть машинного обучения. Этот тип системы позволяет обрабатывать огромные объемы данных с целью отыскания связей и

регулярностей, которые люди часто не в состоянии обнаружить. Слово «глубокий» относится к числу скрытых слоев в нейронной сети, которые обеспечивают большую часть способностей к самообучению.

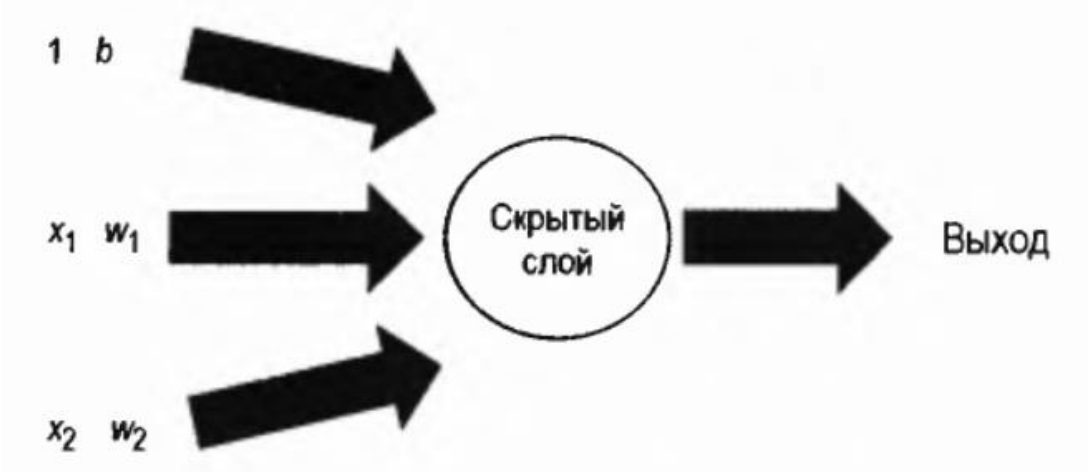
Важно помнить, что глубокое обучение все еще находится на ранних стадиях развития и коммерциализации. Так, примерно в 2015 году компания Google начала использовать эту технологию для своей поисковой системы.

Как мы видели в первой лекции, история нейронных сетей была полна приливов и отливов. Именно Фрэнк Розенблатт был первым, кто создал персептрон, представлявший собой довольно простую систему. Но реальный научный прогресс в нейронных сетях не происходил вплоть до 1980-х годов, в частности, за счет инновационных прорывов с участием обратного распространения, сверточных и рекуррентных нейронных сетей. Для того чтобы глубокое обучение оказало влияние на реальный мир, потребовался ошеломляющий рост объема данных, например, из Интернета, и резкий рост вычислительной мощности.

Искусственные нейронные сети (ANN-сети)

На самом базовом уровне искусственная нейронная сеть (artificial neural network, ANN) – это функция, которая включает в себя блоки (также именуемые нейронами, персептронами, элементами или узлами). Каждый блок будет иметь величину и вес, который указывает на относительную важность, и будет входить в скрытый слой. Скрытый слой использует функцию, результат которой становится выходом. Существует также еще одна величина, именуемая смещением, которая является константой и используется при вычислении этой функции.

Такой тип тренировки модели называется нейронной сетью прямого распространения. Другими словами, сигнал движется от входа в скрытый слой и далее к выходу. Он не возвращается в цикле назад. Но он мог бы перейти в новую нейронную сеть, причем выход может стать входом.



Обратное распространение

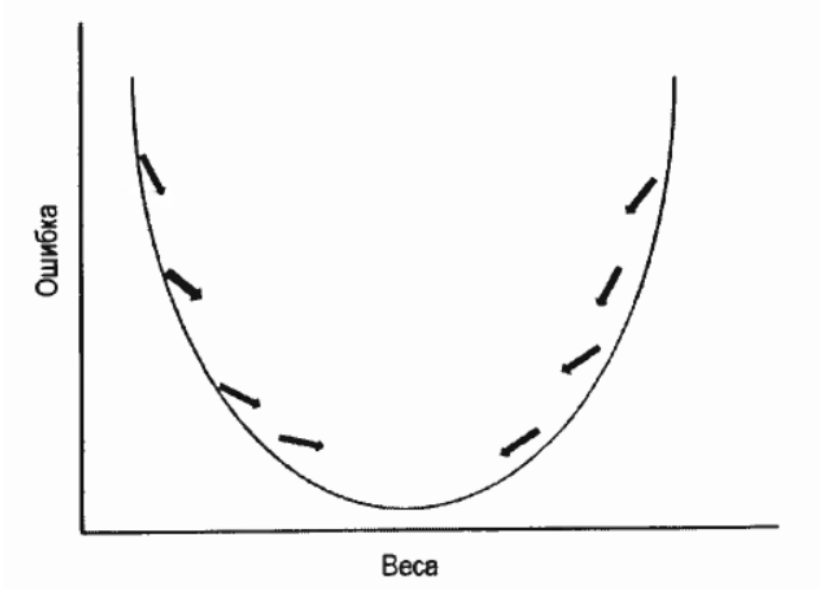
Одним из главных недостатков искусственных нейронных сетей является процесс корректировки весов в модели. Традиционные подходы, такие как

применение мутационного алгоритма, использовали случайные значения, и этот алгоритм оказался времязатратным.

С учетом этого исследователи искали альтернативы, такие как обратное распространение. Этот технический прием существовал с 1970-х годов, но не вызвал большого интереса, т. к. его производительность была недостаточной. Но Дэвид Румелхарт, Джеффри Хинтон и Рональд Уильямс понимали, что обратное распространение все-таки имеет потенциал, если этот прием будет усовершенствован.

Нет ничего удивительного в том, что алгоритм обратного распространения влечет за собой много математических формул. Но если свести все к простой идее, то речь идет о корректировке нейронной сети при обнаружении ошибок, а затем итеративной передаче новых значений через нейронную сеть повторно. По существу, этот процесс предусматривает незначительные изменения, которые продолжают оптимизировать модель.

Сначала имеется высокий уровень ошибок, потому что веса являются слишком крупными. Но в результате выполнений итераций по сети ошибки будут постепенно уменьшаться. Однако слишком большое число итераций может привести к росту ошибок. Другими словами, цель обратного распространения состоит в том, чтобы отыскать срединную точку.



Типы нейронных сетей

Самый базовый тип нейронной сети – это полносвязная нейронная сеть. Как следует из ее названия, именно здесь все нейроны имеют связи от слоя к слою. Эта сеть на самом деле является довольно популярной, т. к. она означает, что при создании модели не нужно много рассуждать.

Какие еще существуют нейронные сети? Самыми распространенными из них являются рекуррентная нейронная сеть (RNN-сеть, recurrent neural network), сверточная нейронная сеть (CNN-сеть, convolutional neural network) и генеративная состязательная сеть (GAN-сеть, generative adversarial network), которые мы рассмотрим далее.

Рекуррентная нейронная сеть (RNN-сеть)

С рекуррентной нейронной сетью (RNN-сетью) функция обрабатывает не только входы, но и входы, предыдущие во времени.

Примером тому является то, что происходит, когда мы вводим символы в приложение обмена сообщениями. Когда мы начнем печатать, система будет предсказывать слова. Поэтому если мы наберем «He», то компьютер предложит «He», «Hello» и «Here's».

RNN-сеть – это, по сути, цепочка нейронных сетей, которые питаются друг от друга на основе сложных алгоритмов.

Как один из примеров, приложение Google Translate не раз выручало врачей, работающих с пациентами, говорящими на других языках. Согласно исследованию Калифорнийского университета в Сан-Франциско (UCSF), опубликованному в медицинском журнале JAMA Internal Medicine, точность перевода с английского на испанский составила 92%.

Сверточная нейронная сеть (CNN-сеть)

В интуитивном плане имеет смысл организовывать структуру нейронной сети так, чтобы все ее блоки были соединены между собой. Это хорошо работает со многими приложениями.

Однако существуют сценарии, где такая организация нейросети является далеко не оптимальной, например, в случае с распознаванием изображений. Только представьте, насколько сложной будет модель, где каждый пиксел является блоком нейросети. Она быстро станет неуправляемой.

Для того чтобы со всем этим справиться, можно применить сверточную нейронную сеть (CNN-сеть). Истоки этой нейросети восходят к 1998 году, когда профессор Янн Лекун опубликовал работу под названием "Градиентное самообучение применительно к распознаванию документов". Несмотря на свои сильные озарения и инновационные прорывы, эта модель набрала мало оборотов. Но по мере того, как в 2012 году глубокое обучение начало демонстрировать значительный прогресс, исследователи пересмотрели эту модель.

Лекун получил свое вдохновение для CNN-сети от лауреатов Нобелевской премии Дэвида Хьюбела (David Hubei) и Торстена Визеля (Torsten Wiesel), которые изучали нейроны зрительной коры. Эта система берет изображение с сетчатки и обрабатывает его на разных уровнях — от простого к более сложному. Каждый уровень называется сверткой. Например, первый уровень будет состоять из идентификаций линий и углов; далее зрительная кора будет находить формы, а затем она будет обнаруживать объекты.

Давайте рассмотрим пример: предположим, мы хотим построить модель, которая может идентифицировать букву. На входе CNN-сеть будет иметь изображение, которое имеет 3072 пикселов. Каждый пиксел будет иметь значение от 0 до 255, указывающее на совокупную интенсивность. Используя CNN-сеть,

компьютер будет проходить через многочисленные вариации, чтобы идентифицировать признаки.

Первым идет сверточный слой, который представляет собой фильтр, сканирующий изображение. В нашем примере это может быть матрица 5x5 пикселов. В результате будет создано признаковое отображение, представляющее собой длинный массив чисел. Далее модель применит к изображению дополнительные фильтры. Делая это, CNN-сеть идентифицирует линии, ребра и фигуры – и все это выражается в числах. С помощью различных выходных слоев модель будет использовать редукцию (сведение), которая комбинирует их, генерируя единый выход, а затем создает полносвязную нейронную сеть.

Генеративные состязательные сети (GAN-сети)

Йен Гудфеллоу (Ian Goodfellow), получивший степень магистра компьютерных наук в Стэнфорде и докторскую степень в области машинного обучения в Монреальском университете, продолжал работать в компании Google. В возрасте 20 лет с небольшим он стал соавтором одной из лучших книг в области ИИ под заголовком «Глубокое обучение», а также внес инновационные изменения в веб-службу Google Карты.

Но именно в 2014 году он совершил свой самый впечатляющий инновационный прорыв. На самом деле это произошло в пабе в Монреале, когда он разговаривал со своими друзьями о том, как глубокое обучение может создавать фотографии. В то время подход состоял в использовании генеративных моделей, но они часто были размытыми и бессмысленными.

Гудфеллоу понял, что должна быть более веская причина. Так почему бы не использовать теорию игр? То есть две модели конкурируют друг с другом в тесном контуре обратной связи. Это также можно делать с непомеченными данными.

Вот базовый рабочий процесс:

– генератор – эта нейронная сеть создает массу новых построений, таких как фотографии или предложения языка;

– дискриминатор – эта нейронная сеть смотрит на эти построения, чтобы увидеть, какие из них являются реальными;

– коррективы – с этими двумя результатами новая модель изменяет построения, чтобы сделать их как можно более реалистичными. В результате многочисленных итераций дискриминатор больше становится ненужным.

Применение технологии глубокого обучения

Пример использования: обнаружение болезни Альцгеймера

Несмотря на десятилетия исследований, лекарство от болезни Альцгеймера пока не найдено. Хотя надо признать, что ученые все-таки разработали препараты, которые замедлили прогрессирование данного заболевания.

В свете этого ранняя диагностика имеет решающее значение, и глубокое обучение потенциально может оказать большую помощь. Исследователи из отдела радиологии и биомедицинской визуализации UCSF использовали эту технологию

для анализа скринов головного мозга – из публичного набора данных Инициативы по нейровизуализации болезни Альцгеймера – и для обнаружения изменений в уровнях глюкозы.

Результатом стала модель, которая позволяет диагностировать болезнь Альцгеймера за шесть лет до клинического диагноза. Один из тестов показал точность 92%, другой – 98%.

Аппаратное обеспечение для технологии глубокого обучения

Что касается чиповых систем для технологии глубокого обучения, то графические процессоры были первоочередным вариантом выбора. Но по мере того, как ИИ становится сложнее, например, благодаря GAN-сетям, а наборы данных намного крупнее, безусловно, появляется больше возможностей для новых подходов. У компаний также есть собственные потребности, например, в отношении функций и данных. В конце концов, приложение для потребителя обычно сильно отличается от того, которое ориентировано на предприятие.

В результате некоторые мегатехнологические компании разрабатывают собственные чипсеты.

Google. Летом 2018 года компания анонсировала третью версию своего тензорного процессора (Tensor Processing Unit, TPU; первый чип был разработан в 2016 году). Эти чипы имеют настолько высокую мощность, обрабатывая более 100 петафлопсов во время тренировки моделей, что в центрах обработки данных должно иметься жидкостное охлаждение.

Amazon. В 2018 году эта компания анонсировала технологию AWS Inferential Данная технология, появившаяся в результате приобретения компании Annapurna в 2015 году, ориентирована на обработку сложных операций логического вывода.

Tesla. Илон Маск (Elon Musk) разработал собственный чип ИИ. Он имеет 6 млрд. транзисторов и может обрабатывать 36 трлн. операций в секунду.

Когда использовать глубокое обучение?

Из-за мощи глубокого обучения возникает соблазн использовать прежде всего именно эту технологию, когда ведется работа над проектом в области ИИ. Такой выбор может быть большой ошибкой. Глубокое обучение по-прежнему имеет узкоспециализированные варианты использования, такие как для наборов текстовых, видео- и фотографических данных, а также наборов данных временных рядов. Существует также потребность в крупных объемах данных и мощных компьютерных системах.

И да, технология глубокого обучения работает лучше, когда результаты могут быть квантифицированы и верифицированы.

Тогда, где сценарий, в котором глубокое обучение не достигнет поставленной цели? Собственно, иллюстрацией этого является Чемпионат мира по футболу 2018 года в России, который выиграла Франция. Многие исследователи пытались предсказать исход всех 64 матчей, но результаты были далеки от точных:

– одна группа исследователей использовала модель консенсуса букмекеров, которая указывала на то, что победит Бразилия;

– другая группа исследователей использовала такие алгоритмы, как случайный лес и пуассоновское ранжирование, спрогнозировав, что победителем будет Испания.

Проблема здесь в том, что трудно отыскать правильные переменные, которые обладают предсказательной силой. На самом деле, модели глубокого обучения в основном не способны справиться со сложностью признаков некоторых событий, в особенности тех, которые имеют элементы хаотичности.

Ключевые моменты

Глубокое обучение, которое является подобластью машинного обучения, обрабатывает огромные объемы данных с целью обнаружения связей и регулярностей, которые люди часто не в состоянии обнаружить. Слово «глубокий» описывает число скрытых слоев.

Искусственная нейронная сеть (ANN) – это функция, которая включает в себя блоки, имеющие вес; эти блоки используются для предсказания значений в модели на основе ИИ.

Скрытый слой – это часть модели, которая обрабатывает входы.

Нейронная сеть прямого распространения содержит данные, которые передаются только от входного слоя в скрытый слой и далее в выходной слой. Результаты не передаются итеративно назад. И тем не менее они могут входить в еще одну нейронную сеть.

Обратное распространение – это изощренный технический прием для корректировки весов в нейронной сети. Этот подход имеет решающее значение для развития глубокого обучения.

Рекуррентная нейронная сеть (RNN-сеть) – это функция, которая обрабатывает не только вход, но и вход, предшествующий во времени.

Сверточная нейронная сеть (CNN-сеть) анализирует данные раздел за разделом (т. е. посверточно). Эта модель предназначена для сложных приложений, таких как распознавание изображений.

Генеративная состязательная сеть (GAN-сеть) – это сеть, в которой две нейронные сети конкурируют друг с другом в тесном контуре обратной связи. Результатом часто является создание нового объекта.

Вывод

Хотя Маркус указал на недостатки в глубоком обучении, факт остается фактом, что этот подход в области ИИ все еще чрезвычайно силен. Менее чем за 10 лет он произвел революцию в мире технологий, а также значительно повлиял на такие области, как финансы, робототехника и здравоохранение.

В связи с резким ростом инвестиций со стороны крупных технологических компаний и венчурных капиталовложений в модели будут появляться новые

инновации. Это также подтолкнет инженеров к получению степеней магистра, тем самым создавая добродетельный цикл инновационных прорывов.

Лекция 5. Роботизированная автоматизация процессов

Что такое РАП?

Термин «роботизированная автоматизация процессов» (РАП) звучит немного запутанно. Слово «роботизированный» не означает физических роботов; скорее наоборот, речь идет о программных роботах, или ботах.

Технология РАП позволяет использовать низкокодовые визуальные системы с поддержкой перетаскивания объектов для автоматизации рабочего процесса. Вот лишь несколько примеров:

– ввод, изменение и отслеживание кадровых документов, контрактов и информации о сотрудниках;

– выявление проблем с обслуживанием клиентов и принятие мер по их устранению;

– оформление страхового требования;

– отправка счетов-фактур;

– выдача сумм возмещения клиентам;

– согласование финансовой документации;

– передача данных из одной системы в другую;

– предоставление стандартных ответов клиентам.

Все это делается с помощью бота, многократно выполняющего рабочие процессы в приложении, скажем, в системах ERP (enterprise resource planning – планирования корпоративных ресурсов) или CRM (customer relationship management – управления взаимоотношениями с клиентами). Всё это можно делать даже с помощью программы РАП, записывающей шаги сотрудников, или с использованием технологии оптического распознавания символов (optical character recognition, OCR) для перевода рукописных заметок. Думайте о РАП как о цифровом сотруднике.

Существует две разновидности этой технологии.

– Несопровождаемая РАП-автоматизация. Это процесс, который полностью автономен, так как бот будет работать в фоновом режиме. Но это не значит, что человеческое вмешательство отсутствует. По-прежнему будет осуществляться вмешательство с целью управления исключительными ситуациями, когда бот сталкивается с чем-то, чего он не понимает.

– Роботизированная автоматизация рабочего стола (robotic desktop automation, RDA). Это место, где РАП-автоматизация помогает сотруднику с выполнением заданий или всей работы. Распространенным случаем ее использования является контактный центр. Например, когда поступает звонок, представитель может использовать роботизированную автоматизацию рабочего стола для оказания помощи в отыскании ответов, отправке сообщения, получения информации о профиле клиента и получении представления о том, что делать дальше. Эта технология помогает улучшить или повысить эффективность работника.

Достоинства и недостатки технологии РАП

Конечно, довольно много времени обычного сотрудника – в операционном отделе – тратится на рутинные задачи. Но с помощью технологии РАП компании часто получают высокую возвратность инвестиций (return in investment, ROI) при условии, что ее имплементация выполнена правильно.

Вот несколько других преимуществ.

– Удовлетворенность потребителя. Технология РАП означает минимальные ошибки, а также высокую скорость. Бот также работает в режиме 24/7. Это означает, что показатели удовлетворенности клиентов, например, индекс потребительской лояльности (net promoter score, NPS – дословно «чистый балл промоутера»), должны улучшаться. Обратите внимание, что все больше клиентов, таких как поколение двухтысячных, предпочитают иметь дело с приложениями / веб-сайтами, а не с людьми! Технология РАП также означает, что у представителей будет больше времени на то, чтобы тратить его на дополнительные задачи, вместо того чтобы заниматься скучными делами, на которые уходит время впустую.

– Масштабируемость. После того как бот создан, его можно быстро расширить, чтобы учесть всплески активности. Это может иметь решающее значение для сезонных предприятий, таких как розничная торговля.

– Соблюдение инструкций. Людям трудно отслеживать правила, предписания и законы. Хуже того, они часто меняются. Но с технологией РАП соблюдение инструкций встроено в процесс, и они всегда соблюдаются. Это может приносить большую выгоду с точки зрения избежания юридических проблем и штрафов.

– Глубокое понимание и аналитика. Платформы РАП следующего поколения оснащены сложными информационными панелями, которые фокусируются на ключевых для бизнеса показателях эффективности. Также можно настраивать оповещения при возникновении каких-либо проблем.

– Унаследованные системы. Компании, с незапамятных времен работающие на рынке, часто вязнут в старых информационно-вычислительных системах, что чрезвычайно затрудняет проведение цифровой трансформации. Но система РАН способна довольно хорошо работать с унаследованными информационно-вычислительными средами.

– Данные. Благодаря автоматизации данные намного чище, так как имеются минимальные ошибки ввода. Это означает, что со временем организации будут иметь более точное представление о своем бизнесе. Качество данных также повысит вероятность успеха имплементации ИИ.

Хотя все это выглядит прекрасно, у технологии РАП все же есть свои недостатки. Например, если наши текущие процессы являются неэффективными, и мы спешим имплементировать систему РАП, то мы будем, по существу, тиражировать плохой подход!

Вот почему перед имплементацией системы крайне важно выполнить оценку рабочих процессов.

Существуют, конечно, и другие потенциальные подводные камни, которые следует отметить.

– *Ломкость*. Система РАП может легко сломаться, если есть изменения в базовых приложениях. Это также может иметь место в случае внесения изменений в процедуры и правила. Следует признать, что новые системы лучше адаптируются и могут также использовать API. Но РАП – это вовсе не та деятельность, которая выполняется без человеческих рук.

– *Виртуализированные приложения*. Этот тип программно-информационного обеспечения, например, от компании Citrix, бывает трудно совместить с системами РАП, поскольку оно не может эффективно захватывать процессы. Причина в том, что данные хранятся на внешнем сервере, а результатами являются моментальные снимки на мониторе. И тем не менее для решения этой проблемы некоторые компании используют ИИ, например, компания UiPath. Указанная компания создала систему Pragmatic AI (Прагматичный ИИ), которая использует компьютерное зрение для интерпретации снимков экрана и записи процессов.

– *Специализация*. Многие инструменты РАП предназначены для деятельности общего назначения. Но существуют области, требующие специализации, например, финансы. В этом случае можно обратиться к нишевому программно-информационному приложению, которое с ними работает.

– *Тестирование*. Оно имеет абсолютно критическое значение. Скорее всего, мы захотим сначала выборочно попробовать некоторые транзакции и убедиться, что система РАП работает правильно. После этого можно приступить к более широкому развертыванию системы.

– *Владение*. Искушение состоит в том, чтобы имплементацией системы РАП и ее управлением владел информационно-вычислительный отдел. Но это не рекомендуется делать, и вот по какой причине. Системы РАП являются довольно низкотехнологичными. В конце концов, они могут разрабатываться непрограммистами. По этой причине для владения процессом идеально подходят бизнес-менеджеры, поскольку они обычно могут справляться с техническими проблемами, а также имеют более четкое представление о рабочих процессах сотрудников.

– *Сопротивление*. Перемены всегда даются с трудом. В случае с технологией РАП могут возникнуть опасения, что она вытеснит рабочие места. Это означает, что мы должны иметь четкий набор сообщений, которые сфокусированы на преимуществах данной технологии. Например, наличие системы РАП будет означать больше времени на то, чтобы сосредоточиться на важных вопросах, которые должны сделать работу человека интереснее и содержательнее.

РАП и искусственный интеллект

Хотя ИИ все еще находится на начальных фазах своего развития, он уже делает уверенные шаги с помощью инструментов РАП, которые приводят к появлению программно-информационных ботов на основе технологии когнитивной роботизированной автоматизации процессов (cognitive robotic process automation, CRPA).

И это имеет смысл. В конце концов, РАП-автоматизация нацелена на оптимизацию процессов и предусматривает крупные объемы данных. Поэтому

производители начинают имплементировать системы, основанные на машинном обучении, глубоком обучении, распознавании речи и обработке естественного языка. Лидерами в сфере когнитивной РАП-автоматизации являются компании UiPath, Automation Anywhere, Blue Prism, NICE Systems и Kiyon Systems.

Например, с помощью продукта компании Automation Anywhere бот может обрабатывать такие задачи, как извлечение счетов-фактур из электронных писем, что требует изощренной обработки текста. Указанная компания также имеет предварительно собранные интеграции со сторонними службами ИИ, такими как IBM Watson, AWS Машинное обучение и Google Облачный ИИ.

По словам Мукунда Шригопала (Mukund Srigopal), директора по маркетингу продуктов в компании Automation Anywhere, «в последние годы наблюдается широкое распространение служб с поддержкой ИИ, но бизнес часто с трудом справляется с их введением в эксплуатацию. РАП – это отличный способ впрыснуть способности ИИ в бизнес-процессы».

Вот ряд других способов, которыми когнитивная РАП-автоматизация может позволить использовать функции ИИ.

– Можно подключить разговорных роботов (чат-ботов) к своей системе, что позволит автоматизировать обслуживание клиентов.

– ИИ может отыскивать подходящий момент для отправки электронного письма или предупреждения.

– Интерактивный автоответчик (interactive voice response, IVR) с годами приобрел плохую репутацию. Проще говоря, клиенты не любят хлопоты, связанные с прохождением многочисленных шагов для решения проблемы. Но с когнитивной РАП-автоматизацией мы можем использовать так называемый динамический интерактивный автоответчик. Он персонализирует голосовые сообщения для каждого клиента, обеспечивая гораздо более качественный опыт.

– Обработка ЕЯ и анализ текста могут выполнять конвертирование неструктурированных данных в структурированные. Это может поднять эффективность технологии когнитивной РАП-автоматизации.

Ключевые моменты

Технология роботизированной автоматизации процессов (РАП) позволяет использовать низкокодовые визуальные системы с поддержкой перетаскивания объектов для автоматизации протекания рабочего процесса.

Несопровождаемая РАП-автоматизация предусматривает полную автоматизацию процесса.

Роботизированная автоматизация рабочего стола – это место, где технология РАП помогает сотруднику с выполнением заданий или всей работы.

Выгоды от внедрения РАП включают более высокую удовлетворенность клиентов, более низкую частоту ошибок, улучшенное соответствие правилам и регламентам и более легкую интеграцию с унаследованными системами.

Недостатки от внедрения РАП включают трудности с адаптацией к изменениям в опорных приложениях, проблемы с виртуализированными приложениями и сопротивление со стороны сотрудников.

РАП лучше всего работает там, где мы можем автоматизировать повторяющиеся, структурированные и рутинные процессы, такие как планирование, ввод / передача данных и следование правилам / рабочим процессам.

Во время имплементации технического решения с использованием РАП-автоматизации необходимо рассмотреть следующие шаги: определение подходящих для автоматизации функций, проведение анализа процессов, выбор поставщика платформы РАП и ее развертывание, а также создание команды по управлению платформой.

Центр передового опыта – это команда, которая управляет имплементацией системы на основе РАП.

Когнитивная РАП-автоматизация (CRPA) – это новая категория РАП, которая фокусируется на технологиях ИИ.

Вывод

Как видно из примера с компанией Microsoft, РАП-автоматизация может привести к значительной экономии. Но в целях понимания процессов все равно требуется проводить тщательное планирование. По большей части основное внимание следует уделять ручным и повторяющимся задачам, а не тем, которые в значительной степени зависят от суждений. Далее, важно организовать работу центра передового опыта для наблюдения за текущим управлением автоматизацией, который поможет в обработке исключений, сборе данных и отслеживании ключевых показателей эффективности.

Технология РАП также является отличным средством имплементации базового ИИ внутри организации. Поскольку в результате может быть обеспечена значительная возвратность инвестиций, она практически может стимулировать еще больше инвестиций в развитие этой технологии.

Лекция 6. Обработка естественного языка

Трудности обработки естественного языка

Язык является ключом к тесту Тьюринга, который предназначен для проверки ИИ. Кроме того, язык также отличает нас от животных.

Но данная область научных исследований является чрезвычайно сложной. Вот лишь некоторые трудности, связанные с обработкой ЕЯ.

– Язык часто бывает неоднозначным. Мы учимся говорить быстро и подчеркиваем смысл с помощью невербальных сигналов, нашей интонации или реакций на окружающую среду. Например, если мяч в игре пасуется в адрес другого игрока, то болельщики будут скандировать: «Давай!» Но система с поддержкой ЕЯ, скорее всего, этого не поймет, потому что она не может обрабатывать контекст ситуации.

– Язык меняется так же часто, как меняется мир. Согласно Оксфордскому словарю английского языка, в 2018 году появилось более 1100 слов, смыслов и подпунктов словарных статей (всего их более 829 тыс.).

– Во время разговора мы делаем грамматические ошибки. Это обычно не проблема, поскольку люди обладают большой способностью к умозаключениям. Однако это является серьезной проблемой для системы с поддержкой ЕЯ, так как слова и фразы могут иметь несколько смыслов (это называется полисемией).

Например, известный исследователь ИИ Джеффри Хинтон любит сравнивать словосочетания «recognize speech» (распознать речь) и «wreck a nice beach» (разрушить красивый пляж).

– Язык имеет акценты и диалекты.

– Смысл слов может меняться в зависимости, скажем, от использования сарказма или других эмоциональных реакций.

– Слова могут быть расплывчатыми. В конце концов, что значит слово «late» (поздний, опоздавший, покойный)?

– Многие слова имеют, по существу, один и тот же смысл, но включают в себя различные нюансы.

– Диалоги могут быть нелинейными и прерываться. Несмотря на все это, в технологии обработки ЕЯ были достигнуты большие успехи, как видно из таких приложений, как Siri, Alexa и Cortana. Большая часть прогресса также произошла в течение последнего десятилетия, движимая мощью глубокого обучения.

Сейчас может возникнуть путаница между человеческими и компьютерными языками. Разве компьютеры не могли понимать такие языки, как BASIC, C и C++, в течение многих лет? Это определенно так. Также верно, что в компьютерных языках есть английские слова, такие как if, then, let и print.

Но этот тип языка очень отличается от человеческого. Следует учесть, что компьютерный язык имеет ограниченный набор команд и строгую логику. Если использовать что-то неправильно, то это приведет к ошибке в исходном коде, а та приведет к сбою.

Понимание того, как искусственный интеллект переводит язык

Обработка ЕЯ находилась в центре внимания с самого начала научных исследователей ИИ. Но из-за ограниченной мощности компьютера ее возможности были довольно слабыми. Цель состояла в том, чтобы создать правила для интерпретации слов и предложений, но они оказались сложными и не очень масштабируемыми. В каком-то смысле обработка ЕЯ в ранние годы была в общих чертах похожа на компьютерный язык.

Но со временем сложилась общая структура обработки ЕЯ. Это было крайне важно, поскольку обработка ЕЯ имеет дело с неструктурированными данными, которые бывают непредсказуемыми и трудными для интерпретации.

Вот общий высокоуровневый взгляд на два ключевых шага.

– Очистка и предобработка текста. Сюда входит использование таких методов, как лексемизация, выделение основ слов и лемматизация для разбора текста.

– Понимание и генерация языка. Эта часть процесса, безусловно, является самой интенсивной, и как раз в ней часто используются алгоритмы глубокого обучения.

Шаг 1. Очистка и предобработка

На этапе очистки и предобработки необходимо выполнить три действия: разбить текст на лексемы, или токены (лексемизация), выделить основы слов (стемминг) и выделить леммы (лемматизация).

Лексемизация

Прежде чем заняться обработкой ЕЯ, текст нужно разобрать и разделить на различные части. Этот процесс называется лексемизацией (или токенизацией). Например, предположим, что у нас есть следующее предложение: «John ate four cupcakes» (Джон съел четыре кекса). Затем мы выделим все элементы и разберете их на категории.

После лексемизации выполняется нормализация текста. Она предусматривает конвертирование части текста так, чтобы облегчить его анализ, например, путем перевода в верхний или нижний регистр, удаления пунктуации и устранения сокращений.

Но это может легко привести к некоторым проблемам. Предположим, что у нас есть предложение, в котором имеется аббревиатура "A.I". Следует ли нам избавиться от точек? И если мы это сделаем, то будет ли компьютер знать, что означает "A I"? Вероятно, нет.

Интересно отметить, что на смысл может оказать значительное влияние даже регистр букв в словах. Посмотрите на разницу между словами «fed» и «Fed». Fed – это часто еще одно название Федеральной резервной системы США. Или в еще одном случае предположим, что у нас есть "us" и "US". О чем тут речь: о Соединенных Штатах?

Вот еще несколько других трудностей.

– Проблема пробела – это место, где два или более слов должны быть одной лексемой, потому что слова образуют составную фразу. Среди некоторых ее примеров такие, как «New York» и «Silicon Valley» (Кремниевая долина).

– Научные слова и фразы – обычно такие слова имеют дефисы, круглые скобки и греческие буквы. Если удалить эти символы, то система не будет в состоянии понять смысл слов и фраз.

– Неряшливый текст – давайте посмотрим правде в глаза, многие документы имеют грамматические и орфографические ошибки.

– Разбивка предложений – такие слова, как «Мг.» или «Mrs.», могут преждевременно заканчивать предложение из-за точки.

– Несущественные слова – это такие слова, которые на самом деле мало или совсем ничего не значат в предложении, как артикли the, a и an. Для того чтобы удалить их, можно использовать простой фильтр, состоящий из стоп-слов.

Как видите, очень легко можно неправильно разобрать предложения (а в некоторых языках, таких как китайский и японский, все может стать еще сложнее с синтаксисом). Но это имеет далеко идущие последствия. Поскольку лексемизация, как правило, является первым шагом, пара ошибок может каскадом пройти через весь процесс обработки ЕЯ.

Выделение основ слов

Выделение основ слов, или стемминг, описывает процесс сведения слова к его корню (основе или лемме), например, путем удаления аффиксов и суффиксов. Оно было по-настоящему эффективно для поисковых систем, которые используют кластеризацию с целью получения более релевантных результатов. С помощью выделения основ слов можно найти больше совпадений, поскольку слово имеет более широкий смысл, и даже обрабатывать такие вещи, как орфографические ошибки. А при использовании приложения ИИ эта процедура помогает улучшать общее понимание.

Существует целый ряд алгоритмов для выделения основ слов, многие из которых являются довольно простыми. Но они дают неоднозначные результаты. Вот данные компании IBM:

«Алгоритм Портера, например, будет утверждать, что слово universal (универсальный) имеет ту же основу, что и слова university (университет) и universities (университеты), и это наблюдение может иметь историческую основу, но больше не является семантически релевантным. Стеммер Портера также не признает, что слова theater и theatre (театр) должны принадлежать к одному и тому же классу основ. По причинам, подобным этим, поисковый механизм Watson Explorer Engine не использует стеммер Портера в качестве своего английского стеммера».

На самом деле компания IBM создала собственный фирменный стеммер, и он позволяет выполнять глубокую настройку под собственные нужды.

Лемматизация

Процедура лемматизации похожа на процедуру выделения основы слова. Но вместо удаления аффиксов или префиксов в центре внимания находится отыскание

похожих корневых (канонических) слов. Примером является слово better (лучше), которое мы могли бы лемматизировать как good (хорошо). Это работает до тех пор, пока смысл остается в основном тем же самым. В нашем примере и то и другое являются примерно одинаковыми, но слово good имеет более ясный смысл. Лемматизация также может работать с обеспечением более качественных операций поиска или понимания языка, в особенности в машинном переводе с языка на язык.

Для того чтобы использовать лемматизацию эффективно, система обработки ЕЯ должна понимать смыслы слов и контекст. Другими словами, эта процедура обычно имеет более высокую результативность, чем процедура выделения основ слов. С другой стороны, это также означает, что ее алгоритмы являются более сложными и для нее требуются более высокие уровни вычислительной мощности.

Шаг 2. Понимание и генерирование языка

После того как текст был помещен в формат, с которым компьютеры способны работать, система обработки ЕЯ должна понять общий смысл. В большинстве случаев эта часть является самой трудной.

Но на протяжении многих лет исследователи разработали массу технических приемов, оказывающих в этом помощь.

– Частеречная разметка (POS tagging, Parts of Speech Tagging) – этот подход сканирует весь текст и относит каждое слово к его правильной грамматической форме, в частности существительным, глаголам, наречиям и т. д. Думайте о ней как об автоматизированной версии урока родного языка в начальной школе. Более того, некоторые системы частеречной разметки имеют вариации. Обратите внимание, что у существительного есть формы существительного в единственном числе (NN), имена собственные в единственном числе (NNP) и формы существительного во множественном числе (NNS).

– Группирование. Затем слова будут проанализированы в терминах словосочетаний. Например, группа существительного (NP) – это существительное, которое действует как подлежащее (субъект) или дополнение (объект глагола).

– Распознавание именованных сущностей – это выявление слов, которые обозначают географические места, людей и организации.

– Тематическое моделирование – поиск скрытых регулярностей и кластеров в тексте. Один из алгоритмов, именуемый латентным размещением Дирихле (Latent Dirichlet Allocation, LDA), опирается на подходы, основывающиеся на неконтролируемом самообучении, т. е. назначаются случайные темы, а затем компьютер в цикле отыскивает совпадения.

Для многих упомянутых выше процессов мы можем использовать модели глубокого обучения. Они могут быть расширены на большее число областей анализа — с целью обеспечения бесшовного понимания и генерации языка. Этот процесс называется дистрибутивной семантикой.

С помощью сверточной нейронной сети (CNN-сети) можно отыскать кластеры слов, которые транслируются в признаковое отображение. Это сделало возможными такие применения, как машинный перевод, распознавание речи, сентиментный анализ и вопросно-ответные системы.

Распознавание голоса

В 1952 году американская корпорация Bell Labs создала первую в мире систему распознавания голоса под названием Audrey (Automatic Digit Recognition – автоматическое распознавание цифр). Она была способна распознавать фонемы, т. е. самые элементарные звуковые единицы в языке. В английском, например, их всего 44.

Система Audrey могла распознавать звук цифры, от нуля до девяти. Она была точной для голоса создателя машины, ХК Дэвиса (НК Davis), примерно в 90% случаев. И для всех остальных ее точность составляла от 70 до 80% или около того.

Система Audrey была большим достижением, в особенности в свете ограниченных вычислительных мощностей и памяти, имевшихся в то время. Но данная программа также высветила первостепенные трудности с распознаванием голоса. Когда мы говорим, наши предложения могут быть сложными и несколько запутанными. Мы также обычно говорим быстро – в среднем 150 слов в минуту.

В результате системы распознавания голоса совершенствовались чрезвычайно медленными темпами. В 1962 году система IBM Shoebox могла распознавать только 16 слов, 10 цифр и 6 математических команд.

Значительный прогресс в этой технологии произошел лишь в 1980-х годах. Ключевым инновационным прорывом стало использование скрытой марковской модели (hidden Markov model, HMM), основанной на изощренной статистике. Например, если произносится слово dog (собака), то будет проведен анализ отдельных звуков d, o и g. Алгоритм скрытой марковской модели назначит каждому из них оценку. Со временем система научится лучше понимать звуки и переводить их в слова.

В то время как скрытая марковская модель стала очень важным инновационным прорывом, она по-прежнему не могла эффективно обрабатывать непрерывную речь. Например, голосовые системы были основаны на сопоставлении с шаблонами. Это включало в себя транслирование звуковых волн в числа, что делалось путем выборки. Программа измеряла частоту интервалов и сохраняла результаты. Но для успешной работы нужно, чтобы имелось очень близкое совпадение. Из-за этого голосовой ввод должен был быть достаточно четким и медленным. Также допускался лишь незначительный фоновый шум.

Но к 1990-м годам разработчики программно-информационного обеспечения добились больших успехов и создали коммерческие системы, такие как Dragon Dictate, которые могли понимать тысячи слов в непрерывной речи. Однако принятие их на вооружение по-прежнему не стало магистральным. Многие люди по-прежнему находили, что проще вводить данные в компьютеры вручную и использовать мышь. Тем не менее в некоторых профессиональных областях деятельности, таких как медицина (популярный случай использования касался расшифровки диагноза пациентов), распознавание речи находило высокий уровень использования.

С появлением машинного обучения и глубокого обучения голосовые системы быстро стали намного изощреннее и точнее. Несколько самых ключевых алгоритмов включают использование долгой краткосрочной памяти (long short-

term memory, LSTM), рекуррентных нейронных сетей и глубоких нейронных сетей прямого распространения. Компания Google продолжила имплементировать эти подходы в бесплатной веб-службе Google Voice, которая была доступна для сотен миллионов пользователей смартфонов. И конечно же, мы стали свидетелями большого прогресса в других предложениях, таких как Siri, Alexa и Cortana.

Виртуальные помощники

В 2003 году Министерство обороны США рассчитывало инвестировать в технологии нового поколения, предназначенные для боевых действий. Одной из ключевых инициатив было строительство изощренного виртуального помощника, который мог бы распознавать устные команды. Министерство обороны выделило на это 150 млн. долларов и поручило лаборатории Стэнфордского научно-исследовательского институа, SRI Lab (Stanford Research Institute), расположенной в Кремниевой долине, разработать приложение. Несмотря на то что указанная лаборатория была некоммерческой, ей все же разрешалось лицензировать свои технологии (например, струйный принтер) для стартапов.

И вот что произошло с виртуальным помощником. Несколько членов лаборатории Стэнфордского научно-исследовательского института – Даг Киттлаус (Dag Kittlaus), Том Грубер (Тош Gruber) и Адам Чейер (Adam Cheyer) – назвали его Siri и организовали собственную компанию, чтобы заработать на этой возможности капитал. Они приступили к своей деятельности в 2007 году – именно тогда, когда был анонсирован и выпущен iPhone от компании Apple.

Однако, чтобы довести продукт до уровня, на котором он может быть полезен потребителям, понадобилось гораздо больше научных исследований и разработок. Основателям пришлось создать систему обработки реально-временных данных, построить поисковую систему для географической информации, а также обеспечить безопасность кредитных карт и персональных данных. Но именно обработка ЕЯ стала самым трудным испытанием.

В одном из интервью Чейер отметил:

«Самая трудная техническая задача с Siri была связана с массивными объемами двусмысленности, присутствующей в человеческом языке. Возьмем фразу «book 4-star restaurant in Boston» (забронировать 4-звездочный ресторан в Бостоне) — на первый взгляд ее очень просто понять. Наша прототипная система могла бы справиться с этим легко. Однако, когда мы загрузили в систему в качестве словаря десятки миллионов названий компаний и сотни тысяч городов (в английском языке почти каждое слово является названием компании), число кандидатных интерпретаций подскочило до небес».

Но команде удалось решить эти проблемы и превратить Siri в мощную систему, которая была запущена в Apple App Store в феврале 2010 года. «Это самое сложное распознавание голоса, которое может появиться в смартфоне», — говорится в обзоре ежемесячного американского журнала Wired.com.

Стив Джобс (Steve Jobs) обратил на это внимание и позвонил основателям. Они договорились встретиться через несколько дней, и переговоры быстро привели

к приобретению, которое произошло в конце апреля, на сумму более 200 млн долларов.

Однако Джобс считал, что помощник Siri нуждается в усовершенствовании. По этой причине в 2011 году был выпущен его повторный релиз. На самом деле это произошло за день до смерти Джобса.

На сегодняшний день Siri занимает самую большую позицию на рынке виртуальных помощников – 48,6%. Доля ассистента Google составляет 28,7%, а доля Alexa от Amazon.com – 13,2%.

По данным «Отчета о переходе потребителей на голосовые помощники», около 146,6 млн. человек в Соединенных Штатах попробовали виртуальные помощники на своих смартфонах и более 50 млн с помощью умных звуковых динамиков. Но это только часть истории. Голосовые технологии также внедряются в носимые устройства, наушники и бытовые приборы.

Вот еще несколько интересных находок:

– использование голоса для поиска товаров обогнало в рейтинге поиск различных вариантов развлечений;

– в том, что касается продуктивности, то наиболее распространенными вариантами использования голоса являются телефонные звонки, отправка электронной почты и установка будильников;

– наиболее часто голосовое приложение в смартфонах используется, когда человек находится за рулем;

– если говорить о жалобах на голосовые помощники в смартфонах, то самый высокий процент был связан с непоследовательностью в понимании запросов. Опять же, это указывает на текущие проблемы технологии обработки ЕЯ.

Перспективы роста у виртуальных помощников остаются яркими, и эта категория, вероятно, будет ключевой для индустрии ИИ. Аналитическая компания Juniper Research прогнозирует, что к 2023 году число виртуальных помощников, используемых на глобальной основе, вырастет более чем в три раза до 2,5 млрд. Ожидается, что самой быстрорастущей категорией на самом деле будут умные телевизоры.

Разговорные роботы

Нередко возникает путаница по поводу разницы между виртуальными помощниками и разговорными роботами (чат-ботами).

Между ними существует много совпадений. Оба используют технологию обработки ЕЯ для интерпретации языка и выполнения задач.

Но есть еще критически важные различия. По большей части разговорные роботы ориентированы в первую очередь на бизнес, например, на поддержку клиентов или сбытовые функции.

Виртуальные помощники, с другой стороны, предназначены практически для каждого, в помощь в их повседневной деятельности.

Происхождение разговорных роботов восходит к 1960-м годам с разработкой программы ELIZA. Но только в последнее десятилетие или около того эта технология стала пригодной для широкого применения.

Перед развертыванием разговорного робота необходимо учесть несколько факторов.

– Задать ожидания. Не переусердствовать со способностями разговорных роботов. Например, мы не должны притворяться, что разговорный робот является человеком. Это верная дорога к тому, чтобы испытать негативные переживания. В результате мы, возможно, захотим начать диалог с разговорным роботом выражением «Привет, я – разговорный робот и готов помочь вам с...».

– Автоматизация. В некоторых случаях разговорный робот может регулировать весь процесс с клиентом. Но в этом цикле все равно должны быть люди. По словам Антонио Канджиано (Antonio Cangiano), пропагандиста ИИ в компании IBM, «цель разговорных роботов состоит не в том, чтобы заменить людей полностью, а в том, чтобы быть, так сказать, первой линией обороны. Это может означать не только экономию денег компаний, но и высвобождение человеческих агентов, которые смогут тратить больше времени на расследование сложных запросов, которые будут им передаваться».

– Трение. Насколько это возможно, постарайтесь найти способы для того, чтобы разговорный робот решал задачи как можно быстрее. И это не обязательно будет сводиться к использованию диалога. Вместо этого более подходящей альтернативой может быть предоставление простой формы для заполнения, а также запланированная демонстрация.

– Повторяющиеся процессы. Они часто подходят для разговорных роботов идеальным образом. Примеры включают аутентификацию, статус заказа, планирование и простые запросы на изменение.

– Централизация. Следует обеспечивать интеграцию данных в разговорные роботы. Это позволит получить более плавно меняющийся опыт. Несомненно, клиенты быстро раздражаются, если им приходится повторять информацию.

– Персонализация опыта. Она достается нелегко, но может принести значительные выгоды. Джонатан Тейлор (Jonathan Taylor), технический директор компании-разработчика разговорных платформ Zoovu, приводит такой пример: Покупка объектива камеры будет разной для каждого покупателя. Существует много разновидностей линз, в которых, возможно, разбирается хотя бы чуть-чуть информированный покупатель, но средний потребитель, как правило, информирован не особо.

– Анализ данных. Очень важно с помощью разговорного робота отслеживать обратную связь. Какой уровень покупательской удовлетворенности? Какова степень точности?

– Разговорный дизайн и впечатление пользователя (user experience, UX – пользовательский опыт). Он совершенно отличается от создания веб-сайта или даже мобильного приложения. С помощью чат-бота мы должны думать о личности пользователя, о том, какой у него пол, и даже о культурном контексте.

Даже с учетом всех проблем, связанных с разговорными роботами, эта технология продолжает совершенствоваться. Что еще важнее, эти системы, вероятно, будут становиться все более важной частью индустрии ИИ. По данным компании 1DC, в 2019 году на разговорных роботов будет потрачено около 4,5

млрд. долларов, что соотносится с общей суммой 35,8 млрд. долларов, оцениваемой для систем ИИ.

Ключевые моменты

Обработка естественного языка (ЕЯ) – это использование ИИ для того, чтобы компьютеры могли понимать людей.

Разговорный робот, или чат-бот, – это система ИИ, которая общается с людьми, в частности, голосом или в онлайновой беседе.

Хотя в технологии обработки ЕЯ были достигнуты ощутимые успехи, еще предстоит проделать большую работу. Перечислим лишь некоторые трудности: неоднозначность языка, невербальные сигналы, разные диалекты и акценты, а также изменения в языке.

Два основных шага в обработке ЕЯ включают очистку/предобработку текста и использование ИИ для понимания и генерации языка.

Лексемизация, или токенизация, – это когда текст разбирается и сегментируется на разные части.

Во время нормализации текст конвертируется в форму, которая облегчает его анализ, например, путем удаления знаков препинания или сокращений.

Выделение основ слов, или стемминг, описывает процесс сокращения слова до его основы / корня (или леммы), например, путем удаления аффиксов и суффиксов.

Аналогично выделению основ слов, лемматизация предусматривает отыскание похожих корневых (канонических) слов.

Для подобласти под названием «понимание языка» в технологии обработки ЕЯ существует целый ряд подходов, таких как частеречная разметка (перевод текста в грамматическую форму), группирование (обработка текста в виде словосочетаний) и тематическое моделирование (отыскание скрытых регулярностей и кластеров).

Фонема – это элементарная звуковая единица в языке.

Вывод

Для многих людей первое взаимодействие с технологией обработки ЕЯ происходит с виртуальными помощниками. Даже несмотря на то, что данная технология далека от совершенства, она все равно оказывается весьма полезной – в особенности для ответов на вопросы или получения информации, скажем, о близлежащем ресторане.

Но технология обработки ЕЯ также оказывает большое влияние на деловой мир. В предстоящие годы эта технология будет приобретать все большее значение для электронной коммерции и обслуживания клиентов, обеспечивая значительную экономию средств и позволяя сотрудникам сосредотачиваться на более эффективных видах деятельности.

Правда, до этого еще далеко из-за сложностей, присущих естественному языку. Но прогресс продолжает идти быстрыми шагами, в особенности за счет технологий ИИ следующего поколения, таких как глубокое обучение.

Лекция 7. Физические роботы. Итоговое воплощение искусственного интеллекта

Что такое робот?

Происхождение слова «робот» восходит к 1921 году и пьесе Карела Чапека «Универсальные роботы Россума». Речь идет о фабрике, которая создавала роботов из органического вещества, и, разумеется, они были враждебными! В конце концов они объединятся и восстанут против своих хозяев-людей (для справки, слово «робот» происходит от чешского слова robota, означающего принудительный труд).

Но каким будет сегодня хорошее определение этого типа систем? Имейте в виду, что существует много вариаций таких систем, поскольку роботы могут иметь массу форм и функций.

Однако мы можем свести их к нескольким ключевым моментам.

– Физический. Размер робота может варьироваться: от крошечных машин, которые могут исследовать наше тело, до массивных промышленных систем, летательных аппаратов и подводных судов. Кроме того, должен иметься какой-то источник энергии, такой как батарея, электричество или солнечная энергия.

– Действие. Робот просто-напросто должен уметь совершать те или иные действия, к которым относятся перемещение предмета или даже беседа.

– Ощущение. Для того чтобы действовать, робот должен понимать окружающую его среду. Это возможно с помощью датчиков и систем обратной связи.

– Интеллект. Это не означает полный набор способностей ИИ. И все же робот должен быть запрограммирован выполнять действия.

В наше время не так уж сложно создать робота с нуля. Например, интернет-магазин RobotShop.com имеет сотни комплектов, которые варьируются от 10 до 35 750 долларов.

Трогательная история изобретательности в строительстве роботов касается двухлетнего Киллиана Джексона (Gillian Jackson). Он родился с редким генетическим заболеванием, из-за которого оказался обездвиженным. Его родители попытались получить компенсацию за специальную электрическую инвалидную коляску, но им было отказано.

Так вот, студенты Фармингтонской средней школы приняли меры и построили систему для Киллиана. По сути, это была инвалидная коляска-робот, и на ее завершение ушел всего месяц. Благодаря ей Киллиан теперь может гоняться за своими двумя корги по всему дому!

В то время как выше мы рассмотрели характерные особенности роботов, следует также рассмотреть их ключевые взаимодействия.

– Датчики. Типичный датчик – это фотокамера или лидар (детекция освещенности и дальности), который использует лазерный сканер для создания 3D-изображений. Но роботы могут также иметь системы для звука, осязания; вкуса и даже запаха. На самом деле, они могут также содержать датчики, которые выходят

за пределы человеческих возможностей, например, датчики ночного видения или обнаружения химических веществ. Информация с датчиков передается в контроллер, который может активировать руку или другие части робота.

– Актуаторы — это электромеханические устройства, такие как двигатели. По большей части они помогают с движением рук, ног, головы и любой другой подвижной части.

– Компьютер. Он имеет память и процессоры, с помощью которых обрабатываются входы от датчиков. В продвинутых роботах также могут иметься чипы ИИ или интернет-соединения с облачными платформами ИИ.

Существуют также два главных способа управления роботом. Прежде всего, это дистанционное управление, осуществляемое человеком. В этом случае робот называется телеуправляемым, или телероботом. Затем идет автономный робот, который для навигации использует собственные способности, в частности, с помощью ИИ.

Как назывался первый мобильный думающий робот? Этот робот назывался Shakey. Его имя (переводится как «трясучка») было неслучайным. Как отметил руководитель проекта по строительству этой системы Чарльз Розен (Charles Rosen), «мы работали в течение месяца, пытаясь подобрать к нему хорошее имя, начиная от греческих имен и заканчивая всякой всячиной, а затем один из нас сказал: «Эй, да он трясется, как черт, и движется повсюду, давайте просто назовем его Трясучкой».

Стэнфордский научно-исследовательский институт (Stanford Research Institute, SRI), финансируемый агентством DARPA, работал над Shakey с 1966 по 1972 год. И эта работа была по тем временам довольно изощренной. Shakey был большим, более пяти футов ростом, и имел колеса для перемещения, датчики и камеры, помогавшие выполнять прикосновения. Он также был подключен по беспроводной сети к компьютерам DEC PDP-10 и PDP-15. Оттуда человек мог вводить команды по телетайпу. И тем не менее Shakey использовал алгоритмы для навигации по своей окружающей среде, и даже закрывал двери.

Разработка этого робота стала результатом бесчисленных инновационных прорывов в области ИИ. Например, Нильс Нильссон (Nils Nilsson) и Ричард Файке (Richard Fikes) создали Stripes (Stanford Research Institute Problem Solver – решатель Стэнфордского научно-исследовательского института), который позволял автоматизировать планирование, а также алгоритм A* для отыскания кратчайшего пути с привлечением наименьшего объема компьютерных ресурсов.

К концу 1960-х годов, когда Америка была сосредоточена на космической программе, вокруг Shakey было довольно много шума. Лестная статья в журнале Life объявила, что робот был «первым электронным человеком».

Но, к сожалению, в 1972 году, когда наступила зима ИИ, агентство DARPA прекратило финансирование работ над Shakey. Тем не менее этот робот по-прежнему оставался ключевой частью истории технологий и в 2004 году был введен в Зал славы роботов.

Промышленные и коммерческие роботы

Первое реальное применение роботов было связано с промышленностью. Но на принятие этих систем на вооружение на самом деле ушло довольно много времени.

История начинается с Джорджа Девола (George Devol), изобретателя, который не закончил среднюю школу. Но это не являлось проблемой. У Девола был талант к инженерному делу и творчеству, поскольку он создал несколько стержневых систем для микроволновых печей, штрих-кодов и автоматических дверей (за свою жизнь он получил более 40 патентов).

В начале 1950-х годов он также получил патент на программируемого робота под названием Unimate. Он изо всех сил пытался заинтересовать своей идеей инвесторов, но все ее отклоняли. Однако в 1957 году его жизнь навсегда изменилась, когда на одной коктейльной вечеринке он познакомился с Джозефом Энгельбергером (Joseph Engelberger). Это знакомство было аналогичным тому, которое произошло между Стивом Джобсом (Steve Jobs) и Стивом Возняком (Steve Wozniak), давшим начало созданию компьютера Apple.

Энгельбергер был не только инженером, но и опытным бизнесменом. Он даже любил читать научную фантастику, например, рассказы Айзека Азимова. Поэтому Энгельбергер хотел, чтобы робот Unimate принес обществу пользу.

Тем не менее сопротивление по-прежнему сохранялось — поскольку многие люди считали эту идею нереалистичной и, скажем так, научно-фантастичной, — и потребовался год, чтобы получить финансирование. Но, как только Энгельбергер этого добился, он потратил на строительство робота совсем немного времени и в 1961 году смог продать его компании General Motors (GM). Робот Unimate был громоздким (весом 2700 фунтов) и имел одну 7-футовую руку, но все равно был довольно полезен, а само его существование означало, что людям не придется заниматься деятельностью, опасной по своей природе. Среди некоторых его главных функций была сварка, распыление и захват, и все это делалось точно и в режиме 24/7.

Энгельбергер изыскивал креативные способы пропагандировать своего робота. С этой целью в 1966 году он выступил на шоу Джонни Карсона «The Tonight Show», в котором робот Unimate отлично поставил мяч для гольфа и даже налил пива. Джонни пошутил, что машина может «заменить чью-то работу».

Но в эксплуатации промышленных роботов были свои трудности. Интересно отметить, что компания General Motors узнала об этом на собственном горьком опыте в 1980-х годах. В то время генеральный директор Роджер Смит (Roger Smith) продвигал идею завода «с выключенным освещением», т. е. такой производственный цикл, при котором роботы могли бы собирать автомобили в темноте!

Он потратил на эту программу колоссальные 90 млрд. долларов и даже создал совместное с компанией Fujitsu-Fanuc предприятие под названием GMF Robotics. Эта организация через некоторое время станет крупнейшим в мире производителем роботов.

Но, к сожалению, эта затея обернулась катастрофой. Помимо того, что они усугубили отношения с профсоюзами, роботы часто не оправдывали ожиданий. В перечне фиаско содержались случаи, когда роботы заваривали двери автомобилей или красили самих себя, а не автомобили!

Тем не менее ситуация с GMF Robotics не является чем-то действительно новым – и это не обязательно вина заблуждающихся старших менеджеров. Взгляните на компанию Tesla, одну из самых инновационных в мире. И тем не менее генеральный директор Илон Маск по-прежнему испытывал серьезные проблемы с роботами на своих заводах. Эти проблемы стали настолько серьезными, что существование компании Tesla оказалось под угрозой.

В интервью телепередаче «This Morning» на канале CBS в апреле 2018 года Маск отметил, что он использовал слишком много роботов при производстве электрического пятиместного седана Model 3, и это фактически замедлило процесс. Он отметил, что ему следовало бы привлечь больше людей.

Все это указывает на то, что однажды написал Ганс Моравек (Hans Moravec): «Сравнительно легко заставить компьютеры демонстрировать результаты взрослого уровня в тестах на интеллект или игре в шашки, и трудно или невозможно придать им навыки годовалого ребенка, когда дело доходит до восприятия и мобильности». Это высказывание часто называют парадоксом Моравека.

Несмотря на все это, промышленные роботы стали массовой индустрией, расширяющейся в различных сегментах, таких как потребительские товары, биотехнологии / здравоохранение и пластмассы. Согласно данным Ассоциации робототехнических отраслей (Robotic Industries Association, RIA), по состоянию на 2018 год в Северной Америке было отгружено 35 880 промышленных и коммерческих роботов. Например, на долю автомобильной промышленности их приходилось около 53%, но этот показатель снижается.

Хорошо, тогда как насчет ИИ и роботов? Где находится этот статус технологии? Даже с учетом инновационных прорывов в области глубокого обучения обычно наблюдается медленный прогресс в использовании ИИ с роботами. Отчасти это связано с тем, что большинство исследований было сосредоточено на программно-информационных моделях, таких как распознавание образов. Однако еще одна причина заключается в том, что физические роботы требуют сложных технологий для понимания окружающей среды – которая часто является зашумленной и отвлекает – в реальном времени. Сюда входит возможность совместных локализации и ориентирования на местности (simultaneous localization and mapping, SLAM) в неизвестных средах при одновременном отслеживании местоположения робота. Для того чтобы делать это эффективно, возможно, даже потребуется изобрести новые технологии, такие как более совершенные нейросетевые алгоритмы и квантовые компьютеры.

Несмотря на все это, прогресс, безусловно, есть, и в особенности с использованием технических решений на основе подкрепляемого самообучения. Рассмотрим несколько таких нововведений.

Osaro. Данная компания разрабатывает системы, которые позволяют роботам быстро учиться. Она описывает это как «способность имитировать поведение,

которое требует усваиваемого обобщения данных датчиков, а также высокоуровневого планирования и манипулирования объектами. Это также обеспечит возможность обучаться от одной машины к другой и совершенствоваться за пределами понимания программиста-человека». Например, один из ее роботов смог всего за пять секунд научиться поднимать и укладывать курицу (предполагается, что эта система будет использоваться на птицефабриках). Но эта технология может иметь много применений, например, в дронах, автономных транспортных средствах и Интернете вещей (Internet of Things, IoT).

OpenAI. Эта компания создала роботизированную руку Dactyl, обладающую человекоподобной ловкостью. Она основана не на взаимодействиях в реальном мире, а на изощренном тренировочном процессе с использованием симуляций. Компания OpenAI называет его «доменной рандомизацией», которая предоставляет роботу большое число сценариев – даже с очень низкой вероятностью произойти. Благодаря роботу Dactyl удалось задействовать симуляции решений задач продолжительностью около 100 лет. Один из удивительных результатов состоял в том, что эта система усвоила действия человеческой руки, которые не были запрограммированы заранее, например, скольжение пальца. Робот Dactyl также был натренирован обращаться с несовершенной информацией, скажем, когда датчики задерживают показания или, когда возникает необходимость обрабатывать несколько объектов сразу.

MIT. Робот может легко брать тысячи образцов данных в целях понимания окружающей его среды, например, для того, чтобы обнаруживать нечто такое же простое, как кружка. Но согласно исследовательской работе профессоров МТИ (MIT), возможно, есть способ сократить эти данные. Они использовали нейронную сеть, которая сосредотачивалась только на нескольких ключевых признаках. Данное исследование все еще находится на ранней стадии, но оно может оказаться очень эффективным для роботов.

Google. Начиная с 2013 года, компания продолжила процесс слияния и поглощения робототехнических компаний. Но результаты оказались неутешительными. Несмотря на это, она не отказалась от своего бизнеса в этой сфере. За последние несколько лет компания Google сосредоточилась на более простых роботах, которые управляются искусственным интеллектом, и компания создала новое подразделение, в Google называемое Robotics. Например, один из роботов может посмотреть на ящик с предметами и идентифицировать тот, который запрашивается – поднимая его трехпалой рукой, – примерно в 85% случаев. Типичный человек способен сделать это примерно в 80% случаев.

Значит ли это, что все указывает на полную автоматизацию? По всей видимости, нет – по крайней мере, в обозримом будущем.

Имейте в виду, что главенствующим трендом является развитие коботов, т. е. сотрудничающих роботов, которые работают вместе с людьми. В общем он превращается в гораздо более мощный подход, поскольку можно задействовать преимущества как машин, так и людей.

Отметим, что одним из главных лидеров в этой категории является компания Amazon.com. Еще в 2012 году указанная компания выделила 775 млн долларов для Kiva, ведущего производителя промышленных роботов. С тех пор компания

Amazon.com развернула около 100 тыс. систем в более чем 25 центрах исполнения заказов (благодаря этому компания наблюдала 40%-е улучшение в складских мощностях). Вот как компания описывает ее:

«Подразделение Amazon Robotics автоматизирует работу центров исполнения заказов с использованием различных методов робототехники, включая автономные мобильные роботы, сложные управляющие программы, восприятие языка, управление питанием, компьютерное зрение, глубинное зондирование, машинное обучение, распознавание объектов и семантическое понимание команд».

Внутри складов роботы быстро перемещаются по полу, помогая локализовывать и поднимать складские поддоны. Но участие людей в рабочем процессе также имеет критическое значение, поскольку они лучше умеют выявлять и выбирать отдельные продукты.

Тем не менее организация работы имеет свои сложности. Например, сотрудники склада носят специальные предохранительные жилеты на основе роботизированной технологии, чтобы их не сбили роботы! Эта технология позволяет роботу идентифицировать человека.

Но с коботами есть и другие проблемы. Например, существует реальный страх, что работники в конечном счете будут заменены машинами. Более того, вполне естественно, что люди чувствуют себя пресловутым винтиком в колесе, что приводит к снижению морального духа. Могут ли люди действительно общаться с роботами? Вероятно, нет, в особенности с промышленными роботами, которые вообще не обладают человеческими качествами.

Роботы в реальном мире

Итак, давайте теперь рассмотрим некоторые другие интересные примеры использования промышленных и коммерческих роботов.

Пример использования: обеспечение безопасности

Хотя использование традиционных технологий обеспечения безопасности достаточно эффективно – скажем, с использованием камер и датчиков, — они являются статичными и не обязательно хорошо подходят для реально-временного реагирования. Но с роботом можно быть гораздо более проактивным из-за мобильности и лежащего в основе интеллекта.

Тем не менее люди по-прежнему участвуют в рабочем процессе. Роботы могут делать то, в чем они хороши, например, 24 часа в сутки обрабатывать данные и зондировать, а люди могут сосредоточиться на критическом мышлении и взвешивании альтернатив.

Пример использования: онлайновая аптека

Будучи фармацевтом во втором поколении, Ти Джей Паркер (TJ Parker) неоднократно наблюдал разочарованных покупателей, которые не могли разобраться со своими рецептами. Поэтому он задался вопросом: существует ли решение по созданию цифровой аптеки?

Он был убежден, что ответом будет «да». Но хотя у него был основательный опыт работы в этой индустрии, он нуждался в солидном технологическом соучредителе, которого он нашел в лице Эллиота Коэна (Elliot Cohen), инженера МТИ. Они продолжат сотрудничество, создав в 2013 году компанию PillPack.

В центре внимания было переосмысление клиентского опыта. Используя приложение или перейдя на веб-сайт PillPack, пользователь мог бы легко зарегистрироваться – например, ввести информацию о страховании, ввести потребности в лекарственных препаратах и запланировать доставку. При получении пакета пользователь мог бы иметь подробную информацию о дозировке и даже изображения каждой таблетки. Кроме того, каждая таблетка снабжалась бы этикетками и предварительно была упакована в контейнеры.

Для того чтобы все это стало реальностью, требовалась сложная технологическая инфраструктура под названием PharmacyOS. Эта инфраструктура также опиралась на сеть роботов, которые были расположены на складе площадью 80 000 кв. футов. Благодаря этому система могла эффективно сортировать и упаковывать рецепты. Но на предприятии также были лицензированные фармацевты, которые управляли процессом и следили за тем, чтобы все было в соответствии с нормативами.

В июне 2018 года компания Amazon.com выложила около 1 млрд. долларов за компанию PillPack. В новостях было сообщено, что стоимость акций таких компаний, как CVS и Walgreens, упала ввиду опасений в том, что гигант электронной коммерции готовится играть по-крупному на рынке здравоохранения.

Пример использования: роботы-ученые

Разработка лекарственных препаратов обходится чрезвычайно дорого. Основываясь на исследованиях Центра университета Тафтса (Tufts Center) по изучению разработки лекарств, средний показатель составляет около 2,6 млрд. долларов за официально одобренное химическое соединение. Кроме того, из-за обременительных правил на вывод нового препарата на рынок может уходить более десятка лет.

Но использование сложных роботов и глубокое обучение могли бы помочь. Для того чтобы понять, как это можно сделать, посмотрите, что сделали исследователи из университетов Аберистуита и Кембриджа. В 2009 году они запустили робота Adam, по сути являвшегося ученым-роботом, который помогал в процессе изыскания новых лекарств. Затем, несколько лет спустя, они запустили робота Eve, ставшего роботом следующего поколения.

Система может выдвигать гипотезы и проверять их, а также проводить эксперименты. Но этот процесс не сводится только к вычислениям методом грубой силы (система может проводить скрининг более 10 тыс. химических соединений в день). С помощью глубокого обучения робот Eve способен использовать интеллект, чтобы лучше выявлять те соединения, которые обладают наибольшим потенциалом. Например, он смог показать, что триклозан – элемент, часто встречающийся в зубной пасте для предотвращения образования зубного налета, – может быть эффективным против роста паразитов при малярии. Это особенно

важно, поскольку указанная болезнь становится все более устойчивой к существующим методам лечения.

Гуманоидные и потребительские роботы

Популярный мультфильм «Джетсоны» (The Jetsons) вышел в начале 1960-х годов и имел большой состав персонажей. Одним из них была Рози (Rosie), робот-горничная, которая всегда держала в руках пылесос.

Кто не хотел бы иметь чего-то подобного? Возможно, многие. Но не нужно ждать, что в ближайшее время кто-то вроде Рози придет в наши дома. Если уж говорить о потребительских роботах, то мы все еще находимся в начале пути. Другими словами, вместо этого мы видим роботов, которые имеют лишь некоторые человеческие признаки.

Вот примечательные примеры.

Sophia. Разработанный гонконгской компанией Hanson Robotics, этот робот является, пожалуй, самым известным. Более того, в конце 2017 года Саудовская Аравия предоставила Софии гражданство! София, похожая на американскую кинозвезду Одри Хепберн, может ходить и говорить. Но в ее действиях есть и тонкости, такие как поддержание зрительного контакта.

Atlas. Его разработчиком является компания Boston Dynamics, которая выпустила этого робота летом 2013 года. Без сомнения, за эти годы робот Atlas стал намного лучше. Он может, например, выполнять обратные сальто и подниматься с пола при падении.

Pepper. Это гуманоидный робот, созданный компанией Soft-Bank Robotics, который ориентирован на обслуживание клиентов, например, в торговых точках. Указанная машина может использовать жесты – в целях улучшения общения, а также говорить на нескольких языках.

По мере того как гуманоидные технологии становятся все более реалистичными и продвинутыми, в обществе неизбежно происходят изменения. Социальные нормы о любви и дружбе будут эволюционировать. В конце концов, как видно из распространенности смартфонов, мы уже видим, как технология изменяет наше отношение к людям, скажем, посредством текстовых сообщений и участия в социальных сетях. Согласно опросу представителей поколения двухтысячных, проведенному компанией-разработчиком мобильных приложений Tappable, около 10% опрошенных скорее пожертвуют своим мизинцем, чем откажутся от своего смартфона!

Разумеется, это, по всей видимости, произойдет далеко в будущем. Но на данный момент, безусловно, среди социальных роботов имеется несколько интересных инноваций. Один из примеров – робот ElliQ, который состоит из планшета и маленькой головы. По большей части он предназначен для тех, кто живет в одиночестве, например, пожилых людей. Робот ElliQ может говорить, а также оказывать неоценимую помощь, например, напоминать о приеме лекарств. Данная система также позволяет вести видеочаты с членами семьи.

Три закона робототехники

Айзек Азимов, автор работ на различные темы, включая научную фантастику, историю, химию и Шекспира, также оказал большое влияние на роботов. В коротком рассказе («Хоровод»), написанном в 1942 году, он изложил свои три закона робототехники:

1. Робот не может причинить вред человеку или своим бездействием допустить, чтобы человеку был причинен вред.
2. Робот должен подчиняться приказам, отдаваемым ему людьми, за исключением тех случаев, когда такие приказы противоречат первому закону.
3. Робот должен защищать свое существование до тех пор, пока такая защита не противоречит первому или второму закону.

Позже Азимов добавит еще один, нулевой закон, который гласит: «Робот не должен причинять человечеству вред или своим бездействием позволить причинить человечеству вред». Он считал этот закон самым важным.

Азимов напишет еще несколько рассказов, отражающих то, как эти законы будут действовать в сложных ситуациях, и они будут собраны в книге под названием «Я, робот». Все события рассказов происходили в мире XXI века.

Эти три закона представляли собой реакцию Азимова на то, как научная фантастика изображала роботов злонамеренными. Но он считал, что это нереально. Азимов предвидел, что появятся этические правила для контроля над мощью роботов.

На данный момент видение Азимова начинает становиться более реалистичным – другими словами, он подал неплохую идею о том, что необходимо разведать этические принципы. Конечно же, вовсе не означает, что его подход является правильным. Но он дает хороший старт, тем более что благодаря мощи ИИ роботы становятся умнее и личностнее.

Кибербезопасность и роботы

Кибербезопасность не была для роботов большой проблемой. Но, к сожалению, это, по всей видимости, продлится недолго. Главная причина заключается в том, что роботы все чаще подключаются к облаку. То же самое касается и других систем, таких как Интернет вещей (IoT) и автономные автомобили. Например, многие из этих систем обновляются по беспроводной сети, что подвергает их воздействию вредоносных программ, вирусов и даже вымогательствам. Более того, когда дело касается электромобилей, то существует также уязвимость к атакам со стороны зарядной сети.

В действительности, данные могут задерживаться внутри автомобиля! И поэтому, если автомобиль поврежден или мы его продаем, то информация – например, видео, навигационные данные и контакты из парных подключений к смартфонам – может стать доступной другим людям. Согласно новостному и деловому веб-сайту CNBC.com, белый хакер под ником GreenTheOnly смог извлечь эти данные из различных моделей Tesla на свалках. Но важно отметить, что компания Tesla все-таки предоставляет опции по очистке данных, и можно

отказаться от сбора данных (но это означает, что не будет некоторых преимуществ, таких как обновления по воздуху (over-the-air, OTA).

Кроме того, если нарушена кибербезопасность робота, то последствия, безусловно, могут быть разрушительными. Только представьте себе, что хакер проник в производственную линию, в снабженческую цепочку или даже в роботизированную хирургическую систему. Жизни людей могут оказаться в опасности.

Несмотря на это, в обеспечение кибербезопасности роботов вложено не так уж много средств. К настоящему времени появилось лишь несколько компаний, таких как Karamba Security и Cybereason, которые сосредоточены на этой области. Но по мере того как проблемы будут усугубляться, неизбежно будет расти объем инвестиций со стороны венчурных капиталовложений и новых инициатив от унаследованных фирм в области обеспечения кибербезопасности.

Будущее роботов

Родни Брукс (Rodney Brooks) – это один из гигантов индустрии робототехники. В 1990 году он стал соучредителем компании iRobot с целью отыскать способы коммерциализации данной технологии. Однако это оказалось нелегко. Только в 2002 году компания запустила свой робот-пылесос Roomba, который стал большим хитом среди потребителей.

Но стартап iRobot не был единственным для Брукса. Он также посодействовал в запуске компании Rethink Robotics – и его видение было амбициозным. Вот как он выразился в 2010 году, когда его компания объявила о финансировании в размере 20 млн долларов:

«Наши роботы будут интуитивно понятными в использовании, интеллектуальными и очень гибкими. Их легко будет приобрести, натренировать и развернуть, и они будут невероятно недорогими. Компания [Rethink Robotics] изменит определение того, как и где роботы могут использоваться, резко расширив рынок роботов».

Но, к сожалению, как и с iRobot, возникло много трудностей. Несмотря на то что идея Брукса в отношении коботов была новаторской – и в конечном счете окажется прибыльным рынком, – ему пришлось бороться с трудностями построения эффективной системы. Акцент на обеспечение безопасности означал, что их прецизионность и точность не соответствовали стандартам промышленных потребителей. Из-за этого спрос на роботов компании Rethink был умеренным.

К октябрю 2018 года у компании закончились наличные деньги, и ей пришлось закрыться. Компания Rethink привлекла всего около 150 млн. долларов от венчурных и стратегических инвесторов, таких как Goldman Sachs, Sigma Partners, GE и Bezos Expeditions. Интеллектуальная собственность компании была продана немецкой фирме по автоматизации HAHN Group.

Правда, это только один пример. Но опять же, это показывает, что даже самые умные технические специалисты могут ошибаться. И что еще более важно, рынок робототехники имеет уникальные сложности. В том, что касается эволюции этой категории, ее прогресс бывает неустойчивым и волатильным.

Шультц, технический директор компании Cobalt, отметил:

«Несмотря на то что за последнее десятилетие робототехника достигла прогресса, эта индустрия еще не полностью реализовала свой потенциал. Любая новая технология создает волну многочисленных новых компаний, но лишь немногие выживают и превращаются в прочные предприятия. Крах доткомов погубил большинство интернет-компаний, но компании Google, Amazon и Netflix выжили. Робототехнические компании должны быть откровенны в том, что их роботы могут сделать для своих клиентов сегодня, преодолеть голливудские стереотипы о роботах как о плохих парнях и продемонстрировать клиентам четкую возвратность инвестиций (ROI)».

Ключевые моменты

Роботы могут совершать действия, чувствовать окружающую среду и обладать определенным уровнем интеллекта. Они также имеют ключевые устройства, такие как датчики, актуаторы (например, двигатели) и компьютеры.

По способу управления роботы бывают двух основных типов: телеробот (он управляется человеком) и автономный робот (основан на системах ИИ).

Разрабатывать роботов невероятно сложно. Даже у лучших технологов мира, таких как Илон Маск из компании Tesla, были серьезные проблемы с этой технологией. Ключевой причиной является парадокс Моравека. В сущности, то, что легко для людей, часто трудновыполнимо для роботов, и наоборот.

Хотя ИИ и оказывает влияние на роботов, этот процесс был медленным. Одна из причин заключается в том, что больше внимания уделяется технологиям, основанным на программно-информационном обеспечении. Но роботы становятся чрезвычайно сложными, когда дело доходит до перемещения в пространстве и восприятия окружающей их среды.

Коботы – это машины, работающие бок о бок с людьми. Идея заключается в том, что такое взаимодействие позволит использовать преимущества как машин, так и людей.

Стоимость роботов является первостепенной причиной недостаточного их внедрения. Но инновационные компании, такие как Cobalt Robotics, используют новые бизнес-модели, которые приходят на выручку, например, с применением подписок.

Потребительские роботы все еще находятся на начальных стадиях своего развития, в особенности по сравнению с промышленными роботами. Но есть несколько интересных примеров использования, таких как машины, которые могут быть компаньонами для людей.

В 1950-х годах писатель-фантаст Айзек Азимов сформулировал три закона робототехники. По большей части они были сосредоточены на обеспечении того, чтобы машины не причиняли вреда людям или обществу. Несмотря на критику подхода Азимова, этот подход по-прежнему широко поддерживается.

Обеспечение безопасности в роботах, как правило, не представляло проблему. Но это, скорее всего, изменится – и довольно быстро. Ведь все больше роботов

подключается к облаку, которое подвержено проникновению вирусов и вредоносных программ.

Операционная система для роботов (ROS) стала стандартом для индустрии робототехники. Указанное промежуточное программно-информационное обеспечение оказывает помощь в планировании, симуляциях, ориентировании на местности, локализации, восприятии и прототипировании.

Разработка интеллектуальных роботов имеет много трудностей из-за необходимости создания физических систем, хотя и существуют инструменты, которые приходят на выручку, например, допуская изощренные симуляции.

Вывод

Вплоть до последних нескольких лет роботы предназначались в основном для высокотехнологичного производства, например, для автомобилей. Но с ростом использования ИИ и снижением затрат на строительство устройств роботы становятся все более распространенными в различных индустриях промышленности.

Существуют интересные примеры использования роботов, которые выполняют такую работу, как мытье полов или обеспечение безопасности объектов.

Но использование ИИ с робототехникой все еще находится в стадии зарождения. Программировать аппаратные системы далеко не просто, и существует потребность в сложных системах по навигации в окружающей среде. Однако с помощью подходов на основе ИИ, таких как подкрепляемое самообучение, прогресс в этой области был значительно ускорен.

Но когда мы думаем об использовании роботов, важно понимать ограничения. И должна иметься четкая цель. В противном случае развертывание может легко привести к дорогостоящему краху. Даже самые инновационные компании в мире, такие как Google и Tesla, столкнулись с трудностями в работе с роботами.

Лекция 8. Внедрение искусственного интеллекта

Подходы к внедрению искусственного интеллекта

Использование ИИ в компании обычно предусматривает два подхода:

– применение программно-информационного обеспечения разработчика / поставщика;

– создание собственных моделей.

Первый из них является наиболее распространенным и может быть достаточным для большого числа компаний.

Шаги внедрения искусственного интеллекта

Если мы планируем внедрить собственные модели ИИ, какие главные шаги следует рассматривать? Каковы лучшие практические приемы? Так вот, прежде всего, критически важно, чтобы данные были достаточно чистыми и структурированными. Это позволит проводить моделирование.

Вот несколько других шагов, к которым следует обратиться:

– выявить задачу, требующую решения;

– собрать сильную команду;

– выбрать подходящие инструменты и платформы;

– создать модель ИИ;

– развернуть и отслеживать работу модели ИИ.

Давайте взглянем на каждый из них.

Выявить задачу, требующую решения

Но когда речь заходит о процессе внедрения ИИ, обычно слишком много внимания уделяется разным технологиям, которые, безусловно, являются увлекательными и мощными. Однако успех – это гораздо больше, чем просто технология; другими словами, сначала должно иметься ясное экономическое обоснование. Следует ответить на вопрос: где в организации это может принести наибольшую выгоду?

Сформировать команду

Не следует спешить с набором команды. Каждый должен быть сконцентрирован на успехе и понимать важность проекта. Если проект в области ИИ мало что покажет, то перспективы будущих инициатив могут оказаться под угрозой.

Команде понадобится лидер, который, как правило, имеет деловой (управленческий) или эксплуатационный (практический) опыт, а также обладает некоторыми техническими навыками. Такой человек должен уметь экономически обосновать проект в области ИИ, а также донести свое видение до нескольких заинтересованных сторон в компании, таких как ИТ-отдел и высшее руководство.

Выбрать походящие инструменты и платформы

Существует много инструментов, которые оказывают помощь в создании моделей ИИ, и большинство из них имеют открытый исходный код. Даже притом, что неплохо их сперва протестировать, все равно рекомендуется сначала оценить ИТ-среду. Делая это, мы окажемся в более оптимальном положении, оценивая инструменты ИИ.

Рассмотрим несколько самых распространенных языков, платформ и инструментов для ИИ.

Язык Python

Гвидо ван Россум (Guido van Rossum), получивший степень магистра математики и компьютерных наук в Амстердамском университете в 1982 году, продолжал работать в различных исследовательских институтах Европы, таких как Корпорация национальных исследовательских инициатив (Corporation for National Research Initiatives, CNRI). Но именно в конце 1980-х годов он создал собственный компьютерный язык, названный им Python.

Название языка на самом деле пришло из популярного британского комедийного сериала «Monty Python» («Монти Пайтон»). Так что язык был немного нестандартным, но как раз это и делало его таким мощным. И вскоре Python станет стандартом для разработки ИИ.

У языка Python было преимущество роста в академическом сообществе, которое имело доступ к Интернету, что помогало ускорить его распространение. Но это также сделало возможным появление глобальной экосистемы с тысячами разных пакетов и библиотек ИИ. Вот только некоторые из них.

NumPy. Эта библиотека позволяет использовать научно-вычислительные приложения. В ее основе лежит способность создавать сложные массивы объектов с высокой производительностью. Это имеет решающее значение для высококлассной обработки данных в моделях ИИ.

Matplotlib. С помощью этой библиотеки можно строить графики наборов данных. Часто библиотека Matplotlib используется в сочетании с библиотеками NumPy / Pandas (Pandas расшифровывается как Python Data Analysis Library, т. е. библиотека анализа данных на Python). Эта библиотека позволяет относительно легко создавать структуры данных для разработки моделей ИИ.

Вычислительные каркасы искусственного интеллекта

Существует масса вычислительных каркасов ИИ, которые предоставляют сквозные системы для построения моделей, их тренировки и развертывания.

Безусловно, самым популярным из них является TensorFlow, который поддерживается компанией Google. Указанная компания начала разработку этого каркаса в 2011 году посредством своего подразделения Google Brain. Цель состояла в том, чтобы найти способ быстрее создавать нейронные сети и встраивать эту технологию во многие приложения Google.

PyTorch. Разработчиком этой платформы, которая была выпущена в 2016 году, является компания Facebook. Как и для TensorFlow, основным языком программирования этой системы служит Python.

Keras. В отличие от вычислительных каркасов TensorFlow и PyTorch, которые предназначены для опытных экспертов по ИИ, Keras предназначен для начинающих. С помощью небольшого объема исходного кода – на языке Python – можно создавать нейронные сети.

В разработке ИИ помогает еще один широко используемый инструмент: блокнот Jupyter. Это не платформа или инструмент разработки. Как раз наоборот, блокнот Jupyter – это веб-приложение, которое позволяет легко программировать на Python и R в целях создания визуализаций и импорта систем ИИ.

За последние несколько лет также появилась новая категория инструментов ИИ под названием автоматизированного машинного обучения, или autoML (automated machine learning). Эти системы помогают справляться с такими процессами, как подготовка данных и отбор признаков. По большей части их цель состоит в том, чтобы оказывать помощь тем организациям, которые не имеют опытных исследователей данных и инженеров ИИ.

Развернуть и отслеживать работу системы искусственного интеллекта

Даже когда мы построим работающую модель ИИ, останется проделать еще много работы. Вам необходимо отыскать способы развернуть ее и потом отслеживать ее работу.

Это требует организации работы по управлению изменениями, которое всегда является сложным. ИИ отличается от типичного ИТ-внедрения тем, что он предусматривает использование предсказаний и глубинных знаний (инсайтов) для принятия решений. Это означает, что люди должны будут переосмыслить то, как они взаимодействуют с указанной технологией.

Также следует учесть существование вероятности, что конечными пользователями станут люди, которые не обладают технической подготовкой, будь то сотрудники или потребители. Вот почему необходимо очень много работать над тем, чтобы сделать модель ИИ максимально простой. Например, если мы создали систему для онлайнового маркетинга, то можем ограничить опции для пользователя, скажем, только четырьмя или пятью.

Еще одна хорошая стратегия – использовать интерактивные визуализации. Другими словами, можно легко увидеть, как меняются тренды, перенастраивая несколько переменных. Также можно обеспечить возможность щелчком по той или иной части диаграммы получить более подробную информацию.

Также важно создать документацию. Но она должна быть чем-то большим, нежели просто письменные материалы. Например, эффективным подходом является разработка видеоуроков. Такие усилия будут иметь большое значение для прочного внедрения технологии.

Ключевые моменты

Даже лучшие компании испытывают трудности с внедрением ИИ. По этой причине необходимо проявлять большую осторожность, усердие и планировать работу. Также важно понимать, что неудачи – обычное явление.

Существует два главных способа использования ИИ в компании: с помощью программно-информационного приложения разработчика/поставщика либо с помощью собственной модели. Последнее сделать гораздо сложнее и требует от организации серьезной решимости.

При использовании готовых приложений ИИ еще предстоит проделать большую работу. Например, если сотрудники неправильно вводят данные, то результаты, скорее всего, будут искажены.

Образование имеет решающее значение при внедрении ИИ, даже для опытных инженеров. Отличные учебные онлайновые ресурсы оказывают в этом помощь.

Главные части процесса внедрения ИИ включают следующее: определение задачи, которую необходимо решить; создание сильной команды; выбор подходящих инструментов и платформ; строительство модели ИИ; развертывание и мониторинг модели ИИ.

Сформировать команду непросто, поэтому торопить этот процесс не стоит. Следует иметь лидера с хорошим управленческим или эксплуатационным опытом в сочетании техническими навыками.

Некоторые популярные инструменты ИИ включают библиотеки TensorFlow, PyTorch, Keras, язык Python и блокнот Jupyter.

Инструменты автоматизированного машинного обучения, или autoML, помогают справляться с такими процессами, как подготовка данных и отбор признаков для моделей ИИ. В центре внимания этих инструментов находятся те, кто не обладает техническими навыками.

Развертывание модели ИИ – это больше, чем просто масштабирование. Кроме того, крайне важно, чтобы система была простой в использовании с целью обеспечения гораздо более широкого ее внедрения.

Вывод

Подходя к внедрению ИИ, важно рассмотреть два пути. Первый – максимально применять любые сторонние системы, использующие данную технологию. Но следует также уделять особое внимание качеству данных. Если этого не сделать, то результаты, скорее всего, будут не на высоте.

Второй путь – это создание проекта в области ИИ, основанного на собственных данных компании. Для того чтобы добиться успеха в этом деле, должна существовать сильная команда, члены которой имеют сочетание технических, деловых и предметных знаний.

На этапах проекта не должно быть никакой спешки: оценивание ИТ-среды, постановка четкой бизнес-цели, очистка данных, подбор подходящих инструментов и платформ, создание модели ИИ и развертывание системы. С

проектами на ранних стадиях разработки неизбежно возникнут проблемы, поэтому крайне важно быть гибким. Но приложенные усилия должны того стоить.